Information-theoretic perspective on massive multiple-access

Yury Polyanskiy

Department of EECS
MIT

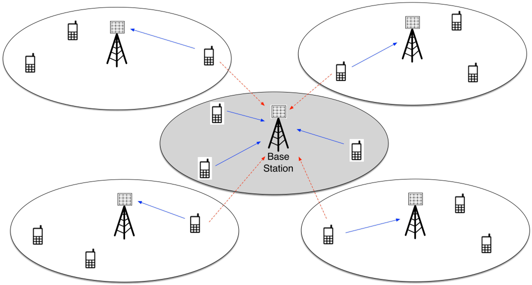
yp@mit.edu

SkolTech Mini Course, Jul. 2018

Lecture plan

1. **Lecture 1: Short packets. Classical MAC.**
   - Motivation: Why work on MAC now? What is new?
   - Finite blocklength IT: a few results
   - Classical MAC IT
2. **Lecture 2: Gaussian MAC. Modulation. CDMA.**
   - Orthogonal modulation (TDMA, FDMA, CDMA) and non-orthogonal (NOMA).
   - Gaussian MAC. TIN. TIN+SIC. Rate-Splitting.
   - Spectral efficiency and E_b/N_0.
   - Randomly-spread CDMA. Effect of MUD.
3. **Lecture 3: Massive MAC. Information theoretic analysis.**
   - Number of users scales with blocklength $K = \mu n$.
   - Per-user probability of error (PUPE). Absence of strong converse.
   - Gaussian-process achievability bound.
4. **Lecture 4: Random-access**
   - Survey of attempts to formalize random-access.
   - Our take: random-access = same-codebook.
   - Achievability bound.
   - Lattice-based coding scheme.

How does your cell phone work?



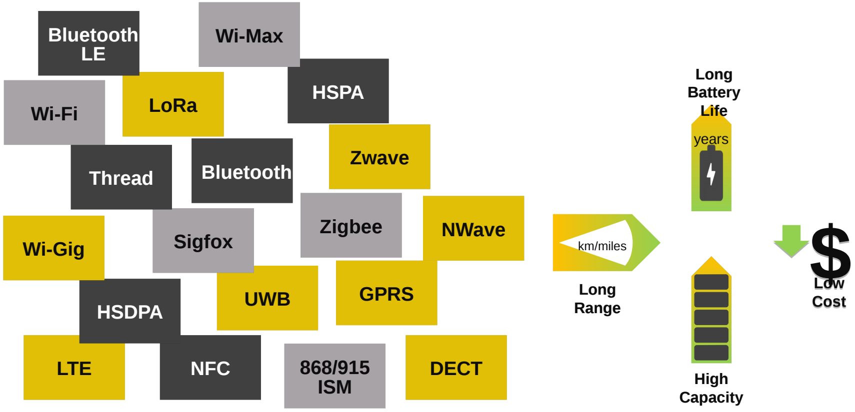
- Cell phone is powered on.
- Announces its presence on PRACH.
- Base station (periodically) gives permission to send.
- Summary:
  - Random-Access is very low duty cycle.
  - BS makes access ORTHOGONAL across users
  - bulk of communication is over an interference-free single-user AWGN.
- What's new in 5G?

Internet-of-Things

- Smart Agriculture
- Advanced Metering systems
- Fire alarms
- Home security and automation
- Oilfield and pipeline monitoring
- M-health
- Smart parking, intelligent traffic
- Waste and recycling
- Asset tracking and geo-location
- Animal tracking and livestock

Expected density: 100-500 devices per household/office

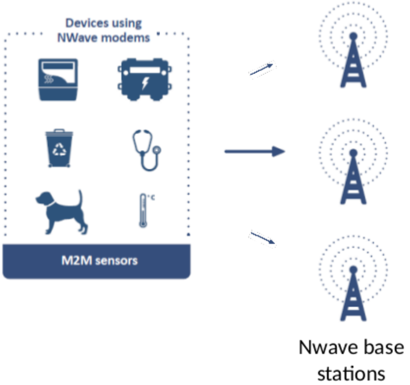
Soup of solutions

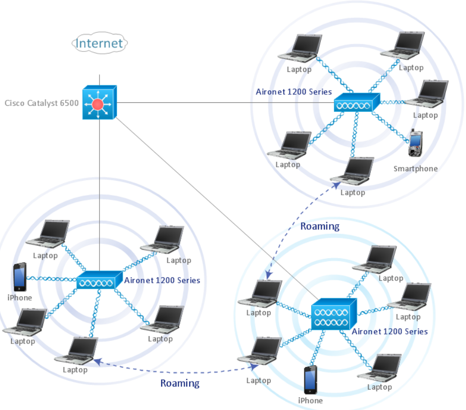


3

Two breeds of IoT

LPWAN





One basestation covers 10 km

IoT is about battery life

Q: What drains the battery? Examples (@ **3.3V**):

	Arduino (w/o reg.)	XBee (Zigbee)	LP-WAN sensor
Sleep	5 uA	1 uA	1-2 uA
CPU Running	50 uA	40 uA	60 uA

IoT is about battery life

Q: What drains the battery? Examples (@ **3.3V**):

	Arduino (w/o reg.)	XBee (Zigbee)	LP-WAN sensor
Sleep	5 uA	1 uA	1-2 uA
CPU Running	50 uA	40 uA	60 uA
Radio Xmit		**40 mA**	**20 mA**

IoT is about battery life

Q: What drains the battery? Examples (@ **3.3V**):

	Arduino (w/o reg.)	XBee (Zigbee)	LP-WAN sensor
Sleep	5 uA	1 uA	1-2 uA
CPU Running	50 uA	40 uA	60 uA
Radio Xmit		**40 mA**	**20 mA**

- Duty-cycle of 1 sec / 20 min radio lasts 6-10 yr / AA bat.
- Caveat: Calculation assumes **single-user**
- Key problem: Energy usage will grow with # of sensors deployed. How much?
- **Sad:** depends on technology? **Happy:** IT comes to rescue!

Outline

Envisioned solution:

- To save battery: sensors sleep all the time, except transmissions.
- ... uncoordinated transmissions.
- ... they wake up, blast the packet, go back to sleep.
- Focus on low-energy (low E_b/N_0)
- Focus on fundamental limits
- ... but with low-complexity solutions (single-user-only decoding).

Outline

Envisioned solution:

- To save battery: sensors sleep all the time, except transmissions.
- ... uncoordinated transmissions.
- ... they wake up, blast the packet, go back to sleep.
- Focus on low-energy (low E_b/N_0)
- Focus on fundamental limits
- ... but with low-complexity solutions (single-user-only decoding).

Issues we need to understand:

1. packets are short: finite-blocklength (FBL) info theory
2. multiple-access channel: Classical MAC
3. low-complexity MAC: modulation, CDMA, multi-user detection
4. massive random-access: many users, same-codebook codes **(NEW)**

Outline

Envisioned solution:

- To save battery: sensors sleep all the time, except transmissions.
- ... uncoordinated transmissions.
- ... they wake up, blast the packet, go back to sleep.
- Focus on low-energy (low E_b/N_0)
- Focus on fundamental limits
- ... but with low-complexity solutions (single-user-only decoding).

Issues we need to understand:

1. packets are short: finite-blocklength (FBL) info theory
2. multiple-access channel: Classical MAC
3. low-complexity MAC: modulation, CDMA, multi-user detection
4. massive random-access: many users, same-codebook codes **(NEW)**

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

FBL Info Theory: short intro

Case study: 1000-bit BSC

- Consider channel $BSC(n = 1000, \delta = 0.11)$
- How many data bits can we transmit with (block) $P_e \leq 10^{-3}$?
- Attempt 1: Repetition

$$k = 47 \text{ bits via [21,1,21]-code}$$

- Attempt 2: Reed-Muller

$$k = 112 \text{ bits via [64,7,32]-code}$$

- Shannon's prediction: $C = 0.5$ bit so

$$k \approx 500 \text{ bit}$$

Case study: 1000-bit BSC

- Consider channel $BSC(n = 1000, \delta = 0.11)$
- How many data bits can we transmit with (block) $P_e \leq 10^{-3}$?
- Attempt 1: Repetition

$$k = 47 \text{ bits via [21,1,21]-code}$$

- Attempt 2: Reed-Muller

$$k = 112 \text{ bits via [64,7,32]-code}$$

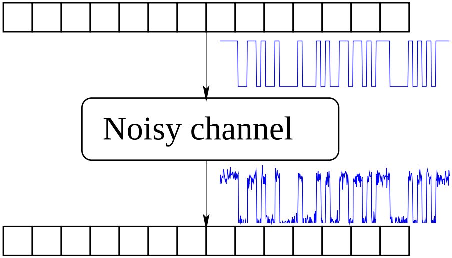
- Shannon's prediction: $C = 0.5$ bit so

$$k \approx 500 \text{ bit}$$

- Finite blocklength IT:

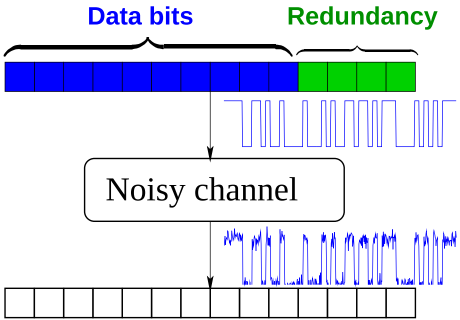
$$414 \leq k \leq 416$$

Abstract communication problem



Goal: Decrease corruption of data caused by noise

Channel coding: principles

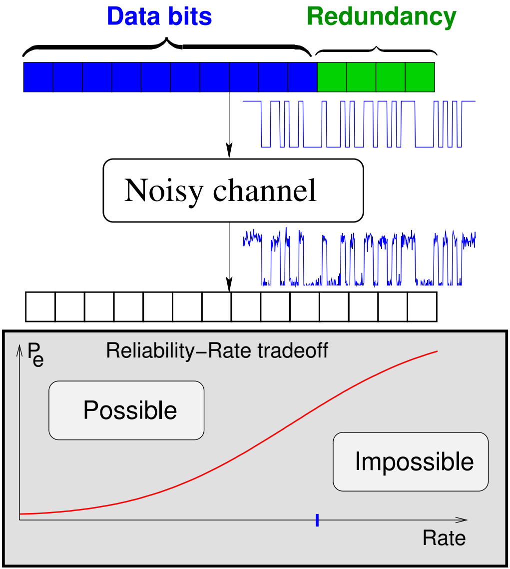


Goal: Decrease corruption of data caused by noise

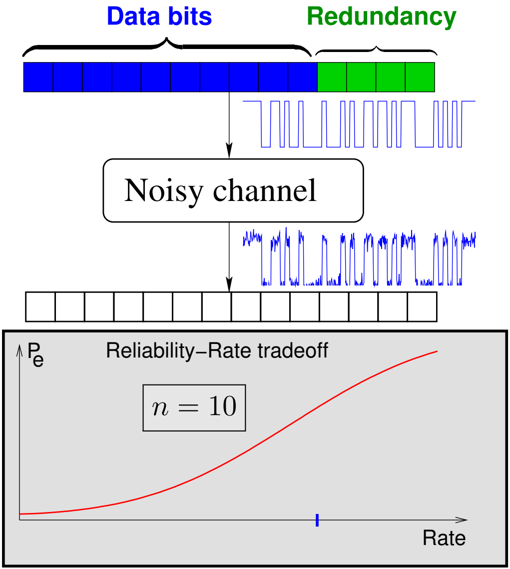
Solution: *Code* to diminish probability of error P_e.

Key metrics: Rate and P_e

Channel coding: principles

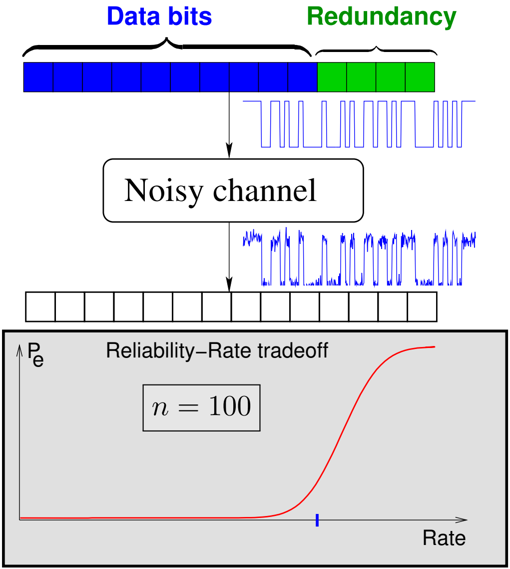


Channel coding: principles



Decreasing P_e further:

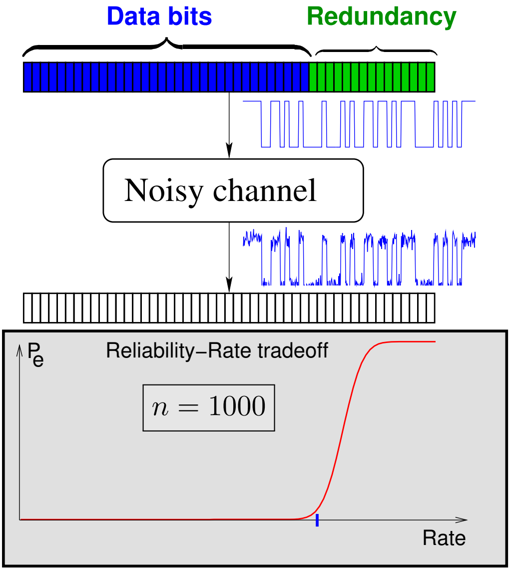
1. More redundancy
   Bad: loses rate
2. Increase blocklength!



Decreasing P_e further:

1. More redundancy
   Bad: loses rate
2. Increase blocklength!

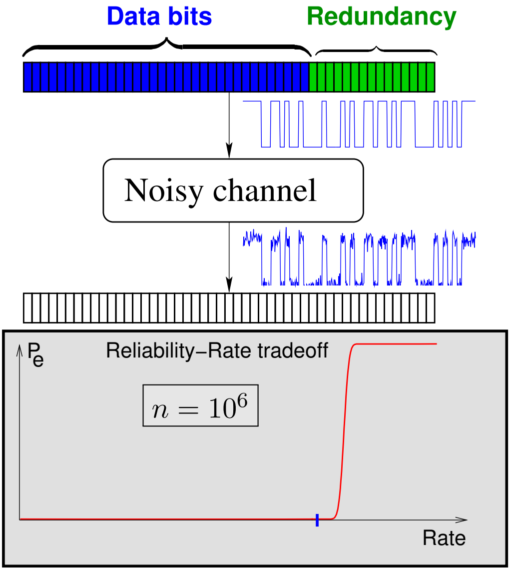
Channel coding: principles



Decreasing P_e further:

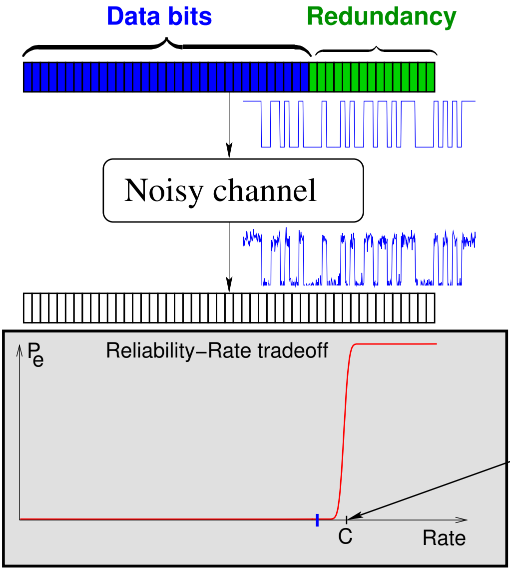
1. More redundancy
   Bad: loses rate
2. Increase blocklength!

Channel coding: principles



Decreasing P_e further:

1. More redundancy
   Bad: loses rate
2. Increase blocklength!

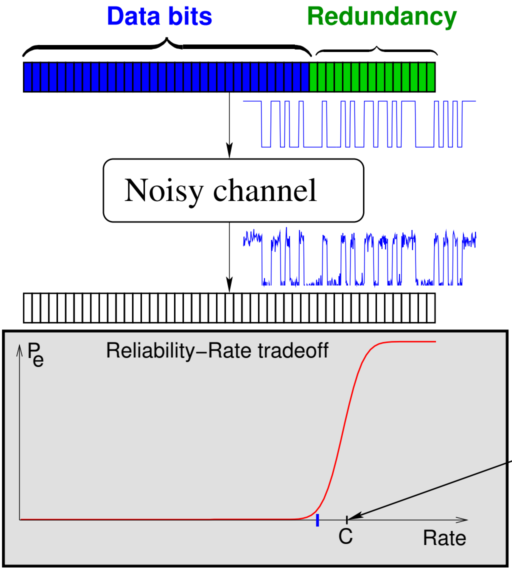


Shannon: Fix $R < C$

$P_e \searrow 0$ as $n \to \infty$

Channel capacity

Channel coding: Shannon capacity



Channel capacity

Shannon: Fix $R < C$

$P_e \searrow 0$ as $n \to \infty$

Question:

For what n will $P_e < 10^{-3}$?

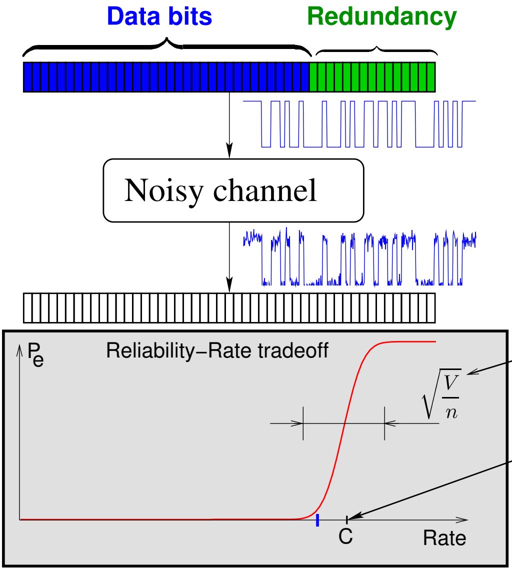
Channel coding: Gaussian approximation

Shannon: Fix $R < C$

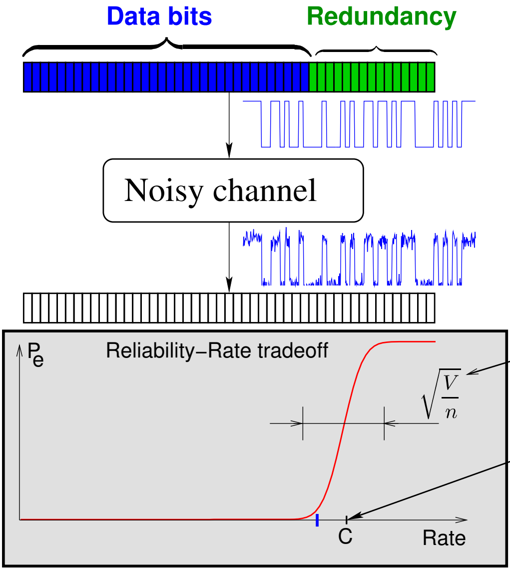
$P_e \searrow 0$ as $n \to \infty$

Question:

For what n will $P_e < 10^{-3}$?



Channel coding: Gaussian approximation



Shannon: Fix $R < C$

$P_e \searrow 0$ as $n \to \infty$

Question:

For what n will $P_e < 10^{-3}$?

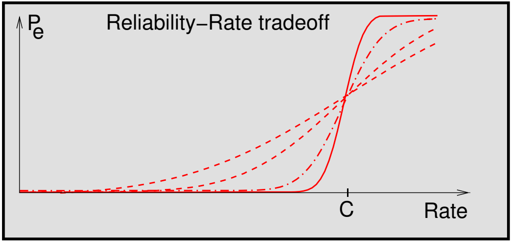
Answer:

$$n \gtrsim \text{const} \cdot \frac{V}{C^2}$$

Channel dispersion

Channel capacity

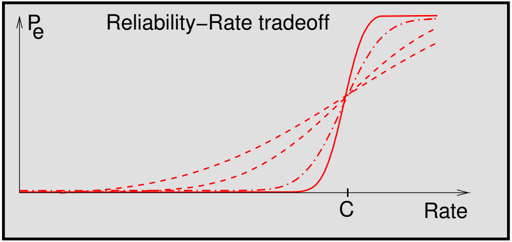
How to describe evolution of the boundary?



Classical results:

- **Vertical asymptotics:** fixed rate, reliability function
  Elias, Dobrushin, Fano, Shannon-Gallager-Berlekamp
- **Horizontal asymptotics:** fixed ϵ, strong converse, $\sqrt{n}$ terms
  Wolfowitz, Weiss, Dobrushin, Strassen, Kemperman

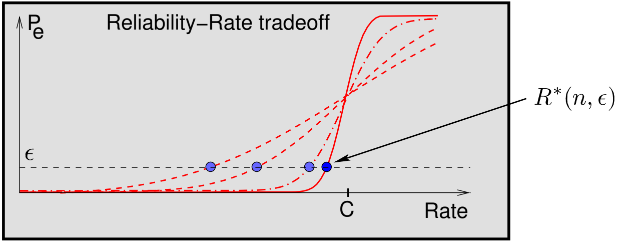
How to describe evolution of the boundary?



XXI century:

- Tight non-asymptotic bounds
- Remarkable precision of normal approximation
- Extended results on *horizontal* asymptotics
  AWGN, $O(\log n)$, cost constraints, feedback, *etc.*

Finite blocklength fundamental limit



Definition

$$R^*(n,\epsilon) = \max\left\{\frac{1}{n}\log M : \exists (n, M, \epsilon)\text{-code}\right\}$$

(max. achievable rate for blocklength n and prob. of error ϵ)

Rough summary: For ergodic channels

$$R^*(n,\epsilon) \approx C - \sqrt{\frac{V}{n}} Q^{-1}(\epsilon)\,.$$

Connection to CLT

- Let $P_{Y^n|X^n} = P^n_{Y|X}$ be memoryless.
- Converse bounds (roughly):

$$R^*(n,\epsilon) \lesssim \epsilon\text{-th quantile of } \frac{1}{n}\log\frac{dP_{Y^n|X^n}}{dQ_{Y^n}}$$

- Achievability bounds (roughly):

$$R^*(n,\epsilon) \gtrsim \epsilon\text{-th quantile of } \frac{1}{n}\log\frac{dP_{Y^n|X^n}}{dQ_{Y^n}}$$

Connection to CLT

- Let $P_{Y^n|X^n} = P^n_{Y|X}$ be memoryless.
- Converse bounds (roughly):

$$R^*(n,\epsilon) \lesssim \epsilon\text{-th quantile of } \frac{1}{n}\log\frac{dP_{Y^n|X^n}}{dQ_{Y^n}}$$

- Achievability bounds (roughly):

$$R^*(n,\epsilon) \gtrsim \epsilon\text{-th quantile of } \frac{1}{n}\log\frac{dP_{Y^n|X^n}}{dQ_{Y^n}}$$

- Info-density $\boxed{i(X^n;Y^n) = \log\frac{dP_{Y^n|X^n}}{dQ_{Y^n}}}$ is a sum of iid.
- Choice of Q_{Y^n} is an art. Often c.a.o.d. works. Then, $\mathbb{E}[i(X^n;Y^n)] = nC$.
- So by CLT

$$R^*(n,\epsilon) \approx \epsilon\text{-quantile of } \mathcal{N}(C, V/n)$$

FBL achievability bounds

- A random transformation A $\xrightarrow{P_{Y|X}}$ B
- (M, ϵ) codes:

$$W \to \mathsf{A} \to \mathsf{B} \to \hat{W} \qquad W \sim Unif\{1, \ldots, M\}$$

$$\mathbb{P}[W \neq \hat{W}] \leq \epsilon$$

- For every $P_{XY} = P_X P_{Y|X}$ define **information density**:

$$\imath(x; y) \triangleq \log \frac{dP_{Y|X=x}}{dP_Y}(y)$$

  - $\mathbb{E}[\imath(X;Y)] = I(X;Y)$
  - $\mathrm{Var}[\imath(X;Y)|X] = V$
  - Memoryless channels: $\imath(A^n; B^n) =$ sum of iid.

$$\imath(A^n; B^n) \stackrel{d}{\approx} nI(A;B) + \sqrt{nV} Z, \qquad Z \sim \mathcal{N}(0,1)$$

- Goal: Prove FBL bounds.
  As by-product: $R^*(n, \epsilon) \gtrsim C - \sqrt{\frac{V}{n}} Q^{-1}(\epsilon)$

DT bound

Theorem (Dependence Testing Bound)

For any P_X there exists a code with M codewords and

$$\epsilon \le \mathbb{E}\left[\exp\left\{-\left|\imath_{X;Y}(X;Y) - \log\frac{M-1}{2}\right|^+\right\}\right].$$

Highlights:

- Strictly stronger than Feinstein-Shannon
- ... and no optimization over γ!
- Easier to compute than RCU
- Easier asymptotics: $\epsilon \le \mathbb{E}\left[e^{-n|\frac{1}{n}\imath(X^n;Y^n)-R|^+}\right]$
  $\approx Q\left(\sqrt{\frac{n}{V}}\{I(X;Y) - R\}\right)$
- Has a form of f-divergence: $1 - \epsilon \ge D_f(P_{XY} \| P_X P_Y)$

DT bound: Proof

- Codebook: random $C_1, \ldots C_M \sim P_X$ iid
- Feinstein decoder:

$$\hat{W} = \text{smallest } j \text{ s.t. } \imath_{X;Y}(C_j; Y) > \gamma$$

- j-th codeword's probability of error:

$$\mathbb{P}[\text{error} \mid W = j] \le \underbrace{\mathbb{P}[\imath_{X;Y}(X;Y) \le \gamma]}_{a} + (j-1)\underbrace{\mathbb{P}[\imath_{X;Y}(\bar{X};Y) > \gamma]}_{b}$$

In a: C_j too far from Y
In b: C_k with $k < j$ is too close to Y

- Average over W:

$$\mathbb{P}[\text{error}] \quad \le \quad \mathbb{P}\left[\imath_{X;Y}(X;Y) \le \gamma\right] + \frac{M-1}{2}\mathbb{P}\left[\imath_{X;Y}(\bar{X};Y) > \gamma\right]$$

DT bound: Proof

- Recap: for every γ there exists a code with

$$\epsilon \le \mathbb{P}\left[\imath_{X;Y}(X;Y) \le \gamma\right] + \frac{M-1}{2}\mathbb{P}\left[\imath_{X;Y}(\bar{X};Y) > \gamma\right] .$$

- Key step: closed-form optimization of γ.
- Introduce $\bar{X} \perp\!\!\!\perp Y$: $\imath_{X;Y} = \log \frac{dP_{XY}}{dP_{\bar{X}Y}}$
- We have

$$P_{XY}\left[\frac{dP_{XY}}{dP_{\bar{X}Y}} \le e^{\gamma}\right] + \frac{M-1}{2} P_{\bar{X}Y}\left[\frac{dP_{XY}}{dP_{\bar{X}Y}} > e^{\gamma}\right]$$

Bayesian dependence testing!

Optimum threshold: Ratio of priors $\Rightarrow$ $\boxed{\gamma^* = \log \frac{M-1}{2}}$

- Change of measure argument:

$$P\left[\frac{dP}{dQ} \le \tau\right] + \tau Q\left[\frac{dP}{dQ} > \tau\right] = \mathbb{E}_P\left[\exp\left\{-\left|\log\frac{dP}{dQ} - \log\tau\right|^+\right\}\right] .$$

FBL Converse bounds

- Take a random transformation $\mathsf{A} \xrightarrow{P_{Y|X}} \mathsf{B}$
  (think $\mathsf{A} = \mathcal{A}^n$, $\mathsf{B} = \mathcal{B}^n$, $P_{Y|X} = P_{Y^n|X^n}$)
- Input distribution P_X induces $P_Y = P_{Y|X} \circ P_X$
  $P_{XY} = P_X P_{Y|X}$
- Fix code:

$$W \xrightarrow{encoder} X \to Y \xrightarrow{decoder} \hat{W}$$

  $W \sim Unif[M]$ and $M = \#$ of codewords
  Input distribution P_X associated to a code:

$$P_X[\cdot] \triangleq \frac{\#\text{ of codewords } \in (\cdot)}{M}.$$

- Goal: Upper bounds on $\log M$ in terms of $\epsilon \triangleq \mathbb{P}[error]$
  As by-product: $R^*(n, \epsilon) \lesssim C - \sqrt{\frac{V}{n}} Q^{-1}(\epsilon)$

Fano's inequality

Theorem (Fano)

For any code

$$W \xrightarrow{\text{encoder}} X \xrightarrow{P_{Y|X}} Y \xrightarrow{\text{decoder}} \hat{W}$$

with $W \sim$ *Unif*$\{1, \ldots, M\}$*:*

$$\log M \le \frac{\sup_{P_X} I(X;Y) + h(\epsilon)}{1-\epsilon}, \qquad \epsilon = \mathbb{P}[W \neq \hat{W}]$$

Implies *weak converse*:

$$R^*(n,\epsilon) \le \frac{C}{1-\epsilon} + o(1)\,.$$

Proof: ϵ-small $\implies H(W|\hat{W})$-small $\implies I(X;Y) \approx H(W) = \log M$

A (very long) proof of Fano via *channel substitution*

Consider two distributions on $(W, X, Y, \hat{W})$:

$$\mathbb{P}: \quad P_{WXY\hat{W}} = P_W \times P_{X|W} \times P_{Y|X} \times P_{\hat{W}|Y}$$

DAG: $W \to X \to Y \to \hat{W}$

$$\mathbb{Q}: \quad Q_{WXY\hat{W}} = P_W \times P_{X|W} \times Q_Y \times P_{\hat{W}|Y}$$

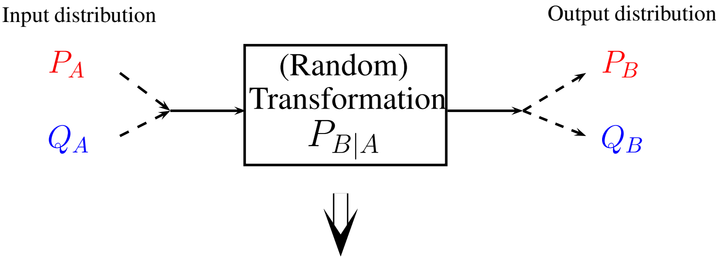
DAG: $W \to X \_\_\_ Y \to \hat{W}$

Under $\mathbb{Q}$ the channel is useless:

$$\mathbb{Q}[W = \hat{W}] = \sum_{m=1}^{M} P_W(m) Q_{\hat{W}}(m) = \frac{1}{M} \sum_{m=1}^{M} Q_{\hat{W}}(m) = \frac{1}{M}$$

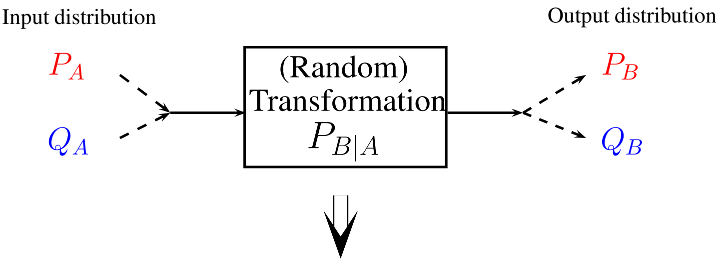
Next step: data-processing for relative entropy $D(\cdot||\cdot)$

Data-processing for $D(\cdot||\cdot)$



$$D(P_A\|Q_A) \quad \ge \quad D(P_B\|Q_B)$$

Data-processing for $D(\cdot||\cdot)$



$$D(P_A \| Q_A) \qquad \geq \qquad D(P_B \| Q_B)$$

Apply to transform: $(W, X, Y, \hat{W}) \mapsto 1\{W \neq \hat{W}\}$:

$$\begin{aligned} D(P_{WXY\hat{W}} \| Q_{WXY\hat{W}}) &\geq d(\,\mathbb{P}[W = \hat{W}] \,\|\, \mathbb{Q}[W = \hat{W}]\,) \\ &= d(1 - \epsilon || \tfrac{1}{M}) \end{aligned}$$

where $d(x||y) = x \log \frac{x}{y} + (1-x) \log \frac{1-x}{1-y}$.

A proof of Fano via *channel substitution*

So far:

$$D(P_{WXY\hat{W}} \| Q_{WXY\hat{W}}) \ge d(1-\epsilon \| \tfrac{1}{M})$$

Lower-bound RHS:

$$d(1-\epsilon \| \tfrac{1}{M}) \ge (1-\epsilon)\log M - h(\epsilon)$$

Analyze LHS:

$$\begin{aligned} D(P_{WXY\hat{W}} \| Q_{WXY\hat{W}}) &= D(P_{XY} \| Q_{XY}) \\ &= D(P_X P_{Y|X} \| P_X Q_Y) \\ &= D(P_{Y|X} \| Q_Y | P_X) \end{aligned}$$

(Recall: $D(P_{Y|X} \| Q_Y | P_X) = \mathbb{E}_{x\sim P_X}[D(P_{Y|X=x} \| Q_Y)]$)

A proof of Fano via *channel substitution*: last step

Putting it all together:

$$\boxed{(1-\epsilon)\log M \le D(P_{Y|X}\|Q_Y|P_X) + h(\epsilon) \quad \forall Q_Y \quad \forall \text{code}}$$

Two methods:

1. Compute $\sup_{P_X} \inf_{Q_Y}$ and recall

$$\inf_{Q_Y} D(P_{Y|X}\|Q_Y|P_X) = I(X;Y)$$

2. Take $Q_Y = P_Y^* =$ the caod (capacity achieving output dist.) and recall

$$D(P_{Y|X}\|P_Y^*|P_X) \le \sup_X I(X;Y) \qquad \forall P_X$$

Conclude:

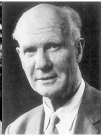
$$(1-\epsilon)\log M \le \sup_{P_X} I(X;Y) + h(\epsilon)$$

Important: Second method is particularly useful for FBL!

Tightening: from $D(\cdot||\cdot)$ to $\beta_\alpha(\cdot,\cdot)$

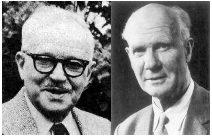
Question: How about replacing $D(\cdot||\cdot)$ with other divergences?

Question: How about replacing $D(\cdot||\cdot)$ with other divergences?

 	$D(\cdot\|\|\cdot)$	relative entropy (KL divergence)	weak converse
	$D_\lambda(\cdot\|\|\cdot)$	Rényi divergence	strong converse
 	$\beta_\alpha(\cdot,\cdot)$	Neyman-Pearson ROC curve	FBL bounds

Tightening: from $D(\cdot||\cdot)$ to $\beta_\alpha(\cdot,\cdot)$

Question: How about replacing $D(\cdot||\cdot)$ with other divergences?

	$D(\cdot\|\|\cdot)$	relative entropy (KL divergence)	weak converse
	$D_\lambda(\cdot\|\|\cdot)$	Rényi divergence	strong converse
	$\beta_\alpha(\cdot,\cdot)$	Neyman-Pearson ROC curve	FBL bounds

Note: Using β_α is aka *meta-converse*.

... and leads to $R^*(n,\epsilon) \le C - \sqrt{\frac{V}{n}}Q^{-1}(\epsilon)$

General meta-converse principle

Steps:

- Select auxiliary channel $Q_{Y|X}$ (art)

  E.g.: $Q_{Y|X=x}$ = center of a cluster of x
- Prove converse bound for channel $Q_{Y|X}$

  E.g.: $\mathbb{Q}[W = \hat{W}] \lesssim \frac{\#\text{ of clusters}}{M}$
- Compute distance $D(\mathbb{P}\|\mathbb{Q})$ between two spaces

$$\mathbb{P}: \; P_{WXY\hat{W}} = P_W \times P_{X|W} \times P_{Y|X} \times P_{\hat{W}|Y}$$

vs.

$$\mathbb{Q}: \; P_{WXY\hat{W}} = P_W \times P_{X|W} \times Q_{Y|X} \times P_{\hat{W}|Y}$$

- Apply data processing: $\boxed{D(P_{W,\hat{W}}\|Q_{W,\hat{W}}) \le D(P_{X,Y}\|Q_{X,Y})}$
- Key observation: This inequality connects $\mathbb{P}$[error], $\mathbb{Q}$[error] and distance $D(\mathbb{P}\|\mathbb{Q})$.

FBL: summary

- All in all, these methods allow us to conclude:

$$R^*(n, \epsilon) \approx C - \sqrt{\frac{V}{n}} Q^{-1}(\epsilon)$$

  for a wide range of channels.
- Typically, $V = \mathrm{Var}[i(X;Y)|X]$ for cap.ach. distribution X.

FBL: summary

- All in all, these methods allow us to conclude:

$$R^*(n,\epsilon) \approx C - \sqrt{\frac{V}{n}} Q^{-1}(\epsilon)$$

  for a wide range of channels.
- Typically, $V = \mathrm{Var}[i(X;Y)|X]$ for cap.ach. distribution X.
- Example: The AWGN Channel

$$\begin{array}{ccccc} & & Z \sim \mathcal{N}(0,\sigma^2) & & \\ & & \downarrow & & \\ X & \longrightarrow & \bigoplus & \longrightarrow & Y \end{array}$$

  Codewords $x^n \in \mathbb{R}^n$ satisfy power-constraint: $\sum_{j=1}^n |x_j|^2 \le nP$

$$C(P) = \frac{1}{2}\log(1+P), \qquad V(P) = \frac{\log^2 e}{2}\left(1 - \frac{1}{(1+P)^2}\right)$$

- Curious property of Gaussian noise: $V(P) \le \frac{\log^2 e}{2}$

FBL: summary

- All in all, these methods allow us to conclude:

$$R^*(n,\epsilon) \approx C - \sqrt{\frac{V}{n}} Q^{-1}(\epsilon)$$

  for a wide range of channels.
- Typically, $V = \mathrm{Var}[i(X;Y)|X]$ for cap.ach. distribution X.
- Example: The AWGN Channel

> Below for Gaussian MAC we focus on m.i./capacity. By FBL there $\exists$ codes within $O(\frac{1}{\sqrt{n}})$ uniformly in P.

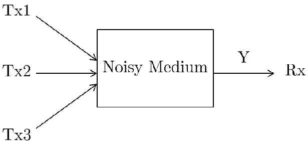
Codewords $x^n \in \mathbb{R}^n$ satisfy power-constraint: $\sum_{j=1}^n |x_j|^2 \le nP$

$$C(P) = \frac{1}{2}\log(1+P), \qquad V(P) = \frac{\log^2 e}{2}\left(1 - \frac{1}{(1+P)^2}\right)$$

- Curious property of Gaussian noise: $V(P) \le \frac{\log^2 e}{2}$

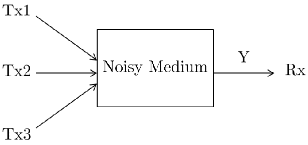
Classical multiple-access IT

IT vs networks view on MAC

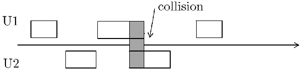


- Core problem: many users, one channel

IT vs networks view on MAC



- Core problem: many users, one channel
- Networking folks:

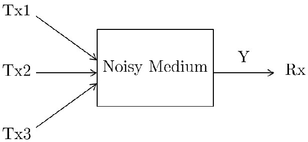


- ALOHA protocol (slotted) achieves:

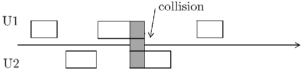
$$\sum_i R_i \approx 0.37C$$

- **Open problem:** what max fraction η^* achievable?
  State of the art [Tsybakov-Lihanov'87]: $0.476 \le \eta^* \le 0.568$
  (collision resolution codes)

IT vs networks view on MAC



- Core problem: many users, one channel
- Networking folks:

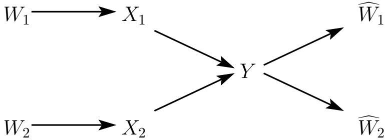


- ALOHA protocol (slotted) achieves:

$$\sum_i R_i \approx 0.37C$$

- **Open problem:** what max fraction η^* achievable?
  State of the art [Tsybakov-Lihanov'87]: $0.476 \leq \eta^* \leq 0.568$
  (collision resolution codes)
- IT: We want $\sum_i R_i \gg C$!
- How? By exploiting physics of collision.

2-user MAC: IT formalism



- 2-input channel: $P_{Y|X_1,X_2}$ (memoryless)
- Random messages $W_1 \in [2^{nR_1}], W_2 \in [2^{nR_2}]$
- Encoders: $X_1^n = f_1(W_1)$, $X_2^n = f_2(W_2)$
- Joint decoder: $(\hat{W}_1, \hat{W}_2) = g(Y)$
- Joint probability of error:

$$\mathbb{P}[W_1 = \hat{W}_1, W_2 = \hat{W}_2] \geq 1 - \epsilon\,.$$

2-user MAC: IT formalism

$$\begin{tikzcd}
W_1 \arrow[r] & X_1 \arrow[dr] & & \widehat{W}_1 \\
& & Y \arrow[ur] \arrow[dr] & \\
W_2 \arrow[r] & X_2 \arrow[ur] & & \widehat{W}_2
\end{tikzcd}$$

- 2-input channel: $P_{Y|X_1,X_2}$ (memoryless)
- Random messages $W_1 \in [2^{nR_1}], W_2 \in [2^{nR_2}]$
- Encoders: $X_1^n = f_1(W_1)$, $X_2^n = f_2(W_2)$
- Joint decoder: $(\hat{W}_1, \hat{W}_2) = g(Y)$
- Joint probability of error:

$$\mathbb{P}[W_1 = \hat{W}_1, W_2 = \hat{W}_2] \geq 1 - \epsilon\,.$$

- FBL fundamental limit (region):

$$R^*(n,\epsilon) = \{(R_1, R_2) : \exists (2^{nR_1}, 2^{nR_2}, \epsilon)\text{-code}\}$$

- Asymptotics: $[\cdot]$ = closure

$$C_\epsilon = \left[\liminf_{n\to\infty} R^*(n,\epsilon)\right], \qquad C = \bigcap_{\epsilon>0} C_\epsilon$$

2-user MAC: capacity region

Theorem (Ahlswede-Liao (capacity) + Dueck (Strong converse))

$$C = C_\epsilon \quad = \quad \left[\mathrm{co}\left\{\bigcup_{P_{X_1},P_{X_2}} \mathrm{Penta}(P_{X_1}, P_{X_2})\right\}\right]$$

$$\mathrm{Penta}(P_{X_1}, P_{X_2}) \triangleq \left\{(R_1, R_2) : \begin{array}{r} R_1 + R_2 \le I(X_1, X_2; Y) \\ R_1 \le I(X_1; Y|X_2) \\ R_2 \le I(X_2; Y|X_1) \end{array}\right\}$$

- $\mathrm{co}\{\cdot\}$ – convex hull
- **Fun fact:** w/o syncronization $C = [\bigcup \mathrm{Penta}]$ but w/o $\mathrm{co}\{\cdot\}$!

2-user MAC: capacity region

Theorem (Ahlswede-Liao (capacity) + Dueck (Strong converse))

$$C = C_\epsilon \quad = \quad \left[\mathrm{co}\left\{\bigcup_{P_{X_1},P_{X_2}} \mathrm{Penta}(P_{X_1}, P_{X_2})\right\}\right]$$

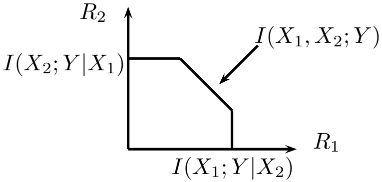
$$\mathrm{Penta}(P_{X_1}, P_{X_2}) \triangleq \left\{(R_1, R_2) : \begin{array}{r} R_1 + R_2 \le I(X_1, X_2; Y) \\ R_1 \le I(X_1; Y|X_2) \\ R_2 \le I(X_2; Y|X_1) \end{array}\right\}$$

- $\mathrm{co}\{\cdot\}$ – convex hull
- **Fun fact:** w/o syncronization $C = [\bigcup \mathrm{Penta}]$ but w/o $\mathrm{co}\{\cdot\}$!
- Not true with cost constraints. In that case need time-sharing:

$$C = \bigcup_{X_1,X_2,U} \left\{(R_1, R_2) : \begin{array}{r} R_1 + R_2 \le I(X_1, X_2; Y|U) \\ R_1 \le I(X_1; Y|X_2, U) \\ R_2 \le I(X_2; Y|X_1, U) \end{array}\right\}.$$

Capacity = Union of pentagons

$$\mathrm{Penta}(P_{X_1}, P_{X_2}) \triangleq \left\{ (R_1, R_2) : \begin{array}{r} R_1 + R_2 \le I(X_1, X_2; Y) \\ R_1 \le I(X_1; Y|X_2) \\ R_2 \le I(X_2; Y|X_1) \end{array} \right\}$$



Note: After taking $\bigcup_{P_{X_1}, P_{X_2}}$ and convex-hull, resulting region may be curvilinear!

MAC theorem: standard proof (outline)

Theorem

$$C = C_\epsilon \quad = \quad \left[\mathrm{co}\left\{\bigcup_{P_{X_1},P_{X_2}} \mathrm{Penta}(P_{X_1}, P_{X_2})\right\}\right]$$

Here is a standard proof

- Weak-converse:
  - sum-rate

$$R_1 + R_2 \lesssim \frac{1}{n} I(X_1^n, X_2^n; Y^n) \le \frac{1}{n}\sum_{i=1}^n I(X_{1i}, X_{2i}; Y_i)\,.$$

  - genie gives X_1^n to decoder

$$R_2 \lesssim \frac{1}{n} I(X_2^n; Y^n | X_1^n) \le \frac{1}{n}\sum_{i=1}^n I(X_{2i}; Y_i | X_{1i})$$

  - Hence $(R_1, R_2) \in \frac{1}{n}\sum_i \mathrm{Penta}(P_{X_{1i}}, P_{X_{2i}})$

MAC theorem: standard proof (outline)

Theorem

$$C = C_\epsilon = \left[\mathrm{co}\left\{\bigcup_{P_{X_1}, P_{X_2}} \mathrm{Penta}(P_{X_1}, P_{X_2})\right\}\right]$$

Here is a standard proof

- Achievability:
  - Fix P_{X_1}, P_{X_2}.
  - Generate codewords for user i from $(P_{X_1})^{\otimes n}$ iid
  - Decode via joint-typicality
  - Have $(M_1 - 1)(M_2 - 1)$ possibilities with both $\hat{W}_1, \hat{W}_2$ wrong (each w.p. $\le 2^{-nI(X_1,X_2;Y)}$)
  - Have $M_i - 1$ possibilities with $\hat{W}_i$ wrong (each w.p. $\le 2^{-nI(X_i;Y|X_{\tilde{i}})}$)
  - Hence, if $(R_1, R_2) \in \mathrm{Penta}(P_{X_1}, P_{X_2})$ all three types of errors are small.
  - Let us understand this more carefully...

MAC achievability: details I

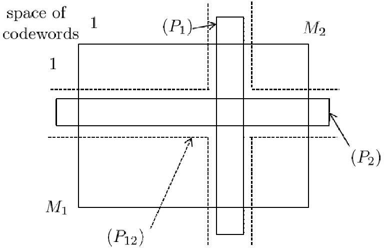
- Gen. $M_1 = 2^{nR_1}$ codewords $C_i \overset{iid}{\sim} (P_{X_1})^{\otimes n}$
- Gen. $M_2 = 2^{nR_2}$ codewords $D_i \overset{iid}{\sim} (P_{X_2})^{\otimes n}$
- True message $W_1 = i_0, W_2 = j_0$.
- Decoder sees y^n. How to decode?

MAC achievability: details I

- Gen. $M_1 = 2^{nR_1}$ codewords $C_i \stackrel{iid}{\sim} (P_{X_1})^{\otimes n}$
- Gen. $M_2 = 2^{nR_2}$ codewords $D_i \stackrel{iid}{\sim} (P_{X_2})^{\otimes n}$
- True message $W_1 = i_0, W_2 = j_0$.
- Decoder sees y^n. How to decode?
- Why is this not the same as decoding single-user $M_1 \times M_2$-size code?

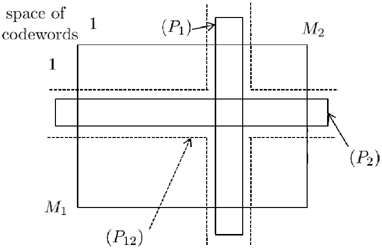
MAC achievability: details I

- Gen. $M_1 = 2^{nR_1}$ codewords $C_i \overset{iid}{\sim} (P_{X_1})^{\otimes n}$
- Gen. $M_2 = 2^{nR_2}$ codewords $D_i \overset{iid}{\sim} (P_{X_2})^{\otimes n}$
- True message $W_1 = i_0, W_2 = j_0$.
- Decoder sees y^n. How to decode?
- Why is this not the same as decoding single-user $M_1 \times M_2$-size code?



- Extra structure: $(C_{i_0}, D_j) \not\perp\!\!\!\perp (C_{i_0}, D_{j_0})$

MAC achievability: details II



- Decoder sees y^n. How to decode?
- A good test for rejecting $(M_1 - 1)(M_2 - 1)$ codewords in (P_{12}):

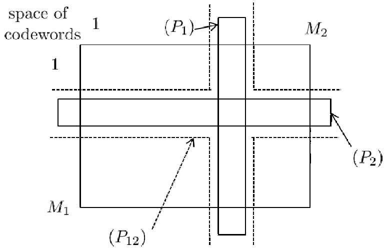
(T_{12}) $\quad i(c_i, d_j; y^n) \le \gamma_{12}$ $\quad \Rightarrow$ remove (i, j) from consideration

- $i(c, d; y^n) \triangleq \log \frac{P_{Y^n|X_1^n, X_2^n}(y^n|c,d)}{P_{Y^n}(y^n)}$
- Standard bound: $\forall i \ne i_0, j \ne j_0$:

$$\mathbb{P}[i(C_i, D_j; Y^n) > \gamma_{12}] \le e^{-\gamma_{12}}$$

- Set $\gamma_{12} = \log(M_1 M_2) + \tau$ then test (T_{12}) filters all $(i, j) \in (P_{12})$

MAC achievability: details III



- Decoder sees y^n. How to decode?
- A good test for rejecting $(M_2 - 1)$ codewords in (P_2):

(T_2) $\quad i(d_j; y^n | c_i) \le \gamma_2 \quad \Rightarrow$ remove (i, j) from consideration

- $i(d; y^n | c) \triangleq \log \frac{P_{Y^n | X_1^n, X_2^n}(y^n | c, d)}{P_{Y^n | X_1^n}(y^n | c)}$
- Standard bound: $\forall j \neq j_0$:

$$\mathbb{P}[i(D_j; Y^n | C_{i_0}) > \gamma_2] \le e^{-\gamma_2}$$

- Set $\gamma_2 = \log(M_2) + \tau$ then test (T_2) filters all $(i_0, j) \in (P_2)$

- Decoder sees y^n. How to decode?

(T_{12})	$i(c_i, d_j; y^n) \le n(R_1 + R_2) + \tau$	$\Rightarrow$ remove (i,j)
(T_1)	$i(c_i; y^n \mid d_j) \le nR_1 + \tau$	$\Rightarrow$ remove (i,j)
(T_2)	$i(d_j; y^n \mid c_i) \le nR_2 + \tau$	$\Rightarrow$ remove (i,j)

- This achieves:

$$\begin{aligned}\epsilon \;\le\; & 3e^{-\tau} + \mathbb{P}\Big[\{i(X_1^n, X_2^n; Y^n) \le n(R_1+R_2)+\tau\} \cup \\ & \{i(X_1^n; Y^n \mid X_2^n) \le nR_1 + \tau\} \cup \{i(X_2^n; Y^n \mid X_1^n) \le nR_2 + \tau\}\Big].\end{aligned}$$

- By CLT a (R_1, R_2) within $\frac{1}{\sqrt{n}}$ of the boundary of Penta is achievable.

- Decoder sees y^n. How to decode?

(T_{12})	$i(c_i, d_j; y^n) \le n(R_1 + R_2) + \tau$	$\Rightarrow$ remove (i, j)
(T_1)	$i(c_i; y^n \mid d_j) \le nR_1 + \tau$	$\Rightarrow$ remove (i, j)
(T_2)	$i(d_j; y^n \mid c_i) \le nR_2 + \tau$	$\Rightarrow$ remove (i, j)

- This achieves:

$$\begin{aligned}\epsilon \;\; \le \;\; & 3e^{-\tau} + \mathbb{P}\Big[\{i(X_1^n, X_2^n; Y^n) \le n(R_1 + R_2) + \tau\} \cup \\ & \{i(X_1^n; Y^n \mid X_2^n) \le nR_1 + \tau\} \cup \{i(X_2^n; Y^n \mid X_1^n) \le nR_2 + \tau\}\Big].\end{aligned}$$

- By CLT a (R_1, R_2) within $\frac{1}{\sqrt{n}}$ of the boundary of Penta is achievable.
- Typical decoding
  - Use (T_{12}) rule – this is like decoding single-user $M_1 \times M_2$-code (LDPC+LDGM structure!)
  - After applying it, most often get only one (true) message left (!)

- Decoder sees y^n. How to decode?

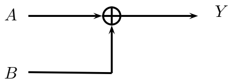
(T_{12})	$i(c_i, d_j; y^n) \le n(R_1 + R_2) + \tau$	$\Rightarrow$ remove (i,j)
(T_1)	$i(c_i; y^n \| d_j) \le nR_1 + \tau$	$\Rightarrow$ remove (i,j)
(T_2)	$i(d_j; y^n \| c_i) \le nR_2 + \tau$	$\Rightarrow$ remove (i,j)

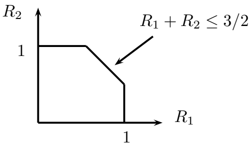
- This achieves:

$$\epsilon \le 3e^{-\tau} + \mathbb{P}\Big[\{i(X_1^n, X_2^n; Y^n) \le n(R_1+R_2)+\tau\} \cup \{i(X_1^n; Y^n|X_2^n) \le nR_1 + \tau\} \cup \{i(X_2^n; Y^n|X_1^n) \le nR_2 + \tau\}\Big].$$

- By CLT a (R_1, R_2) within $\frac{1}{\sqrt{n}}$ of the boundary of Penta is achievable.
- Typical decoding
  - Use (T_{12}) rule – this is like decoding single-user $M_1 \times M_2$-code (LDPC+LDGM structure!)
  - After applying it, most often get only one (true) message left (!)
  - Unless $R_1 = I(X_1; Y|X_2) + O(\frac{1}{\sqrt{n}})$.
  - In this case, many (i,j)'s remain. But they are all in one column!
  - Hence decode W_2. Conditioned on X_2 – decode M_1-code.

Example: Binary Adder Channel (BAC)





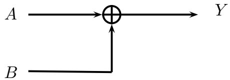
$$Y = X_1 + X_2 \qquad X_i \in \{0,1\}, Y \in \{0,1,2\}$$

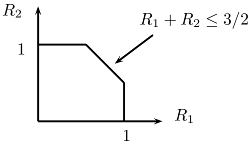
- Maximal sum-rate:

$$C_{sum} = \max_{A,B} I(A,B;Y) = \max H(A+B) = \frac{3}{2}\log 2$$

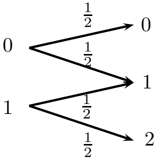
- Each user can send 1 bit/ch.use. But together $\frac{3}{2}$ bit/ch.use. How?

Example: Binary Adder Channel (BAC)

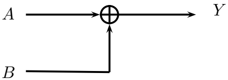


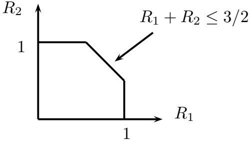


- Take $R_1 = 1$. Then $X_2 \to Y$ sees channel:

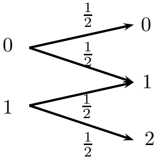


Example: Binary Adder Channel (BAC)

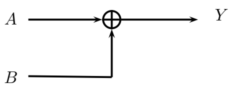


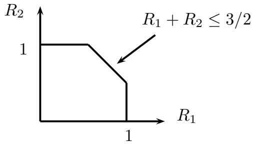


- Take $R_1 = 1$. Then $X_2 \to Y$ sees channel:

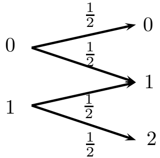


$= \textsf{BEC}(1/2)$



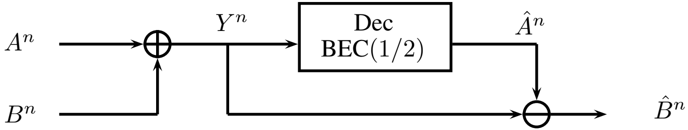


- Take $R_1 = 1$. Then $X_2 \to Y$ sees channel:

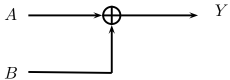


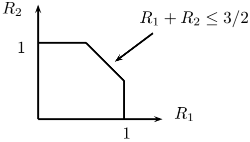
$= \textbf{BEC}(1/2)$

- successive interference cancellation (SIC):



Example: Binary Adder Channel (BAC)





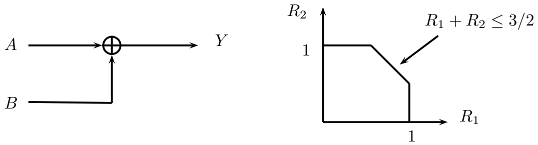
$$Y = X_1 + X_2 \qquad X_i \in \{0, 1\}, Y \in \{0, 1, 2\}$$

- Analyzing FBL achievability we can show: (maximal sumrate)

$$R^*_{sum}(n, \epsilon) \geq \frac{3}{2} - \sqrt{\frac{1}{4n}} Q^{-1}(\epsilon) + O(\log n)\,.$$

- **Open problem:** Prove $R^*_{sum}(n, \epsilon) \leq \frac{3}{2} + \sqrt{\frac{1}{n}} K_\epsilon$

Example: Binary Adder Channel (BAC)



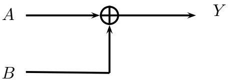
$$Y = X_1 + X_2 \qquad X_i \in \{0,1\}, Y \in \{0,1,2\}$$

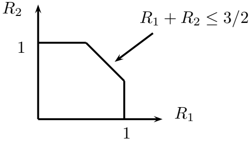
- Analyzing FBL achievability we can show: (maximal sumrate)

$$R^*_{sum}(n,\epsilon) \ge \frac{3}{2} - \sqrt{\frac{1}{4n}}Q^{-1}(\epsilon) + O(\log n)\,.$$

- **Open problem:** Prove $R^*_{sum}(n,\epsilon) \le \frac{3}{2} + \sqrt{\frac{1}{n}}K_\epsilon$
- ... not even asking for $K_\epsilon < 0$
- ... So far best-known result (Ahslwede): $R^*_{sum} \le \frac{3}{2} + c\sqrt{\frac{1}{n}}\log n$

Example: Binary Adder Channel (BAC)





$$Y = X_1 + X_2 \qquad X_i \in \{0,1\}, Y \in \{0,1,2\}$$

- Analyzing FBL achievability we can show: (maximal sumrate)

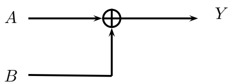
$$R^*_{sum}(n,\epsilon) \ge \frac{3}{2} - \sqrt{\frac{1}{4n}} Q^{-1}(\epsilon) + O(\log n)\,.$$

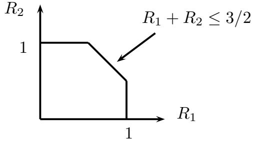
- **Open problem:** Prove $R^*_{sum}(n,\epsilon) \le \frac{3}{2} + \sqrt{\frac{1}{n}} K_\epsilon$
- ... not even asking for $K_\epsilon < 0$
- ... So far best-known result (Ahslwede): $R^*_{sum} \le \frac{3}{2} + c\sqrt{\frac{1}{n}} \log n$
- The state is so bad that even for $\epsilon = 0$ we only know (Fano):

$$R^*_{sum}(n,\epsilon = 0) \le \frac{3}{2}$$

- **Open problem:** Prove $\lim_{n\to\infty} R^*_{sum}(n,\epsilon = 0) < \frac{3}{2}$.

Example: Binary Adder Channel (BAC)





$$Y = X_1 + X_2 \qquad X_i \in \{0,1\}, Y \in \{0,1,2\}$$

- Analyzing FBL achievability we can show: (maximal sumrate)

$$R^*_{sum}(n,\epsilon) \ge \frac{3}{2} - \sqrt{\frac{1}{4n}}Q^{-1}(\epsilon) + O(\log n)\,.$$

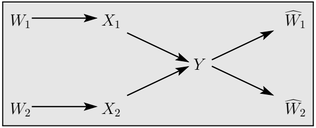
- **Open problem:** Prove $R^*_{sum}(n,\epsilon) \le \frac{3}{2} + \sqrt{\frac{1}{n}}K_\epsilon$
- **Conjecture:** [Ajjanagadde-P.'15] for all $0 < \alpha < 1$

$$\max_{A^n \perp\!\!\!\perp B^n} H_\alpha(A^n + B^n) = nH_\alpha(\tfrac{1}{4}, \tfrac{1}{2}, \tfrac{1}{4})$$

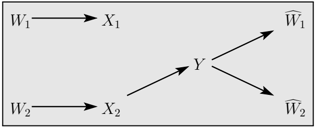
  where $H_\alpha(\cdot)$ is Renyi entropy.
- If true implies **Open problem**. How?

MAC: revisit weak-converse (genie)

$\mathbb{P}$:



$\mathbb{Q}$:



$$\mathbb{P}[\hat{W}_{1,2} = W_{1,2}] = 1 - \epsilon \qquad \mathbb{Q}[\hat{W}_{1,2} = W_{1,2}] = \frac{1}{M_1}$$

... apply data processing of $D(\cdot||\cdot)$ ...

$$\Downarrow$$

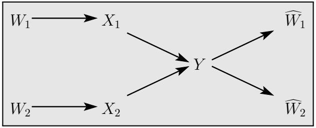
$$d(1-\epsilon\|\tfrac{1}{M_1}) \le D(P_{Y|X_1X_2}\|Q_{Y|X_1}|P_{X_1}P_{X_2})$$

Optimizing $Q_{Y|X_1}$:

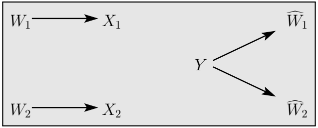
$$\log M_1 \le \frac{I(X_1;Y|X_2) + h(\epsilon)}{1-\epsilon}$$

MAC: revisit weak-converse (genie)

$\mathbb{P}$:



$\mathbb{Q}$:



$$\mathbb{P}[\hat{W}_{1,2} = W_{1,2}] = 1 - \epsilon \qquad \mathbb{Q}[\hat{W}_{1,2} = W_{1,2}] = \frac{1}{M_1 M_2}$$

... apply data processing of $D(\cdot||\cdot)$ ...

$$\Downarrow$$

$$d(1-\epsilon \| \tfrac{1}{M_1}) \le D(P_{Y|X_1X_2} \| Q_Y | P_{X_1} P_{X_2})$$

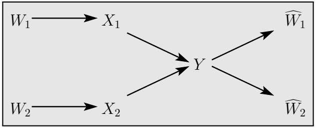
Optimizing Q_Y:

$$\log M_1 M_2 \le \frac{I(X_1, X_2; Y) + h(\epsilon)}{1 - \epsilon}$$

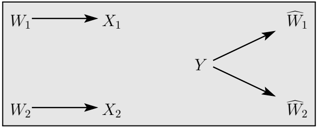
Together with previous: full (pentagon) weak converse

MAC: towards strong-converse

$\mathbb{P}$:



$\mathbb{Q}$:



$$\mathbb{P}[\hat{W}_{1,2} = W_{1,2}] = 1 - \epsilon \qquad \mathbb{Q}[\hat{W}_{1,2} = W_{1,2}] = \frac{1}{M_1 M_2}$$

... use Renyi $D_\lambda(\cdot\|\cdot)$ ...

$$\Downarrow$$

$$D_\lambda(P_{X_1X_2Y} \| P_{X_1}P_{X_2}Q_Y) \geq d_\lambda(1-\epsilon \| \tfrac{1}{M_1M_2})$$

Selecting $\lambda = 1 + \frac{1}{\sqrt{n}}$ yields (for BAC)

$$\log M_1, M_2 \leq \sup_{A^n \perp\!\!\!\perp B^n} H_{\alpha_n}(A^n + B^n) + K\sqrt{n}$$

with $\alpha_n = 1 - \frac{1}{\sqrt{n}}$.

Classical MAC: summary

- Trivially generalizes to K-user MAC:

$$\text{Penta} = \{(R_1, \ldots, R_K) : \sum_{i \in S} R_i \le I(X_S; Y | X_{S^c}) \forall S \subset [K]\}$$

- Classic IT: Fix K let $n \to \infty$.
- Use joint probability of error:

$$\mathbb{P}[W_1 = \hat{W}_1, \ldots, W_K = \hat{W}_k] \ge 1 - \epsilon\,.$$

- New **FBL** issue: for $K = 100$ need 2^{100} tests in achievability.

Classical MAC: summary

- Trivially generalizes to K-user MAC:

$$\text{Penta} = \{(R_1, \ldots, R_K) : \sum_{i \in S} R_i \leq I(X_S; Y | X_{S^c}) \forall S \subset [K]\}$$

- Classic IT: Fix K let $n \to \infty$.
- Use joint probability of error:

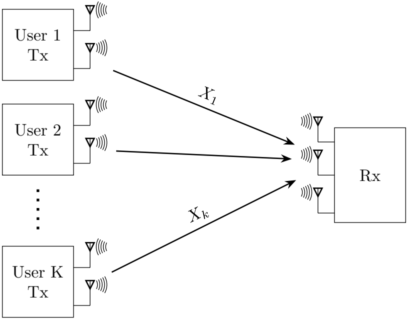
$$\mathbb{P}[W_1 = \hat{W}_1, \ldots, W_K = \hat{W}_k] \geq 1 - \epsilon\,.$$

- New **FBL** issue: for $K = 100$ need 2^{100} tests in achievability.
- What is new today?
  - Many-user scaling [D. Guo et al]: $K = \mu n$, $n \to \infty$
  - New probability of error [P.'17]: $\frac{1}{K} \sum_i \mathbb{P}[W_i \neq \hat{W}_i] \leq \epsilon$
  - Same-codebook coding [P.'17]: $X_i \in \mathcal{C}$ for all i.

Gaussian MAC. Modulation

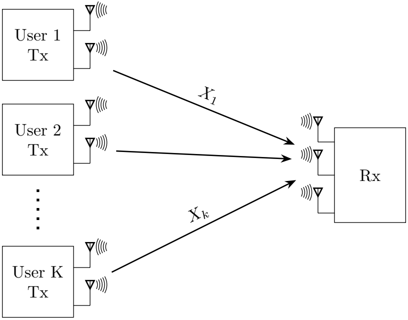
Let's put on our engineering boots.

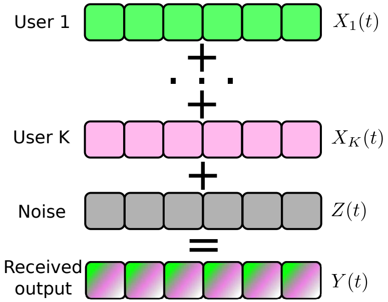
The classical model: K-user multiple-access channel



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

The classical model: K-user multiple-access channel



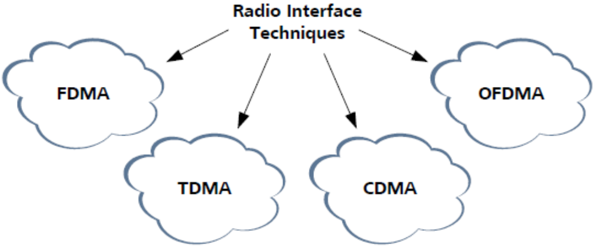


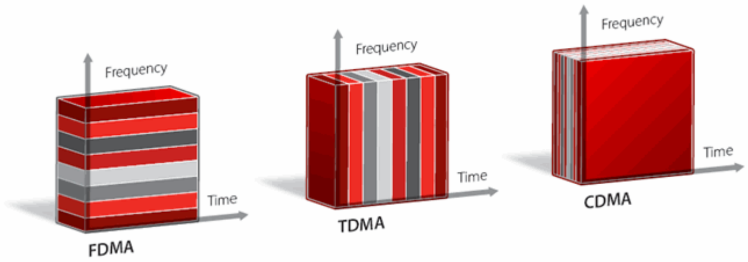
$$Y(t) = X_1(t) + \cdots + X_K(t) + Z(t)$$

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

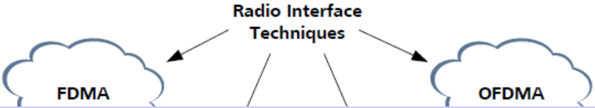
Tech note: synchronized block coding

How to avoid inter-user interference?





How to avoid inter-user interference?

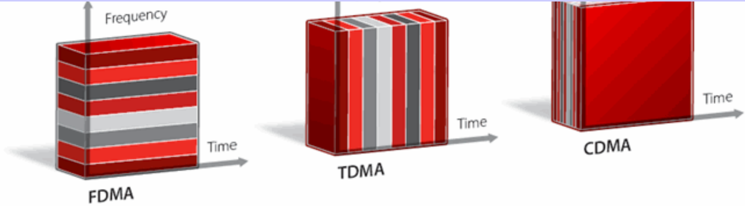


These are called orthogonal schemes

Key problem: resources divided among active and inactive (!) users

(or need costly resource ack/grant protocol)

in IoT most are inactive $\Rightarrow$ huge waste of bandwidth



Orthogonal and non-orthogonal multiple access (NOMA)

This "pie-slicing" philosophy comes from:

- Given: W Hz bandwidth and duration T sec:
- By *XYZ* Theorem: d.o.f. $n = 2WT$

  XYZ $\in$ { Kotelnikov, Nyquist, Shannon, Slepian, ... }
- TDMA, FDMA, CDMA: just different bases in $\mathbb{R}^{2WT}$.
  (Fine print: CDMA = Orthogonal CDMA here).

Orthogonal and non-orthogonal multiple access (NOMA)

This "pie-slicing" philosophy comes from:

- Given: W Hz bandwidth and duration T sec:
- By *XYZ* Theorem: d.o.f. $n = 2WT$

  XYZ $\in$ { Kotelnikov, Nyquist, Shannon, Slepian, ... }
- TDMA, FDMA, CDMA: just different bases in $\mathbb{R}^{2WT}$.
  (Fine print: CDMA = Orthogonal CDMA here).
- Is there value in having $K > n$? (non-orthogonal signalling)
- Is it even possible to have $K > n$ or even $K \gg n$?

Orthogonal and non-orthogonal multiple access (NOMA)

This "pie-slicing" philosophy comes from:

- Given: W Hz bandwidth and duration T sec:
- By *XYZ* Theorem: d.o.f. $n = 2WT$

  XYZ $\in$ { Kotelnikov, Nyquist, Shannon, Slepian, ... }
- TDMA, FDMA, CDMA: just different bases in $\mathbb{R}^{2WT}$.
  (Fine print: CDMA = Orthogonal CDMA here).
- Is there value in having $K > n$? (non-orthogonal signalling)
- Is it even possible to have $K > n$ or even $K \gg n$?
- Silly: Take $n = 1$ and let user j send a bit via $\{0, 2^j\}$.

Orthogonal and non-orthogonal multiple access (NOMA)

This "pie-slicing" philosophy comes from:

- Given: W Hz bandwidth and duration T sec:
- By *XYZ* Theorem: d.o.f. $n = 2WT$

  XYZ $\in$ { Kotelnikov, Nyquist, Shannon, Slepian, ... }
- TDMA, FDMA, CDMA: just different bases in $\mathbb{R}^{2WT}$.
  (Fine print: CDMA = Orthogonal CDMA here).
- Is there value in having $K > n$? (non-orthogonal signalling)
- Is it even possible to have $K > n$ or even $K \gg n$?
- Silly: Take $n = 1$ and let user j send a bit via $\{0, 2^j\}$.
- ... cheating: user K's power is 2^{2K} larger than user 1's.

Orthogonal and non-orthogonal multiple access (NOMA)

This "pie-slicing" philosophy comes from:

- Given: W Hz bandwidth and duration T sec:
- By *XYZ* Theorem: d.o.f. $n = 2WT$

  XYZ $\in$ { Kotelnikov, Nyquist, Shannon, Slepian, ... }
- TDMA, FDMA, CDMA: just different bases in $\mathbb{R}^{2WT}$.
  (Fine print: CDMA = Orthogonal CDMA here).
- Is there value in having $K > n$? (non-orthogonal signalling)
- Is it even possible to have $K > n$ or even $K \gg n$?
- Silly: Take $n = 1$ and let user j send a bit via $\{0, 2^j\}$.
- ... cheating: user K's power is 2^{2K} larger than user 1's.
- Challenge: users only allowed to send ± 1, can we have $K \gg n$?

Achieving capacity of K-user BAC with zero-error

$$Y = \sum_{j=1}^{K} X_j \qquad X_i \in \{\pm 1\}$$

- Known: $C_{sum}(K) = H(\mathrm{Bin}(K, 1/2)) \approx \frac{1}{2}\log K$.
- IOW, for sending 1-bit (each) the frame-length $n \approx \frac{2K}{\log_2 K} \ll K$.

How can $K > n$ users signal in n dimensions simultaneously?

Achieving capacity of K-user BAC with zero-error

$$Y = \sum_{j=1}^{K} X_j \qquad X_i \in \{\pm 1\}$$

- Known: $C_{sum}(K) = H(\mathrm{Bin}(K, 1/2)) \approx \frac{1}{2}\log K$.
- IOW, for sending 1-bit (each) the frame-length $n \approx \frac{2K}{\log_2 K} \ll K$.

How can $K > n$ users signal in n dimensions simultaneously?

- Lindström, Cantor-Mills, Khachatrian-Martirossian: even with zero-error!

  First, recall a particularly nice orthogonal basis:

$$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad H_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad H_{m+1} = \begin{bmatrix} H_m & H_m \\ H_m & -H_m \end{bmatrix}$$

  (each user is modulating his row)

- K.-M. noticed you can add more rows!

Recursive construction (Cantor-Mills, Khachatrian-Martirossian)

How can $K > n$ users signal in n dimensions simultaneously?

- Walsh-Hadamard basis:

$$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad H_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad H_{m+1} = \begin{bmatrix} H_m & H_m \\ H_m & -H_m \end{bmatrix}$$

- K.-M. signals:

$$A_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad A_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix}$$

- Key property: $x \mapsto xA_m$ is injective on $\{\pm 1\}^{K_m}$, $K_m = \frac{m}{2}2^m + 1$

Recursive construction (Cantor-Mills, Khachatrian-Martirossian)

How can $K > n$ users signal in n dimensions simultaneously?

- Walsh-Hadamard basis:

$$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad H_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad H_{m+1} = \begin{bmatrix} H_m & H_m \\ H_m & -H_m \end{bmatrix}$$

- K.-M. signals:

$$A_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad A_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix} \quad \widetilde{A}_{m+1} = \left[\begin{array}{cc} A_m & A_m \\ A_m & -A_m \\ 2^m \left\{ \underbrace{\begin{matrix} 1 & \cdots & 1 \\ 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{matrix}}_{2^m} \right. & \underbrace{\begin{matrix} -1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & \cdots & -1 \end{matrix}}_{2^m} \end{array}\right]$$

- Key property: $x \mapsto xA_m$ is injective on $\{\pm 1\}^{K_m}$, $K_m = \frac{m}{2}2^m + 1$
- Number of users at dimension n: $K \approx \frac{1}{2} n \log_2 n$ (optimal!)

Recursive construction (Cantor-Mills, Khachatrian-Martirossian)

How can $K > n$ users signal in n dimensions simultaneously?

- Walsh-Hadamard basis:

$$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad H_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad H_{m+1} = \begin{bmatrix} H_m & H_m \\ H_m & -H_m \end{bmatrix}$$

- K.-M. signals:

$$A_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad A_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ \boxed{1} & \boxed{1} & \boxed{1} & \boxed{-1} \end{bmatrix} \quad A_{m+1} = \left[\begin{array}{ccc|cccc} & A_m & & & A_m & & \\ & A_m & & & -A_m & & \\ \hline 1 & \cdots & 1 & -1 & 1 & \cdots & 1 \\ 1 & \cdots & 1 & 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & 1 & 1 & \cdots & \cdots & -1 \end{array}\right]$$

2^m $\quad 2^m \quad 2^m$

- Key property: $x \mapsto xA_m$ is injective on $\{\pm 1\}^{K_m}$, $K_m = \frac{m}{2}2^m + 1$
- Number of users at dimension n: $K \approx \frac{1}{2} n \log_2 n$ (optimal!)
- Idea: $(\pm 1)^{2^m} \cdot H_m$ has many "holes"; add ± 1-vectors there.

$$\tilde{A}_{m+1} = \begin{bmatrix} A_m & A_m \\ A_m & -A_m \\ \underbrace{\begin{matrix} 1 & \cdots & 1 \\ 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{matrix}}_{2^m} & \underbrace{\begin{matrix} -1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & \cdots & -1 \end{matrix}}_{2^m} \end{bmatrix} \qquad A_{m+1} = \begin{bmatrix} A_m & A_m \\ A_m & -A_m \\ \underbrace{\begin{matrix} 1 & \cdots & 1 \\ 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{matrix}}_{2^m} & \underbrace{\begin{matrix} -1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & \cdots & -1 \end{matrix}}_{2^m} \end{bmatrix}$$

- Want to show: v is decodable from $v\tilde{A}_m$ for any $v \in \{\pm 1\}^{\otimes K_m}$ and $v_{2K_{m-1}+1} = 0$.
- Equivalently: $v \in \{0,1\}^{\otimes K_m}$ (just use $v \mapsto \frac{1+v}{2}$)

$$\tilde{A}_{m+1} = \begin{array}{c} \\ \\ 2^m \\ \\ \\ \end{array} \left[\begin{array}{ccc|cccc} & A_m & & & A_m & & \\ & A_m & & & -A_m & & \\ 1 & \cdots & 1 & -1 & 1 & \cdots & 1 \\ 1 & \cdots & 1 & 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & 1 & 1 & \cdots & \cdots & -1 \end{array}\right] \qquad A_{m+1} = \begin{array}{c} \\ \\ 2^m \\ \\ \\ \end{array} \left[\begin{array}{ccc|cccc} & A_m & & & A_m & & \\ & A_m & & & -A_m & & \\ 1 & \cdots & 1 & -1 & 1 & \cdots & 1 \\ 1 & \cdots & 1 & 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & 1 & 1 & \cdots & \cdots & -1 \end{array}\right]$$

$$\begin{array}{cc} 2^m & 2^m \end{array} \qquad \begin{array}{cc} 2^m & 2^m \end{array}$$

- Want to show: v is decodable from $v\tilde{A}_m$ for any $v \in \{\pm 1\}^{\otimes K_m}$ and $v_{2K_{m-1}+1} = 0$.
- Equivalently: $v \in \{0, 1\}^{\otimes K_m}$ (just use $v \mapsto \frac{1+v}{2}$)
- Let $v = [x\, y\, z]$ and

$$[x\, y\, z]\tilde{A}_m = [g\, h]$$

$$\tilde{A}_{m+1} = \begin{bmatrix} A_m & A_m \\ A_m & -A_m \\ 2^m \left\{ \begin{matrix} 1 & \cdots & 1 \\ 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{matrix} \right. & \begin{matrix} -1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & \cdots & -1 \end{matrix} \\ \scriptstyle 2^m & \scriptstyle 2^m \end{bmatrix} \qquad A_{m+1} = \begin{bmatrix} A_m & A_m \\ A_m & -A_m \\ 2^m \left\{ \begin{matrix} \cancel{1} & \cancel{\cdots} & \cancel{1} \\ 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{matrix} \right. & \begin{matrix} \cancel{-1} & \cancel{1} & \cancel{\cdots} & \cancel{1} \\ 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & \cdots & -1 \end{matrix} \\ \scriptstyle 2^m & \scriptstyle 2^m \end{bmatrix}$$

- Want to show: v is decodable from $v\tilde{A}_m$ for any $v \in \{\pm 1\}^{\otimes K_m}$ and $v_{2K_{m-1}+1} = 0$.
- Equivalently: $v \in \{0, 1\}^{\otimes K_m}$ (just use $v \mapsto \frac{1+v}{2}$)
- Let $v = [x\, y\, z]$ and

$$[x\, y\, z]\tilde{A}_m = [g\, h] \Rightarrow \quad g - h = [x\, y\, z] \begin{pmatrix} 0 \\ 2A_{m-1} \\ 2I_{2^{m-1}} \end{pmatrix}$$

$$\tilde{A}_{m+1} = \begin{bmatrix} A_m & A_m \\ A_m & -A_m \\ \begin{matrix} 1 & \cdots & 1 \\ 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{matrix} & \begin{matrix} -1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & \cdots & -1 \end{matrix} \end{bmatrix} \qquad A_{m+1} = \begin{bmatrix} A_m & A_m \\ A_m & -A_m \\ \begin{matrix} \cancel{1} & \cancel{\cdots} & \cancel{1} \\ 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{matrix} & \begin{matrix} \cancel{-1} & \cancel{1} & \cancel{\cdots} & \cancel{1} \\ 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & \cdots & -1 \end{matrix} \end{bmatrix}$$

- Want to show: v is decodable from $v\tilde{A}_m$ for any $v \in \{\pm 1\}^{\otimes K_m}$ and $v_{2K_{m-1}+1} = 0$.
- Equivalently: $v \in \{0, 1\}^{\otimes K_m}$ (just use $v \mapsto \frac{1+v}{2}$)
- Let $v = [x\, y\, z]$ and

$$[x\, y\, z]\tilde{A}_m = [g\, h] \Rightarrow \quad g - h = [x\, y\, z] \begin{pmatrix} 0 \\ 2A_{m-1} \\ 2I_{2^{m-1}} \end{pmatrix}$$

- $z_1 = 0$, so by adding $(g-h)_1$ to $(g-h)_\ell$ we get:

(*) $$2z_\ell = (g-h)_1 + (g-h)_\ell - 2y \cdot v_\ell \qquad \ell = 2, \ldots, 2^{m-1}$$

where v_ℓ is sum of 1-st and ℓ-th column of A_{m-1}

$$\tilde{A}_{m+1} = \begin{array}{c} \\ \\ 2^m \end{array} \begin{bmatrix} A_m & A_m \\ A_m & -A_m \\ \underbrace{\begin{matrix} 1 & \cdots & 1 \\ 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{matrix}}_{2^m} & \underbrace{\begin{matrix} -1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & \cdots & -1 \end{matrix}}_{2^m} \end{bmatrix} \qquad A_{m+1} = \begin{array}{c} \\ \\ 2^m \end{array} \begin{bmatrix} A_m & A_m \\ A_m & -A_m \\ \underbrace{\begin{matrix} 1 & \cdots & 1 \\ 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{matrix}}_{2^m} & \underbrace{\begin{matrix} -1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & \cdots & -1 \end{matrix}}_{2^m} \end{bmatrix}$$

- Want to show: v is decodable from $v\tilde{A}_m$ for any $v \in \{\pm 1\}^{\otimes K_m}$ and $v_{2K_{m-1}+1} = 0$.
- Equivalently: $v \in \{0, 1\}^{\otimes K_m}$ (just use $v \mapsto \frac{1+v}{2}$)
- Let $v = [x\, y\, z]$ and

$$[x\, y\, z]\tilde{A}_m = [g\, h] \Rightarrow \quad g - h = [x\, y\, z] \begin{pmatrix} 0 \\ 2A_{m-1} \\ 2I_{2^{m-1}} \end{pmatrix}$$

- $z_1 = 0$, so by adding $(g-h)_1$ to $(g-h)_\ell$ we get:

$$(*) \qquad 2z_\ell = (g-h)_1 + (g-h)_\ell - 2y \cdot v_\ell \qquad \ell = 2, \ldots, 2^{m-1}$$

where v_ℓ is sum of 1-st and ℓ-th column of A_{m-1}

- Key: v_ℓ's entries are $\{0, 2\}$. Take $\mod 4$ of $(*)$ and decode z_ℓ's !
- Subtracting z_ℓ's we get system:

$$[x\, y] \begin{pmatrix} A_{m-1} & A_{m-1} \\ A_{m-1} & -A_{m-1} \end{pmatrix} = [g'\, h']$$

$$\tilde{A}_{m+1} = \begin{array}{c} \\ \\ 2^m \left\{ \begin{array}{c} \\ \\ \\ \\ \end{array} \right. \end{array} \begin{bmatrix} A_m & A_m \\ A_m & -A_m \\ \begin{matrix} 1 & \cdots & 1 \\ 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{matrix} & \begin{matrix} -1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & \cdots & -1 \end{matrix} \\ \underbrace{\hphantom{1 \cdots 1}}_{2^m} & \underbrace{\hphantom{-1\ 1 \cdots 1}}_{2^m} \end{bmatrix} \qquad A_{m+1} = \begin{array}{c} \\ \\ 2^m \left\{ \begin{array}{c} \\ \\ \\ \\ \end{array} \right. \end{array} \begin{bmatrix} A_m & A_m \\ A_m & -A_m \\ \begin{matrix} 1 & \cdots & 1 \\ 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{matrix} & \begin{matrix} -1 & 1 & \cdots & 1 \\ 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & \cdots & -1 \end{matrix} \\ \underbrace{\hphantom{1 \cdots 1}}_{2^m} & \underbrace{\hphantom{-1\ 1 \cdots 1}}_{2^m} \end{bmatrix}$$

- Want to show: v is decodable from $v\tilde{A}_m$ for any $v \in \{\pm 1\}^{\otimes K_m}$ and $v_{2K_{m-1}+1} = 0$.
- Equivalently: $v \in \{0,1\}^{\otimes K_m}$ (just use $v \mapsto \frac{1+v}{2}$)
- Let $v = [x\, y\, z]$ and

$$[x\, y\, z]\tilde{A}_m = [g\, h] \Rightarrow \quad g - h = [x\, y\, z] \begin{pmatrix} 0 \\ 2A_{m-1} \\ 2I_{2^{m-1}} \end{pmatrix}$$

- $z_1 = 0$, so by adding $(g-h)_1$ to $(g-h)_\ell$ we get:

(*) $$2z_\ell = (g-h)_1 + (g-h)_\ell - 2y \cdot v_\ell \qquad \ell = 2, \ldots, 2^{m-1}$$

where v_ℓ is sum of 1-st and ℓ-th column of A_{m-1}
- Key: v_ℓ's entries are $\{0, 2\}$. Take $\mod 4$ of $(*)$ and decode z_ℓ's !
- Subtracting z_ℓ's we get system:

$$[x\, y] \begin{pmatrix} A_{m-1} & A_{m-1} \\ A_{m-1} & -A_{m-1} \end{pmatrix} = [g'\, h'] \quad \Rightarrow xA_{m-1} = \frac{g' + h'}{2}$$

$$\tilde{A}_{m+1} = \begin{bmatrix} A_m & & & A_m & & & \\ A_m & & & -A_m & & & \\ 1 & \cdots & 1 & -1 & 1 & \cdots & 1 \\ 1 & \cdots & 1 & 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & 1 & 1 & \cdots & \cdots & -1 \end{bmatrix} \qquad A_{m+1} = \begin{bmatrix} A_m & & & A_m & & & \\ A_m & & & -A_m & & & \\ 1 & \cdots & 1 & -1 & 1 & \cdots & 1 \\ 1 & \cdots & 1 & 1 & -1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \cdots & 1 & 1 & \cdots & \cdots & -1 \end{bmatrix}$$

- Want to show: v is decodable from $v\tilde{A}_m$ for any $v \in \{\pm 1\}^{\otimes K_m}$ and $v_{2K_{m-1}+1} = 0$.
- Equivalently: $v \in \{0,1\}^{\otimes K_m}$ (just use $v \mapsto \frac{1+v}{2}$)
- Let $v = [x\,y\,z]$ and

$$[x\,y\,z]\tilde{A}_m = [g\,h] \Rightarrow \quad g - h = [x\,y\,z]\begin{pmatrix} 0 \\ 2A_{m-1} \\ 2I_{2^{m-1}} \end{pmatrix}$$

- $z_1 = 0$, so by adding $(g-h)_1$ to $(g-h)_\ell$ we get:

$$(*) \qquad 2z_\ell = (g-h)_1 + (g-h)_\ell - 2y \cdot v_\ell \qquad \ell = 2, \ldots, 2^{m-1}$$

where v_ℓ is sum of 1-st and ℓ-th column of A_{m-1}

- Key: v_ℓ's entries are $\{0,2\}$. Take $\mod 4$ of $(*)$ and decode z_ℓ's !
- Subtracting z_ℓ's we get system:

$$[x\,y]\begin{pmatrix} A_{m-1} & A_{m-1} \\ A_{m-1} & -A_{m-1} \end{pmatrix} = [g'\,h'] \quad \Rightarrow xA_{m-1} = \frac{g'+h'}{2} \quad \Rightarrow \text{induct}$$

Reflections

- When user inputs are constrained (to ± 1), can have $K \gg n$ and still recover inputs.
- Total information grows with K: $H(X_1 + \cdots + X_K) \sim \frac{1}{2} \log K$. (This is similar to $\frac{1}{2} \log(1 + KP)$ in GMAC.)
- Lots of smart ideas in MAC codes.

Reflections

- When user inputs are constrained (to ± 1), can have $K \gg n$ and still recover inputs.
- Total information grows with K: $H(X_1 + \cdots + X_K) \sim \frac{1}{2}\log K$. (This is similar to $\frac{1}{2}\log(1+KP)$ in GMAC.)
- Lots of smart ideas in MAC codes.
- Information theory structures it all into:

$$C = \bigcup_{X_1,\ldots,X_K,U} \{(R_1,\ldots,R_K) : R_S \le I(X_S; Y | X_{S^c}, U)\}$$

Reflections

- When user inputs are constrained (to ± 1), can have $K \gg n$ and still recover inputs.
- Total information grows with K: $H(X_1 + \cdots + X_K) \sim \frac{1}{2}\log K$. (This is similar to $\frac{1}{2}\log(1+KP)$ in GMAC.)
- Lots of smart ideas in MAC codes.
- Information theory structures it all into:

$$C = \bigcup_{X_1,\ldots,X_K,U} \{(R_1,\ldots,R_K) : R_S \le I(X_S; Y | X_{S^c}, U)\}$$

- Similar to how all the smarts (Reed-Muller, BCH, LDPC, Polar, ...) are hidden behind

$$C = \max_X I(X;Y)$$

Reflections

- When user inputs are constrained (to ± 1), can have $K \gg n$ and still recover inputs.
- Total information grows with K: $H(X_1 + \cdots + X_K) \sim \frac{1}{2} \log K$. (This is similar to $\frac{1}{2} \log(1 + KP)$ in GMAC.)
- Lots of smart ideas in MAC codes.
- Information theory structures it all into:

$$C = \bigcup_{X_1,\ldots,X_K,U} \{(R_1, \ldots, R_K) : R_S \le I(X_S; Y | X_{S^c}, U)\}$$

- Similar to how all the smarts (Reed-Muller, BCH, LDPC, Polar, ...) are hidden behind

$$C = \max_X I(X; Y)$$

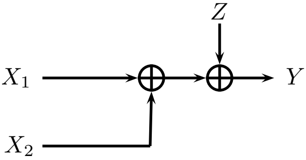
- We understand that "pie-slicing" point of view of radio-MAC is wrong. What is right?

2-user Gaussian MAC

$$Y = X_1 + X_2 + Z$$

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

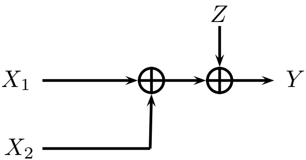
$$\mathbb{E}[(X_1)^2] \le P_1, \mathbb{E}[(X_2)^2] \le P_2$$



$$Y = X_1 + X_2 + Z$$

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

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



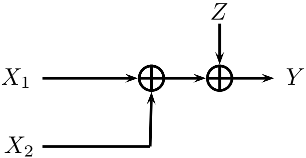
- Evaluating capacity region:

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

$$Y = X_1 + X_2 + Z$$

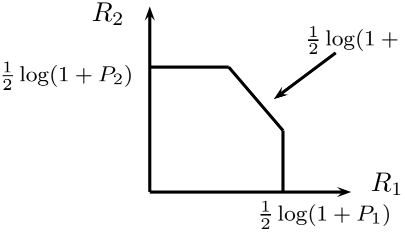
$$Z \overset{iid}{\sim} \mathcal{N}(0,1)$$

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



- Evaluating capacity region:

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

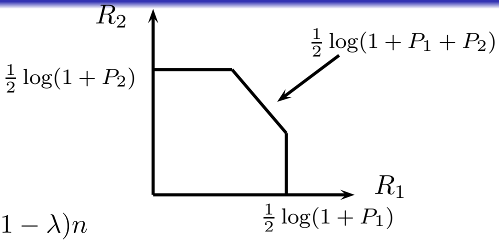


2-GMAC rates for TDMA

$$Y = X_1 + X_2 + Z$$

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

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



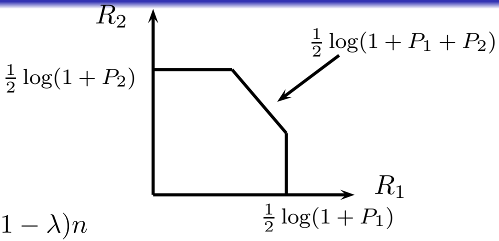
- Here is a TDMA:
  - Partition block: $n = \lambda n + (1-\lambda)n$
  - User 1 sends in λn: $R_1 = \frac{\lambda}{2}\log(1+P_1)$
  - User 2 sends in $\bar{\lambda} n$: $R_2 = \frac{\bar{\lambda}}{2}\log(1+P_2)$

2-GMAC rates for TDMA

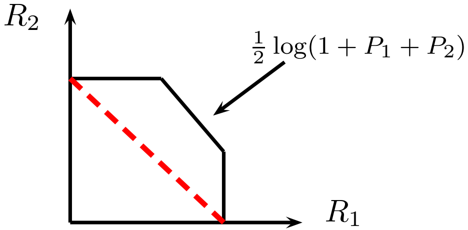
$$Y = X_1 + X_2 + Z$$

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

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



- Here is a TDMA:
  - Partition block: $n = \lambda n + (1-\lambda)n$
  - User 1 sends in λn: $R_1 = \frac{\lambda}{2}\log(1+P_1)$
  - User 2 sends in $\bar{\lambda} n$: $R_2 = \frac{\bar{\lambda}}{2}\log(1+P_2)$

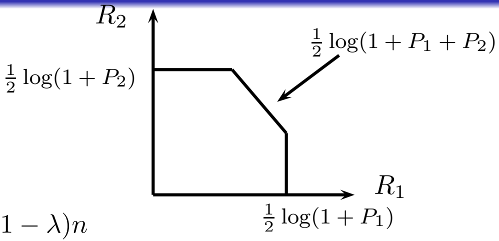


2-GMAC rates for TDMA

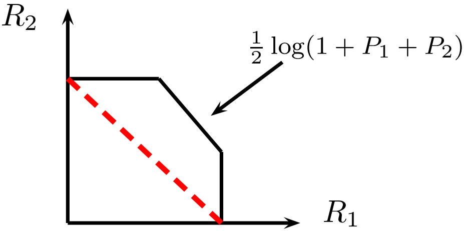
$$Y = X_1 + X_2 + Z$$

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

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



- Here is a TDMA:
  - Partition block: $n = \lambda n + (1-\lambda)n$
  - User 1 sends in λn: $R_1 = \frac{\lambda}{2}\log(1+P_1)$
  - User 2 sends in $\bar{\lambda} n$: $R_2 = \frac{\bar{\lambda}}{2}\log(1+P_2)$



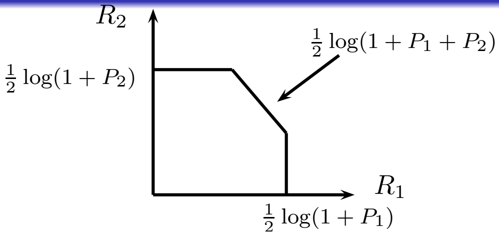
- Note: low-complexity decoder – two users are decoded separately.

2-GMAC rates for FDMA

$$Y = X_1 + X_2 + Z$$

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

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



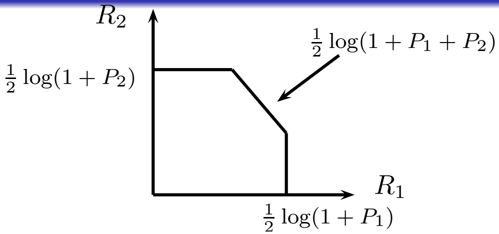
- Here is a FDMA:
  - Use Fourier transform to change n=time to n=frequency.
  - Partition block: $n = \lambda n + (1-\lambda)n$
  - User 1 sends in λn: $\boxed{R_1 = \frac{\lambda}{2}\log(1+\frac{P_1}{\lambda})}$
  - User 2 sends in $\bar{\lambda} n$: $\boxed{R_2 = \frac{\bar{\lambda}}{2}\log(1+\frac{P_2}{\bar{\lambda}})}$

2-GMAC rates for FDMA

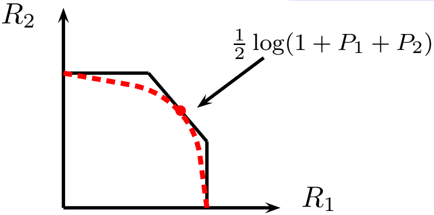
$$Y = X_1 + X_2 + Z$$

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

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



- 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 $\bar{\lambda} n$: $R_2 = \frac{\bar{\lambda}}{2}\log(1+\frac{P_2}{\bar{\lambda}})$

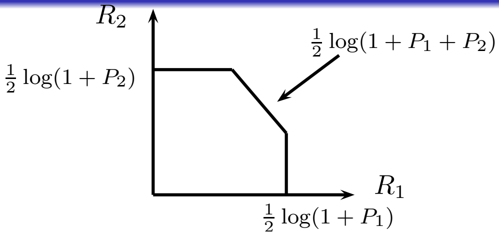


2-GMAC rates for FDMA

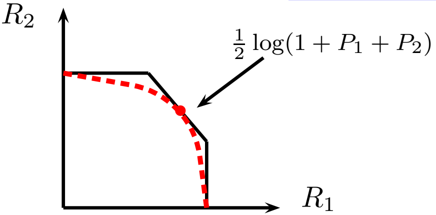
$$Y = X_1 + X_2 + Z$$

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

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



- 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 $\bar{\lambda} n$: $R_2 = \frac{\bar{\lambda}}{2}\log(1+\frac{P_2}{\bar{\lambda}})$



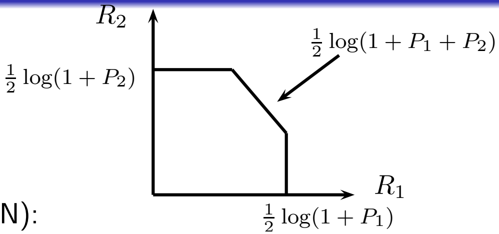
$\lambda^* = \frac{P_1}{P_1+P_2}$ achieves optimal sumrate

2-GMAC rates for TIN

$$Y = X_1 + X_2 + Z$$

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

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



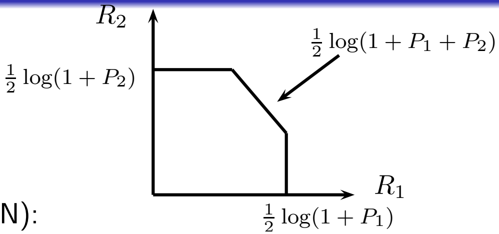
- Treat-interference-as-noise (TIN):
  - Each user treats the other as noise (single-user decoders)
  - Random coding ensures noise is Gaussian.
  - Rates: $\boxed{R_1 = \frac{1}{2}\log(1+\frac{P_1}{1+P_2}), R_2 = \frac{1}{2}\log(1+\frac{P_2}{1+P_1})}$

2-GMAC rates for TIN

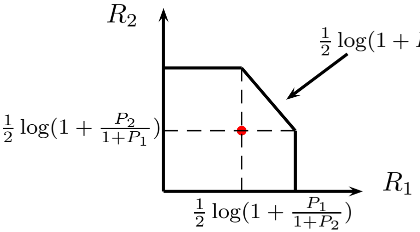
$$Y = X_1 + X_2 + Z$$

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

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



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



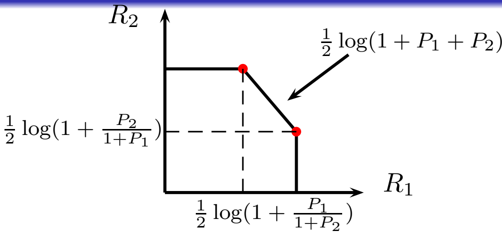
- TIN point can be inside/outside TDMA.

TIN + SIC

$$Y = X_1 + X_2 + Z$$

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

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



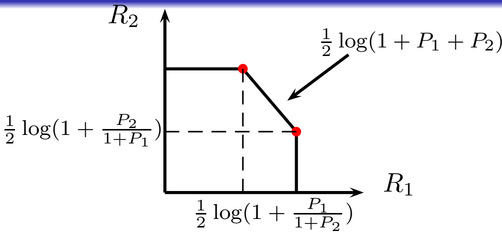
- Consider a corner point:

$$R_1 = \frac{1}{2}\log(1+\frac{P_1}{1+P_2}), \qquad R_2 = \frac{1}{2}\log(1+P_2)\,.$$

$$Y = X_1 + X_2 + Z$$

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

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



- Consider a corner point:

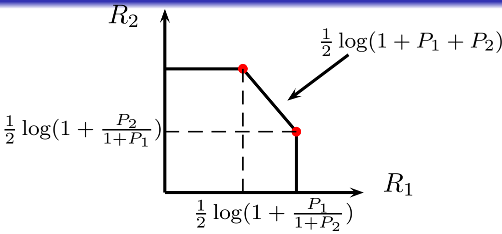
$$R_1 = \frac{1}{2}\log(1+\frac{P_1}{1+P_2}), \qquad R_2 = \frac{1}{2}\log(1+P_2)\,.$$

- User 1 can be decoded by TIN. But then can subtract it out!

$$Y = X_1 + X_2 + Z$$

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

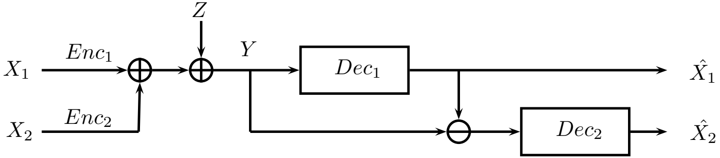
$$\mathbb{E}[(X_1)^2] \le P_1, \mathbb{E}[(X_2)^2] \le P_2$$



- Consider a corner point:

$$R_1 = \frac{1}{2}\log(1+\frac{P_1}{1+P_2}), \qquad R_2 = \frac{1}{2}\log(1+P_2)\,.$$

- User 1 can be decoded by TIN. But then can subtract it out!

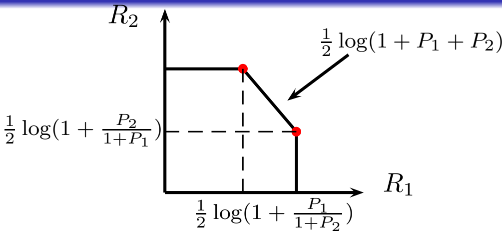


TIN + SIC

$$Y = X_1 + X_2 + Z$$

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

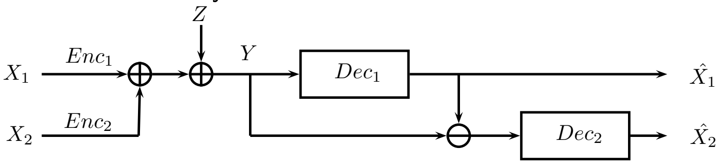
$$\mathbb{E}[(X_1)^2] \le P_1, \mathbb{E}[(X_2)^2] \le P_2$$



- Consider a corner point:

$$R_1 = \frac{1}{2}\log(1+\frac{P_1}{1+P_2}), \qquad R_2 = \frac{1}{2}\log(1+P_2)\,.$$

- User 1 can be decoded by TIN. But then can subtract it out!



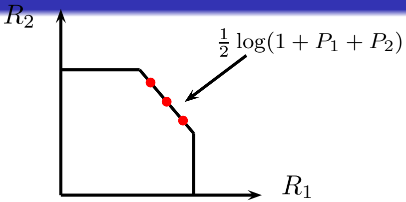
- So far: achieved three optimal points via SU-decoding. Any more?

Rate-splitting

$$Y = X_1 + X_2 + Z$$

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

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



- Split user 1 into two virtual users $1A$ and $1B$:

$$R_1 = R_{1A} + R_{1B}, \quad P_1 = P_{1A} + P_{1B}$$

- A funny order of decoding:
  - Decode X_{1A} via TIN: $R_{1A} = \frac{1}{2}\log(1 + \frac{P_{1A}}{1+P_{1B}+P_2})$
  - Subtract X_{1A}, decode X_2: $R_2 = \frac{1}{2}\log(1 + \frac{P_2}{1+P_{1B}})$
  - Subtract X_2, decode X_{1B}: $R_{1B} = \frac{1}{2}\log(1 + P_{1B})$
- Simple check:

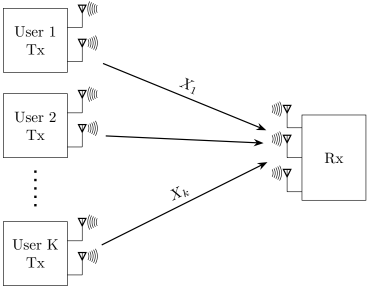
$$R_{1A} + R_{1B} + R_2 = \frac{1}{2}\log(1+P_1+P_2)$$

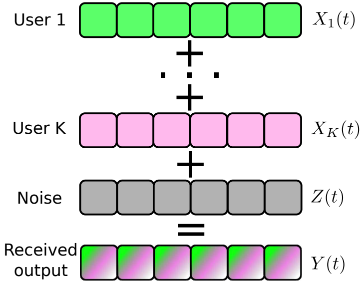
sumrate optimal

by varying $P_{1A} + P_{1B} = P_1$ can achieve any point!!

K-user GMAC

[t]





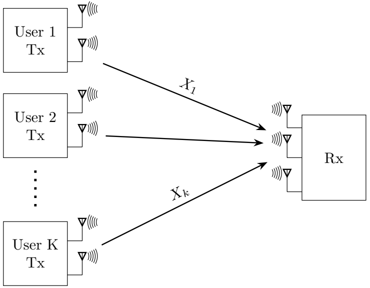
$$Y(t) = X_1(t) + \cdots + X_K(t) + Z(t)$$

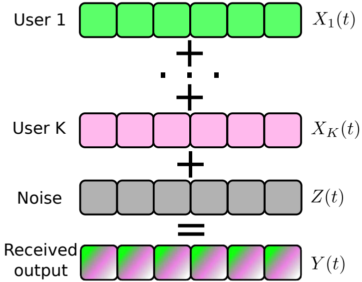
- Assume equal-power setting $P_i = P$. Capacity region (sumrate):

$$\sum_{i=1}^{K} R_i \leq \frac{1}{2}\log(1 + KP)$$

K-user GMAC

[t]





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

- single-user decoders achieve:
  - FDMA optimal at symmetric point: $R_i = \frac{1}{2K}\log(1 + KP)$
  - TIN+SIC achieves all vertices.
  - Rate-Splitting all points of optimal sumrate.
- Is that it? Let us see...

K-user GMAC: Reflections

- So total capacity:

$$C_{sum} = \frac{1}{2}\log_2(1 + KP) \qquad bit/rdof$$

growing to ∞ as $K \to \infty$.

K-user GMAC: Reflections

- So total capacity:
$$C_{sum} = \frac{1}{2}\log_2(1+KP) \qquad \color{red}{bit/rdof}$$
growing to ∞ as $K \to \infty$.
- But at the same time, per-user rate:
$$C_{sym} = \frac{1}{2K}\log_2(1+KP) \to 0\,.$$
- The crucial performance metric: HRH **energy-per-bit**
$$\frac{E_b}{N_0} \triangleq \frac{\text{total energy spent}}{2 \times \text{total \# bits}} = \frac{nKP}{2nC_{sum}}$$

K-user GMAC: Reflections

- So total capacity:

$$C_{sum} = \frac{1}{2}\log_2(1+KP) \qquad bit/rdof$$

growing to ∞ as $K \to \infty$.

- But at the same time, per-user rate:

$$C_{sym} = \frac{1}{2K}\log_2(1+KP) \to 0\,.$$

- The crucial performance metric: HRH **energy-per-bit**

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

- As $K \to \infty$:

$$\frac{E_b}{N_0} = \frac{KP}{\log(1+KP)} \to \infty \qquad !!!$$

- Capacity $\nearrow$, but each user works harder and moves fewer bits/sec!

K-user GMAC: Reflections

- So total capacity:
$$C_{sum} = \frac{1}{2}\log_2(1+KP) \qquad bit/rdof$$
growing to ∞ as $K \to \infty$.
- But at the same time, per-user rate:
$$C_{sym} = \frac{1}{2K}\log_2(1+KP) \to 0\,.$$
- The crucial performance metric: HRH **energy-per-bit**
$$\frac{E_b}{N_0} \triangleq \frac{\text{total energy spent}}{2\times \text{total \# bits}} = \frac{nKP}{2nC_{sum}}$$
- As $K \to \infty$:
$$\frac{E_b}{N_0} = \frac{KP}{\log(1+KP)} \to \infty \qquad !!!$$
- Capacity $\nearrow$, but each user works harder and moves fewer bits/sec!
- Correct scaling: $P_{tot} = KP$ should be fixed!

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

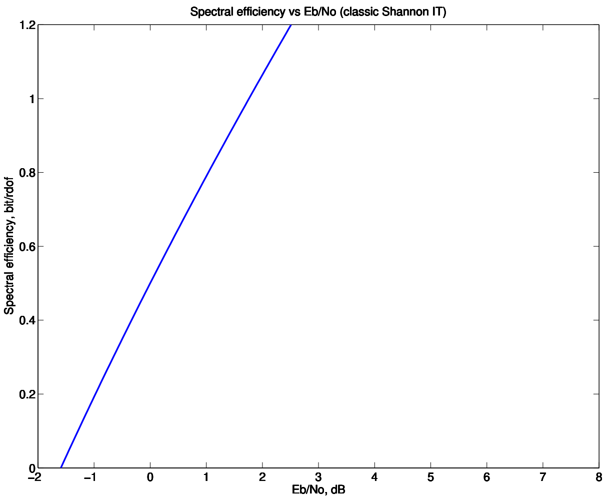
- Studying this tradeoff is the favorite pastime of ComSoc
- Sp.eff. $\rho \triangleq \frac{\text{total \# of data bits}}{\text{total real d.o.f.}}$
- We have:

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

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

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

- Stud
- Sp.e
- We l
- rega



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

- Studying this tradeoff is the favorite pastime of ComSoc
- Sp.eff. $\rho \triangleq \dfrac{\text{total \# of data bits}}{\text{total real d.o.f.}}$
- We have:

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

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

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

- 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. $\frac{E_b}{N_0}$

- Studying this tradeoff is the favorite pastime of ComSoc
- Sp.eff. $\rho \triangleq \frac{\text{total \# of data bits}}{\text{total real d.o.f.}}$
- We have:

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

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

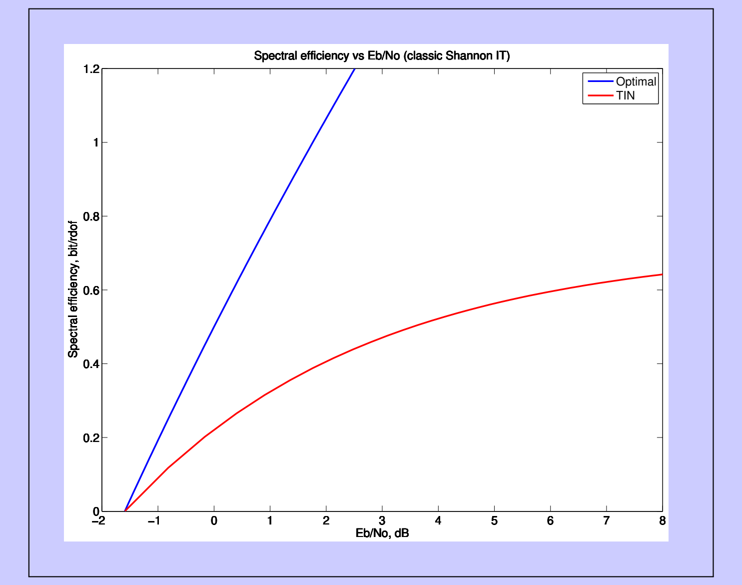
$$\frac{E_b}{N_0} = \frac{2^{2\rho}-1}{2\rho} \ge -1.59\ dB$$

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

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

Spectral efficiency vs. $\frac{E_b}{No}$

- Stud
- Sp.e
- We l
- rega
- Com



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

- Studying this tradeoff is the favorite pastime of ComSoc
- Sp.eff. $\rho \triangleq \frac{\text{total \# of data bits}}{\text{total real d.o.f.}}$
- We have:

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

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

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

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

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

- **IMPORTANT:** $\rho \le \frac{1}{2\ln 2} = 0.72$ bit/rdof
- **IMPORTANT:** Essentially optimal for low sp.eff.

Modulation

- Given that TIN is not bad for low sp.eff., let us try to achieve it.
- **Problem:** Per-user rate $= \frac{\rho}{K}$ and is very small for large K.

Modulation

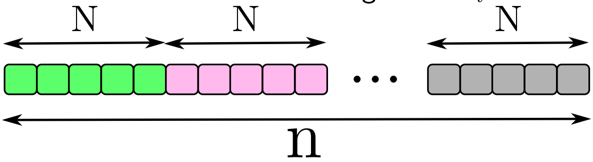
- Given that TIN is not bad for low sp.eff., let us try to achieve it.
- **Problem:** Per-user rate $= \frac{\rho}{K}$ and is very small for large K. Aside:
  - For IT Soc: Channel with $C = 0.5$ and channel with $C = 0.001$ are not fundamentally different.
  - For ComSoc: First channel is OK (turbo/LDPC/polar), second is a nightmare.
  - Why? First, SNR needs to be brought up to a reasonable level.
  - This is the idea of modulation.

Modulation

- Given that TIN is not bad for low sp.eff., let us try to achieve it.
- **Problem:** Per-user rate $= \frac{\rho}{K}$ and is very small for large K. Aside:
  - For IT Soc: Channel with $C = 0.5$ and channel with $C = 0.001$ are not fundamentally different.
  - For ComSoc: First channel is OK (turbo/LDPC/polar), second is a nightmare.
  - Why? First, SNR needs to be brought up to a reasonable level.
  - This is the idea of modulation.
  - Another issue: how do you do TIN practically? A code with ± 1 entries will create a very non-Gaussian interference!

Modulation

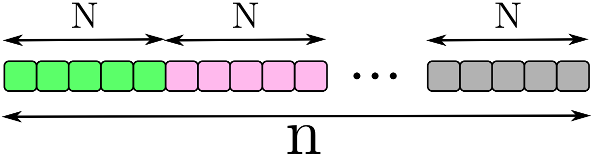
- Given that TIN is not bad for low sp.eff., let us try to achieve it.
- **Problem:** Per-user rate $= \frac{\rho}{K}$ and is very small for large K.
- **Solution:** each user modulates some N-signature $s_i \in \mathbb{R}^N$



- Think of N-blocks as new super-symbols. Effective channel:

$$Y^N = s_1 B_1 + s_2 B_2 + \cdots s_K B_k + Z^N \,, \qquad \|s_i\| = 1$$

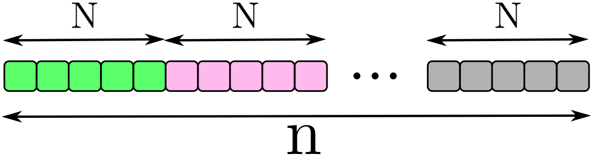
  - Set $\beta = \frac{K}{N}$
  - new power-constraint: $\mathbb{E}[B_i^2] \le NP = \frac{P_{tot}}{\beta}$.
  - new rate: $\frac{\rho N}{K} = \frac{\rho}{\beta}$ in bits / one B-symbol.
  - with proper choice should have $\frac{\rho}{\beta} \sim 1$ as ComSoc likes.



- N-blocks are new super-symbols. Effective channel:

$$Y^N = s_1 B_1 + s_2 B_2 + \cdots s_K B_k + Z^N , \qquad \|s_i\| = 1$$

  - Set $\beta = \frac{K}{N}$
  - new power-constraint: $\mathbb{E}[B_i^2] \le NP = \frac{P_{tot}}{\beta}$.
- Side observation:
  - If s_i's are chosen orthogonally and $K = N$, this is FDMA (hence optimal).
  - But incurs FBL loss – important when $K \sim n$. Ignore for now.
  - So why not do so? Many reasons:
    - K may vary, but N should be constant.
    - Requires central distribution of signatures among ACTIVE users.
    - Asynchrony kills orthogonality
  - Early Qualcomm: random-like s_i's resolve all issues, and are good enough for TIN !



- N-blocks are new super-symbols. Effective channel:

$$Y^N = s_1 B_1 + s_2 B_2 + \cdots s_K B_k + Z^N\,, \qquad \|s_i\| = 1$$

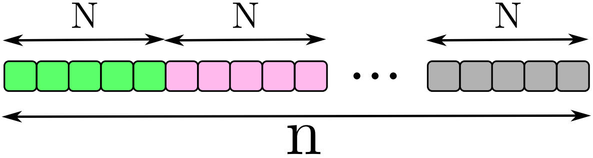
  - Set $\beta = \frac{K}{N}$
  - new power-constraint: $\mathbb{E}[B_i^2] \le NP = \frac{P_{tot}}{\beta}$.
- Idea 1: Decode via matched-filter + SU decoders:

$$\hat{B}_i = \langle s_i, Y^N \rangle = B_i + \tilde{Z}_i, \qquad \mathrm{Var}[\tilde{Z}_i] = 1 + NP \sum\nolimits_{j \neq i} |\langle s_i, s_j \rangle|^2$$

- Idea 2: Select s_i randomly. (attractive sys. arch.)
- When s_i's are random and N large:

$$|\langle s_i, s_j \rangle| \approx \frac{1}{\sqrt{N}} \qquad \text{w.h.p.}$$

- So SU-decoder sees effective SNR $= \frac{NP}{1+(K-1)P} = \frac{P_{tot}}{1+P_{tot}} \frac{1}{\beta}$



- N-blocks are new super-symbols. Effective channel:

$$Y^N = s_1 B_1 + s_2 B_2 + \cdots s_K B_k + Z^N\,, \qquad \|s_i\| = 1$$

  - Set $\beta = \frac{K}{N}$
  - new power-constraint: $\mathbb{E}[B_i^2] \le NP = \frac{P_{tot}}{\beta}$.
  - random (non-orthogonal) signatures
  - matched-filter + SU-decoder

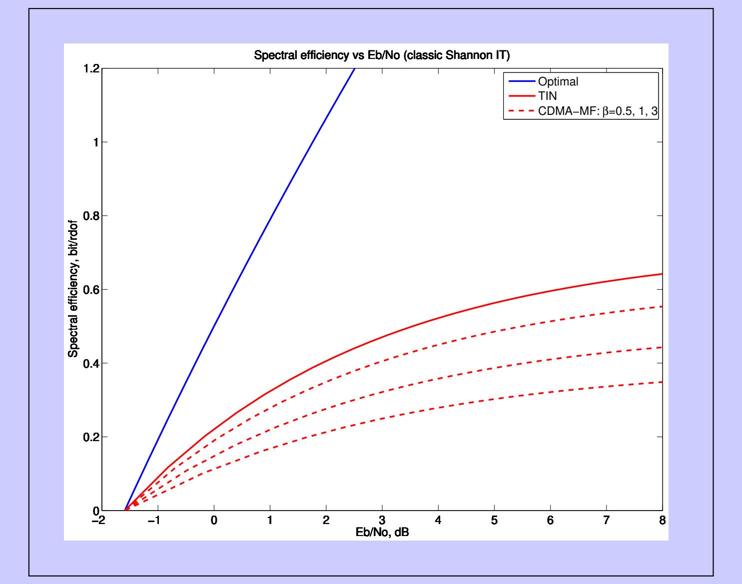
- End result:

$$\rho_{CDMA} = \frac{\beta}{2} \log_2(1 + \frac{P_{tot}}{1+P_{tot}} \frac{1}{\beta}) \qquad \frac{E_b}{N_0} = \frac{P_{tot}}{2\rho_{CDMA}}$$

  - As $\beta \to \infty$ we approach TIN.
  - So classical CDMA folks (Viterbi...) were only trying to achieve TIN.

N N N

- N-b



- End

  - As $\beta \to \infty$ we approach TIN.
  - So classical CDMA folks (Viterbi...) were only trying to achieve TIN.

CDMA: going beyond TIN

- Set $\beta = \frac{K}{N}$
- new power-constraint: $\mathbb{E}[B_i^2] \le NP = \frac{P_{tot}}{\beta}$.
- random (non-orthogonal) signatures
- matched-filter + SU-decoder

$$\rho_{CDMA} = \frac{\beta}{2}\log_2\left(1 + \frac{P_{tot}}{1+P_{tot}}\frac{1}{\beta}\right) \qquad \frac{E_b}{N_0} = \frac{P_{tot}}{2\rho_{CDMA}}$$

- So far we considered matched-filter arch.:

$$\begin{aligned} \hat{B}_1 &= \langle s_1, Y^N \rangle \\ &\cdots \\ \hat{B}_K &= \langle s_K, Y^N \rangle \end{aligned}$$

- Can we do better?

CDMA: going beyond TIN

- Set $\beta = \frac{K}{N}$
- new power-constraint: $\mathbb{E}[B_i^2] \le NP = \frac{P_{tot}}{\beta}$.
- random (non-orthogonal) signatures
- matched-filter + SU-decoder

$$\rho_{CDMA} = \frac{\beta}{2}\log_2(1 + \frac{P_{tot}}{1+P_{tot}}\frac{1}{\beta}) \qquad \frac{E_b}{N_0} = \frac{P_{tot}}{2\rho_{CDMA}}$$

- So far we considered matched-filter arch.:

$$\begin{aligned} \hat{B}_1 &= \langle s_1, Y^N \rangle \\ &\cdots \\ \hat{B}_K &= \langle s_K, Y^N \rangle \end{aligned}$$

- Can we do better? Yes! via multi-user detection (MUD).
- In one of two ways:
  - Signal-processing: Estimate $\hat{B}^K$ via MMSE or decorrelator. Note: does not leverage knowledge of distribution of B_i
  - Coding: Use joint-decoding of $\hat{B}^K$ also leveraging knowledge that (e.g.) $B_i = \pm 1$

CDMA+MUD vs OFDM

- Set $\beta = \frac{K}{N}$
- new power-constraint: $\mathbb{E}[B_i^2] \le NP = \frac{P_{tot}}{\beta}$.
- random (non-orthogonal) signatures
- matched-filter + SU-decoder

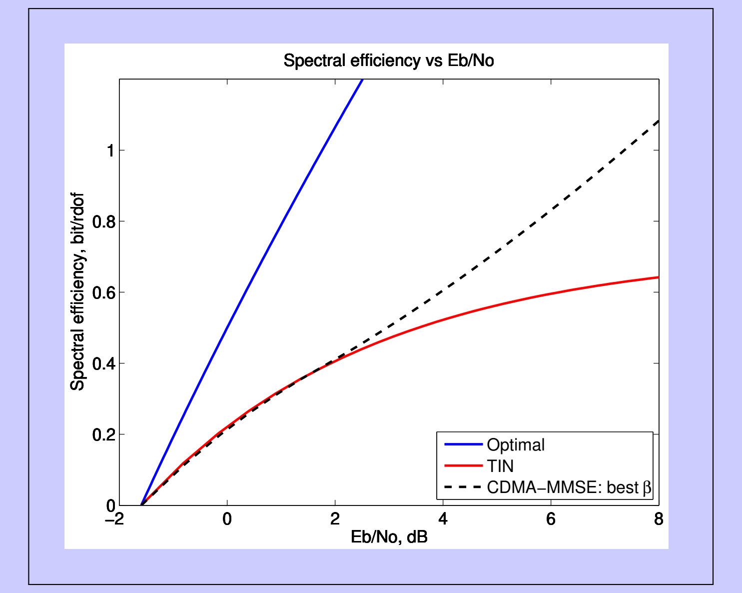
$$\rho_{CDMA} = \frac{\beta}{2}\log_2(1 + \frac{P_{tot}}{1+P_{tot}}\frac{1}{\beta}) \qquad \frac{E_b}{N_0} = \frac{P_{tot}}{2\rho_{CDMA}}$$

- multi-user detectors (MUD) improve performance of random-CDMA.
- E.g. MMSE detector yields (Tse-Hanly/Verdú-Shamai formula)

$$\rho_{MMSE} = \frac{\beta}{2}\log_2(1 + P_1 - \frac{1}{4}\mathcal{F}), \qquad P_1 = \frac{P_{tot}}{\beta}$$

where $\mathcal{F} = (\sqrt{P_1(1+\sqrt{\beta})^2+1} - \sqrt{P_1(1-\sqrt{\beta})^2+1})^2$

CDMA+MUD vs OFDM



-
-
-
-
- -CDMA.
- a)

CDMA+MUD vs OFDM

- Set $\beta = \frac{K}{N}$
- new power-constraint: $\mathbb{E}[B_i^2] \le NP = \frac{P_{tot}}{\beta}$.
- random (non-orthogonal) signatures
- matched-filter + SU-decoder

$$\rho_{CDMA} = \frac{\beta}{2}\log_2(1 + \frac{P_{tot}}{1+P_{tot}}\frac{1}{\beta}) \qquad \frac{E_b}{N_0} = \frac{P_{tot}}{2\rho_{CDMA}}$$

- multi-user detectors (MUD) improve performance of random-CDMA.
- E.g. MMSE detector yields (Tse-Hanly/Verdú-Shamai formula)

$$\rho_{MMSE} = \frac{\beta}{2}\log_2(1 + P_1 - \frac{1}{4}\mathcal{F}), \qquad P_1 = \frac{P_{tot}}{\beta}$$

where $\mathcal{F} = (\sqrt{P_1(1+\sqrt{\beta})^2+1} - \sqrt{P_1(1-\sqrt{\beta})^2+1})^2$

- Allows to beat TIN's $\rho \le 0.72$ bit/rdof bottleneck.
- Still, industry converged to OFDM: spectrum is too precious.

CDMA+MUD vs OFDM

- Set $\beta = \frac{K}{N}$
- new power-constraint: $\mathbb{E}[B_i^2] \le NP = \frac{P_{tot}}{\beta}$.
- random (non-orthogonal) signatures
- matched-filter + SU-decoder

$$\rho_{CDMA} = \frac{\beta}{2}\log_2(1 + \frac{P_{tot}}{1+P_{tot}}\frac{1}{\beta}) \qquad \frac{E_b}{N_0} = \frac{P_{tot}}{2\rho_{CDMA}}$$

- multi-user detectors (MUD) improve performance of random-CDMA.
- E.g. MMSE detector yields (Tse-Hanly/Verdú-Shamai formula)

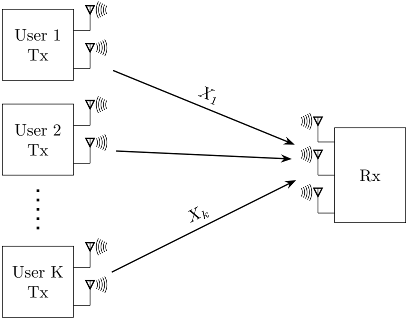
$$\rho_{MMSE} = \frac{\beta}{2}\log_2(1 + P_1 - \frac{1}{4}\mathcal{F}), \qquad P_1 = \frac{P_{tot}}{\beta}$$

where $\mathcal{F} = (\sqrt{P_1(1+\sqrt{\beta})^2+1} - \sqrt{P_1(1-\sqrt{\beta})^2+1})^2$

- Allows to beat TIN's $\rho \le 0.72$ bit/rdof bottleneck.
- Still, industry converged to OFDM : spectrum is too precious.
- IoT: centralized orthogonalization impossible! Comeback of MUD?

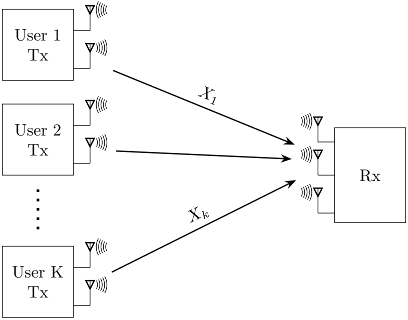
New problems: many users with short packets

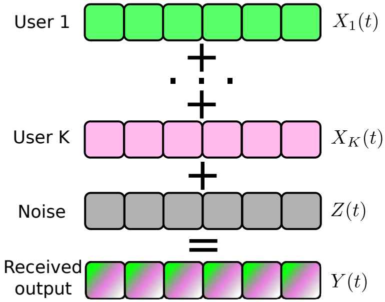
The classical model: K-user multiple-access channel



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

The classical model: K-user multiple-access channel





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

- Before: Fix K, let $n \to \infty$. Few users. Large payloads.
- **Now:** Huge K. Small payload.
- Random-access: User activity – random, uncoordinated

On number of sensors (user density)

- **Key metric:** μ in users/rdof

$$\mu = \frac{\#\text{ of active users per frame}}{\text{size of frame}}$$

- K_{tot} sensors sending with period T_{per} (sec) in band B (Hz)

$$\boxed{\mu = \frac{K_{tot}}{2BT_{per}}}$$

- Futuristic example:
  - City of 10^6.
  - Each house has 10^2 devices.
  - Each dev sends every 10 min, $T_{per} = 600$ s.
  - sub-GHz bandwidth is scarce: ISM $B = 20$ MHz.
  - $\mu \approx 4 \cdot 10^{-3}$.

On number of sensors (user density)

- **Key metric:** μ in users/rdof

$$\mu = \frac{\text{\# of active users per frame}}{\text{size of frame}}$$

- K_{tot} sensors sending with period T_{per} (sec) in band B (Hz)

$$\boxed{\mu = \frac{K_{tot}}{2BT_{per}}}$$

- Futuristic example:
  - City of 10^6.
  - Each house has 10^2 devices.
  - Each dev sends every 10 min, $T_{per} = 600$ s.
  - sub-GHz bandwidth is scarce: ISM $B = 20$ MHz.
  - $\mu \approx 4 \cdot 10^{-3}$.
- Another point of view:
  - Traditional comm: focus on sp.eff. ρ vs $\frac{E_b}{N_0}$. Why?
  - $\frac{\rho B}{K}$ = per-user speed?
  - or is it $\frac{\rho B}{\text{speed}}$ = number of happy users?

New twists compared to classic MAC

Problem 1 large $K \to \infty$, fixed payload $\log_2 M$

Relevant asymptotics: $K, n \to \infty$ with $\frac{K}{n} = \mu$.

Problem 2 "user-centric" probability of error

$$P_e \triangleq \frac{1}{K} \sum_j \mathbb{P}[\hat{X}_j \neq X_j]$$

Problem 3 "random-access"

indistiguishable users (same-codebook), non-asymptotics.

Recap: MAC setting and performance metrics

- Perfectly synchronized K-user Gaussian MAC with blocklength n
- Each user transmits $\log_2 M \approx 10^2$ bits.
- Figures of merit: energy-per-bit and user density

$$\frac{E_b}{N_0} \triangleq \frac{\mathbb{E}[\|X^n\|^2]}{2\log_2 M} \qquad \mu \triangleq \frac{K}{n}$$

Fundamental limits: for K-user MAC with $K \sim n$

Recap: MAC setting and performance metrics

- Perfectly synchronized K-user Gaussian MAC with blocklength n
- Each user transmits $\log_2 M \approx 10^2$ bits.
- Figures of merit: energy-per-bit and user density

$$\frac{E_b}{N_0} \triangleq \frac{\mathbb{E}[\|X^n\|^2]}{2\log_2 M} \qquad \mu \triangleq \frac{K}{n}$$

Problem 1: "massive" number of users

- Number of users $K = \mu n$ scales linearly with blocklength!
- Q: Why scale linearly? A: $\#$ of devices waking up $\asymp \mathrm{time}$.
- Q: Ok, but what μ should we look at?

Fundamental limits: for K-user MAC with $K \sim n$

Recap: MAC setting and performance metrics

- Perfectly synchronized K-user Gaussian MAC with blocklength n
- Each user transmits $\log_2 M \approx 10^2$ bits.
- Figures of merit: energy-per-bit and user density

$$\frac{E_b}{N_0} \triangleq \frac{\mathbb{E}[\|X^n\|^2]}{2\log_2 M} \qquad \mu \triangleq \frac{K}{n}$$

Problem 1: "massive" number of users

- Number of users $K = \mu n$ scales linearly with blocklength!
- Q: Why scale linearly? A: # of devices waking up $\asymp \mathrm{time}$.
- Q: Ok, but what μ should we look at?
  A: $\mu \sim 10^{-3}$. Here is why:
  - City of 10^6.
  - Each house has 10^2 devices.
  - Each dev sends 1-10 times/hour.
  - sub-GHz bandwidth is scarce, unlikely to ever get > 20 MHz.
  - $\Rightarrow \frac{K}{n} \approx 10^{-3} \dots 10^{-2}$. This relation is unlikely to change soon.

Fundamental limits: for K-user MAC with $K \sim n$

Recap: MAC setting and performance metrics

- Perfectly synchronized K-user Gaussian MAC with blocklength n
- Each user transmits $\log_2 M$ bits.
- Figures of merit: energy-per-bit and user density

$$\frac{E_b}{N_0} \triangleq \frac{\mathbb{E}[\|X^n\|^2]}{2\log_2 M} \qquad \mu \triangleq \frac{K}{n}$$

Problem 1: "massive" number of users

- Number of users $K = \mu n$ scales linearly with blocklength!
- [Chen-Chen-Guo'17]: Fix per-user power to P (i.e. codeword $\|c\|_2^2 \le nP$), then

$$\log M^*_{user}(K = \mu n, n, P) \approx \frac{1}{2\mu} \log(1 + \mu n P)$$

- Note: this corresponds to $\frac{E_b}{N_0} \to \infty$.
- Our work: What about finite $\frac{E_b}{N_0}$?

New twists compared to classic MAC

Problem 1 large $K \to \infty$, fixed payload $\log_2 M$

Relevant asymptotics: $K, n \to \infty$ with $\frac{K}{n} = \mu$.

Problem 2 "user-centric" probability of error

$$P_e \triangleq \frac{1}{K} \sum_j \mathbb{P}[\hat{X}_j \neq X_j]$$

Problem 3 "random-access"

indistiguishable users (same-codebook), non-asymptotics.

Fundamental limits: for K-user MAC with $K \sim n$

Recap: MAC setting and performance metrics

- Perfectly synchronized K-user Gaussian MAC with blocklength n
- Each user transmits $\log_2 M$ bits.
- Figures of merit: energy-per-bit and user density

$$\frac{E_b}{N_0} \triangleq \frac{\mathbb{E}[\|X^n\|^2]}{2\log_2 M} \qquad \mu \triangleq \frac{K}{n}$$

- Regime: $K = \mu n$, $n \to \infty$.

Problem 2: "user-centric" prob. of error

- For finite $\frac{E_b}{N_0}$ we have (Why? See next...)

$$\mathbb{P}[W_1 = \hat{W}_1, \dots W_K = \hat{W}_K] \to 0 \qquad \text{as } n \to \infty$$

- $\Rightarrow$ NEED to switch to per-user P_e, PUPE:

$$P_e = \frac{1}{K}\sum_{i=1}^{K} \mathbb{P}[W_i \neq \hat{W}_i]$$

$Eb/N_0 \to \infty$ for classical probability of error

Theorem

Suppose K users send one bit each with finite energy $\mathcal{E}$ over the GMAC (with arbitrary n): $Y^n = \sum_{i=1}^K X_i + Z^n$. Then we have

$$\mathbb{P}[X_1 = \hat{X}_1, \ldots, X_K = \hat{X}_K] \le \frac{\mathcal{E}\frac{\log e}{2} + \log 2}{\log K}\,.$$

And, thus, classical probability of error $\to 1$ as $K \to \infty$.

$Eb/N_0 \to \infty$ for classical probability of error

Theorem

Suppose K users send one bit each with finite energy $\mathcal{E}$ over the GMAC (with arbitrary n): $Y^n = \sum_{i=1}^K X_i + Z^n$. Then we have

$$\mathbb{P}[X_1 = \hat{X}_1, \ldots, X_K = \hat{X}_K] \le \frac{\mathcal{E}\frac{\log e}{2} + \log 2}{\log K}\,.$$

And, thus, classical probability of error $\to 1$ as $K \to \infty$.

Proof:

- WLOG can assume: $Y = \sum c_i W_i + Z$, where $c_i \in \mathbb{R}^n$ and $W_i \sim \mathrm{Ber}(1/2)$.
- Genie: Reveal vector of W_i's to within Hamming-distance 1.
- New problem: See $Y = c_U + Z$, $U \sim [K]$. Goal: find U.

$Eb/N_0 \to \infty$ for classical probability of error

Theorem

Suppose K users send one bit each with finite energy $\mathcal{E}$ over the GMAC (with arbitrary n): $Y^n = \sum_{i=1}^K X_i + Z^n$. Then we have

$$\mathbb{P}[X_1 = \hat{X}_1, \ldots, X_K = \hat{X}_K] \le \frac{\mathcal{E}\frac{\log e}{2} + \log 2}{\log K}\,.$$

And, thus, classical probability of error $\to 1$ as $K \to \infty$.

Proof:

- WLOG can assume: $Y = \sum c_i W_i + Z$, where $c_i \in \mathbb{R}^n$ and $W_i \sim \mathrm{Ber}(1/2)$.
- Genie: Reveal vector of W_i's to within Hamming-distance 1.
- New problem: See $Y = c_U + Z$, $U \sim [K]$. Goal: find U.
- Fano + Capacity calculation:

$$\mathbb{P}[U = \hat{U}] \log K - \log 2 \le I(c_U; Y)$$

$Eb/N_0 \to \infty$ for classical probability of error

Theorem

Suppose K users send one bit each with finite energy $\mathcal{E}$ over the GMAC (with arbitrary n): $Y^n = \sum_{i=1}^K X_i + Z^n$. Then we have

$$\mathbb{P}[X_1 = \hat{X}_1, \ldots, X_K = \hat{X}_K] \le \frac{\mathcal{E}\frac{\log e}{2} + \log 2}{\log K}\,.$$

And, thus, classical probability of error $\to 1$ as $K \to \infty$.

Proof:

- WLOG can assume: $Y = \sum c_i W_i + Z$, where $c_i \in \mathbb{R}^n$ and $W_i \sim \mathrm{Ber}(1/2)$.
- Genie: Reveal vector of W_i's to within Hamming-distance 1.
- New problem: See $Y = c_U + Z$, $U \sim [K]$. Goal: find U.
- Fano + Capacity calculation:

$$\mathbb{P}[U = \hat{U}] \log K - \log 2 \le I(c_U; Y) \le \frac{n}{2}\log\left(1 + \frac{\mathcal{E}}{n}\right) \le \frac{\log e}{2}\mathcal{E}$$

$Eb/N_0 \to \infty$ for classical probability of error

Theorem (AWGN)

Suppose K users send one bit each with finite energy $\mathcal{E}$ over the GMAC (with arbitrary n): $Y^n = \sum_{i=1}^K X_i + Z^n$. Then we have

$$\mathbb{P}[X_1 = \hat{X}_1, \ldots, X_K = \hat{X}_K] \le \frac{\mathcal{E}\frac{\log e}{2} + \log 2}{\log K}\,.$$

Same proof:

Theorem (BSC)

Let G be a $K \times n$ generating matrix with $\le \mathcal{E}$ ones per row. Then over $BSC(\delta)$ and all n:

$$1 - \mathbb{P}[\text{block error}] \le \frac{d(\delta \| \bar{\delta})\mathcal{E} + \log 2}{\log K}$$

$Eb/N_0 \to \infty$ for classical probability of error

Theorem (AWGN)

Suppose K users send one bit each with finite energy $\mathcal{E}$ over the GMAC (with arbitrary n): $Y^n = \sum_{i=1}^K X_i + Z^n$. Then we have

$$\mathbb{P}[X_1 = \hat{X}_1, \ldots, X_K = \hat{X}_K] \le \frac{\mathcal{E}\frac{\log e}{2} + \log 2}{\log K}\,.$$

Same proof:

Theorem (BSC)

Let G be a $K \times n$ generating matrix with $\le \mathcal{E}$ ones per row. Then over $BSC(\delta)$ and all n:

$$1 - \mathbb{P}[\text{block error}] \le \frac{d(\delta\|\bar{\delta})\mathcal{E} + \log 2}{\log K}$$

Puzzle: Genie + Fano method fails for BEC! (Proof by induction works.)

K-user GMAC under PUPE: surprise

- Per-user probability of error as

$$P_e = \frac{1}{K}\sum_{i=1}^{K}\mathbb{P}[W_i \neq \hat{W}_i]\,.$$

- Let's forget about $K = \mu n$ and consider ...
- **Classical regime:** K-fixed, power P fixed, $n \to \infty$. Symmetric capacity

$$C_{sym}(K) = \frac{1}{2K}\log(1 + KP)\,.$$

- But no strong converse (!)

$$C_{sym,\epsilon}(K) > C_{sym}(K-1) \qquad \forall \epsilon \gtrsim \frac{1+\log_e K}{K}$$

- Lesson: When PUPE above $\frac{\log K}{K}$, far from usual GMAC+JPE.

K-user GMAC under PUPE: no strong converse

- Let $C_{sym,\epsilon}(K)$ be the max achievable symmetric rate (K-fixed, $n \to \infty$) under PUPE

$$\frac{1}{K}\sum_{i=1}^{K}\mathbb{P}[W_i \neq \hat{W}_i] \leq \epsilon\,.$$

K-user GMAC under PUPE: no strong converse

- Let $C_{sym,\epsilon}(K)$ be the max achievable symmetric rate (K-fixed, $n \to \infty$) under PUPE

$$\frac{1}{K}\sum_{i=1}^{K}\mathbb{P}[W_i \neq \hat{W}_i] \le \epsilon\,.$$

Theorem (P.-Telatar'16)

We have: $C_{sym,\epsilon}(K,\epsilon) = \begin{cases} \frac{1}{2K}\log(1+KP), & \epsilon < 1/K \\ \ge \frac{1}{2(K-1)}\log(1+(K-1)P), & \epsilon \gtrsim \frac{1+\log_e K}{K} \end{cases}$

- Note that sequence: $\frac{1}{2K}\log(1+KP)$ is monotonically decreasing.
- First part: by union bound PUPE $\le \epsilon$ implies JPE $\le K\epsilon$ + strong-converse for GMAC.

K-user GMAC under PUPE: no strong converse

- Let $C_{sym,\epsilon}(K)$ be the max achievable symmetric rate (K-fixed, $n \to \infty$) under PUPE

$$\frac{1}{K}\sum_{i=1}^{K} \mathbb{P}[W_i \neq \hat{W}_i] \le \epsilon\,.$$

Theorem (P.-Telatar'16)

We have: $C_{sym,\epsilon}(K,\epsilon) = \begin{cases} \frac{1}{2K}\log(1+KP), & \epsilon < 1/K \\ \ge \frac{1}{2(K-1)}\log(1+(K-1)P), & \epsilon \gtrsim \frac{1+\log_e K}{K} \end{cases}$

- Note that sequence: $\frac{1}{2K}\log(1+KP)$ is monotonically decreasing.
- First part: by union bound PUPE $\le \epsilon$ implies JPE $\le K\epsilon$ + strong-converse for GMAC.
- Second part: Choose codebooks for symmetric-rate point of $(K-1)$-GMAC
- Each user sends 0 w.p. ϵ. Then w.p. $1-(1-\epsilon)^K$ only $(K-1)$ are active.

New twists compared to classic MAC

Problem 1 large $K \to \infty$, fixed payload $\log_2 M$

☑ Relevant asymptotics: $K, n \to \infty$ with $\frac{K}{n} = \mu$.

Problem 2 "user-centric" probability of error

☑ $$P_e \triangleq \frac{1}{K} \sum_j \mathbb{P}[\hat{X}_j \neq X_j]$$

Problem 3 "random-access"

indistiguishable users (same-codebook), non-asymptotics.

Fundamental limits: for K-user MAC with $K \sim n$

Recap: MAC setting and performance metrics

- Perfectly synchronized K-user Gaussian MAC with blocklength n
- Each user transmits $\log_2 M$ bits.
- Figures of merit: energy-per-bit and user density

$$\frac{E_b}{N_0} \triangleq \frac{\mathbb{E}[\|X^n\|^2]}{2\log_2 M} \qquad \mu \triangleq \frac{K}{n}$$

- Regime: $K = \mu n$, $n \to \infty$.
- PUPE definition: $P_e \triangleq \frac{1}{K}\sum_{j=1}^{K} \mathbb{P}[X_j \neq \hat{X}_j]$.

Next: new results

Fundamental limits: for K-user MAC with $K \sim n$

Recap: MAC setting and performance metrics

- Perfectly synchronized K-user Gaussian MAC with blocklength n
- Each user transmits $\log_2 M$ bits.
- Figures of merit: energy-per-bit and user density

$$\frac{E_b}{N_0} \triangleq \frac{\mathbb{E}[\|X^n\|^2]}{2\log_2 M} \qquad \mu \triangleq \frac{K}{n}$$

- Regime: $K = \mu n$, $n \to \infty$.
- PUPE definition: $P_e \triangleq \frac{1}{K}\sum_{j=1}^{K} \mathbb{P}[X_j \neq \hat{X}_j]$.

Next: new results

- Converse bound (via reduction to known problems)
- Achievability bound (via Gaussian process theory)

Theorem

Communication with (μ, M, ϵ) is asymptotically ($n \to \infty$) feasible only if both of these hold:

$$\begin{aligned}(1-\epsilon)\mu \log_2 M &\le \frac{1}{2}\log_2(1+\mu P_{tot}) + \mu h(\epsilon) \\ \frac{1}{M} &\ge Q\left(\sqrt{\frac{P_{tot}}{\mu}} + Q^{-1}(1-\epsilon)\right).\end{aligned}$$

where $P_{tot} = 2\mu \log_2 M \cdot \frac{E_b}{N_0}$ is the total received power.

- First bound: A working code recovers $W \in [M]^K$ with Hamming distortion $\le \epsilon$. Comparing sum-capacity with rate-distortion function we get the bound.

Theorem

Communication with (μ, M, ϵ) is asymptotically ($n \to \infty$) feasible only if both of these hold:

$$
\begin{aligned}
(1-\epsilon)\mu \log_2 M &\le \frac{1}{2}\log_2(1+\mu P_{tot}) + \mu h(\epsilon) \\
\frac{1}{M} &\ge Q\left(\sqrt{\frac{P_{tot}}{\mu}} + Q^{-1}(1-\epsilon)\right).
\end{aligned}
$$

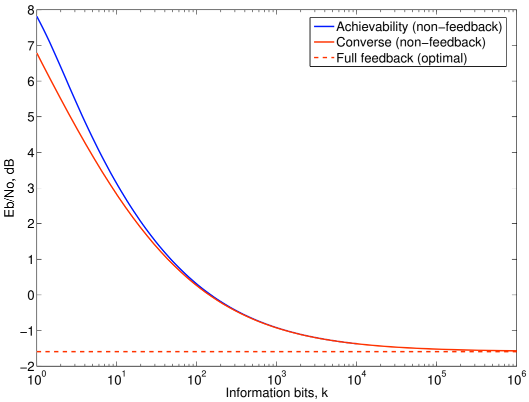
where $P_{tot} = 2\mu \log_2 M \cdot \frac{E_b}{N_0}$ is the total received power.

- Second bound: To get small $\frac{E_b}{N_0}$ one necessarily needs to code over large payloads (i.e. $\log_2 M \gg 1$) – this is [PPV'11].
- Namely, we use the genie argument. At least one of K users should have $P_e \le \epsilon$.
- Even if that user communicated alone over a $n = \infty$ AWGN channel, he'd need large total energy-per-bit if M is small.

Theorem

Communication with (μ, M, ϵ) is asymptotically ($n \to \infty$) feasible only if ***both*** *of these hold:*

[P.-Poor-Verdú'11]



where P_{tot} ...

- Second... ...de over large p...
- Namely ... should have P...
- Even if ... channel, he'd need large total energy-per-bit if M is small.

Theorem (Thrampoulidis-Zadik-P.'18)

For each $\beta > 0$ *there exists codes with* $\frac{E_b}{N_0} = \frac{\beta^2}{2\log_2 M}$ *and PUPE* ϵ *provided that*

$$\theta\mu\log M + \mu h(\theta) < \frac{1}{2}\log(1+\beta^2\theta\mu) + \frac{\log e}{2}\left(\frac{\psi(\beta,\theta,\mu)}{1+\beta^2\theta\mu} - 1\right)$$

for all $\theta \in [\epsilon, 1]$ *where*

$$\psi(\beta,\theta,\mu) = \sqrt{1+\beta^2\theta\mu} - \frac{\beta\mu}{\sqrt{2\pi}} e^{-\frac{1}{2}(Q^{-1}(\theta))^2}$$

Theorem (Thrampoulidis-Zadik-P.'18)

For each $\beta > 0$ there exists codes with $\frac{E_b}{N_0} = \frac{\beta^2}{2\log_2 M}$ and PUPE ϵ provided that

$$\theta\mu \log M + \mu h(\theta) < \frac{1}{2}\log(1+\beta^2\theta\mu) + \frac{\log e}{2}\left(\frac{\psi(\beta,\theta,\mu)}{1+\beta^2\theta\mu} - 1\right)$$

for all $\theta \in [\epsilon, 1]$ where

$$\psi(\beta,\theta,\mu) = \sqrt{1+\beta^2\theta\mu} - \frac{\beta\mu}{\sqrt{2\pi}} e^{-\frac{1}{2}(Q^{-1}(\theta))^2}$$

Proof outline:

- Use random gaussian codebooks
- Use maximum likelihood decoder (not optimal!): $\min \|Y - \sum_i c_i\|_2$
- Use information-density thresholding trick
- Use Gaussian process theory (Gordon's lemma) to evaluate the bound

- Generate codewords $c_m^{(j)} \stackrel{iid}{\sim} \mathcal{N}(0, PI_n)$, $j \in [K], m \in [M]$, where $P = \frac{\beta^2}{n}$
- Use ML decoder (suboptimal!):

$$\hat{W} = \mathrm{argmin}_{w_1,\dots,w_K} \|Y - (c_{w_1}^{(1)} + \cdots + c_{w_K}^{(K)})\|_2^2 \,.$$

- Define

$$F(S_0) = \{\exists (m_j)_{j\in S_0} : \|Y - (c(S_0^c) + \sum_{j\in S_0} c_{m_j}^{(j)})\|_2 \le \|Y - c([K])\|_2, m_j \neq$$

- We have:

$$\mathbb{P}[d_H(W, \hat{W}) = t] \le \mathbb{P}\left[\cup_{S_0:|S_0|=t} F(S_0)\right]$$

- Main goal: Show $\mathbb{P}\left[\cup_{S_0:|S_0|=t} F(S_0)\right] \to 0$ for all $t = \theta n, \theta \in [\epsilon, 1]$.
- Intermediate step: Bound $\mathbb{P}[F(S_0) | c_{[K]}, Y, W_{[K]}]$

- Define information density

$$i_t(u;y|v) = \frac{n}{2}\log(1+Pt) + \frac{\log e}{2}\left(\frac{\|y-v\|_2^2}{1+P't} - \|y-u-v\|_2^2\right),$$

- Define $c(T) = \sum_{j\in T} c^{(j)}_{W_j}$, $c' = \sum_{j\in S_0} c^{(j)}_{m_j}$ for some $m_j \neq W_j$. Then:

$$\{\|Y-(c(S_0^c)+c')\|_2 \le \|Y-c([K])\|_2\} = \{i_t(c';Y|c(S_0^c)) \ge i_t(c(S_0);Y|c$$

- Let $A_1,\dots,A_K \overset{iid}{\sim} \mathcal{N}(0,PI_n)$ and $B = \sum_i A_i + Z$. For any $S_0 \in \binom{[K]}{t}$:

$$\log \frac{dP_{A_{S_0}|A_{S_0^c},B}}{dP_{A_{S_0^c}}} = i_t(u;y|v),$$

where $u = \sum_{j\in S_0} A_j$, $v = \sum_{j\in S_0^c} A_j$and $y = B$.
- And thus we get:

$$\mathbb{P}\left[i_t(c';Y|c(S_0^c)) > \gamma | Y, c_{[K]}, W_{[K]}\right] \le e^{-\gamma}$$

- We have shown (via union bound):

$$\mathbb{P}[F(S_0)|c_{[K]}, Y, W_{[K]}] \le M^t \exp\{-i_t(c(S_0); Y|c(S_0^c))\}\,.$$

- So we now use a smart union bound:

$$\mathbb{P}\left[\cup_{S_0} F(S_0)\right] \le M^t \binom{K}{t} \exp\{-\gamma\} + \mathbb{P}[I_t \le \gamma]\,,$$

  where $I_t = \min_{S_0} i_t(c(S_0); Y|c(S_0^c))$

- Left to study the extrema of Gaussian matrix $G \in \mathbb{R}^{n\times\mu n}$ with $\stackrel{iid}{\sim} \mathcal{N}(0,1)$

$$\Phi \triangleq \frac{1}{n}\min\left\{\left\|\frac{\beta}{\sqrt{n}}Gx + Z\right\|_2 : x \in \{0,1\}^{\mu n}, \|x\|_0 = \theta\mu n\right\}$$

- After dualizing norm, we get a problem:

$$\mathbb{P}[\min_u \max_v A_{u,v} \le c] \le ?$$

- We have shown (via union bound):

$$\mathbb{P}[F(S_0)|c_{[K]}, Y, W_{[K]}] \le M^t \exp\{-i_t(c(S_0); Y|c(S_0^c))\}\,.$$

- So we now use a smart union bound:

$$\mathbb{P}\left[\cup_{S_0} F(S_0)\right] \le M^t \binom{K}{t} \exp\{-\gamma\} + \mathbb{P}[I_t \le \gamma]\,,$$

where $I_t = \min_{S_0} i_t(c(S_0); Y|c(S_0^c))$

- Left to study the extrema of Gaussian matrix $G \in \mathbb{R}^{n\times\mu n}$ with $\stackrel{iid}{\sim} \mathcal{N}(0,1)$

$$\Phi \triangleq \frac{1}{n} \min\left\{\left\|\frac{\beta}{\sqrt{n}} G x + Z\right\|_2 : x \in \{0,1\}^{\mu n}, \|x\|_0 = \theta\mu n\right\}$$

- After dualizing norm, we get a problem:

$$\mathbb{P}[\min_u \max_v A_{u,v} \le c] \le \mathbb{P}[\min_u \max_v B_{u,v} \le c]$$

- Gaussian comparison method: Bound extrema of A via extrema of a simpler process B

Slepian's lemma (1962) and Gordon's lemma (1985)

Theorem (Slepian)

Let $\{A_v\}_{v\in\mathcal{V}}$ *and* $\{B_v\}_{v\in\mathcal{V}}$ *be zero-mean Gaussian processes, s.t.* $\mathrm{Cov}(A) \le \mathrm{Cov}(B)$ *and* $\mathrm{Var}[A_v] = \mathrm{Var}[B_v]$ *for all* v *then*

$$\mathbb{E}[\max_v A_v] \ge \mathbb{E}[\max_v B_v]$$

Slepian's lemma (1962) and Gordon's lemma (1985)

Theorem (Slepian)

Let $\{A_v\}_{v\in\mathcal{V}}$ *and* $\{B_v\}_{v\in\mathcal{V}}$ *be zero-mean Gaussian processes, s.t.* $\mathrm{Cov}(A) \le \mathrm{Cov}(B)$ *and* $\mathrm{Var}[A_v] = \mathrm{Var}[B_v]$ *for all* v *then*

$$\max_v A_v \succeq \max_v B_v \qquad \text{(stoch. domination)}$$

Slepian's lemma (1962) and Gordon's lemma (1985)

Theorem (Slepian)

Let $\{A_v\}_{v\in\mathcal{V}}$ *and* $\{B_v\}_{v\in\mathcal{V}}$ *be zero-mean Gaussian processes, s.t.* $\mathrm{Cov}(A) \le \mathrm{Cov}(B)$ *and* $\mathrm{Var}[A_v] = \mathrm{Var}[B_v]$ *for all* v *then*

$$\max_v A_v \succeq \max_v B_v \qquad \textit{(stoch. domination)}$$

Theorem (Gordon)

Let $\{A_{u,v}\}$ *and* $\{B_{u,v}\}$ *be zero-mean Gaussian processes, s.t.*

1. $\mathrm{Var}[A_{u,v}] = \mathrm{Var}[B_{u,v}]$
2. $\mathbb{E}[A_{u,v}A_{u,v'}] \le \mathbb{E}[B_{u,v}B_{u,v'}]$ *for all* u, v, v'
3. $\mathbb{E}[A_{u,v}A_{u',v'}] \ge \mathbb{E}[B_{u,v}B_{u',v'}]$ *for all* $u \ne u', v, v'$. ***Then:***

$$\min_u \max_v A_{u,v} \succeq \min_u \max_v B_{u,v}$$

Slepian's lemma (1962) and Gordon's lemma (1985)

Theorem (Slepian)

Let $\{A_v\}_{v\in\mathcal{V}}$ *and* $\{B_v\}_{v\in\mathcal{V}}$ *be zero-mean Gaussian processes, s.t.* $\mathrm{Cov}(A) \le \mathrm{Cov}(B)$ *and* $\mathrm{Var}[A_v] = \mathrm{Var}[B_v]$ *for all* v *then*

$$\max_v A_v \succeq \max_v B_v \qquad \textit{(stoch. domination)}$$

Theorem (Gordon)

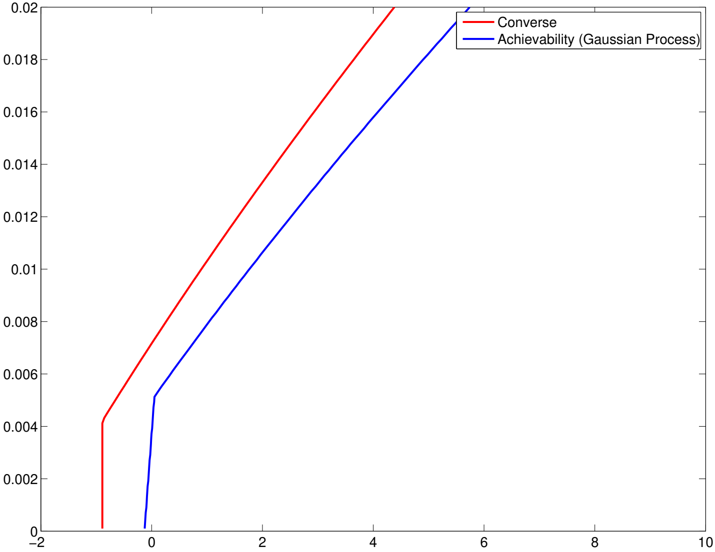
Let $\{A_{u,v}\}$ *and* $\{B_{u,v}\}$ *be zero-mean Gaussian processes, s.t.*

1. $\mathrm{Var}[A_{u,v}] = \mathrm{Var}[B_{u,v}]$
2. $\mathbb{E}[A_{u,v}A_{u,v'}] \le \mathbb{E}[B_{u,v}B_{u,v'}]$ *for all* u, v, v'
3. $\mathbb{E}[A_{u,v}A_{u',v'}] \ge \mathbb{E}[B_{u,v}B_{u',v'}]$ *for all* $u \ne u', v, v'$. ***Then:***

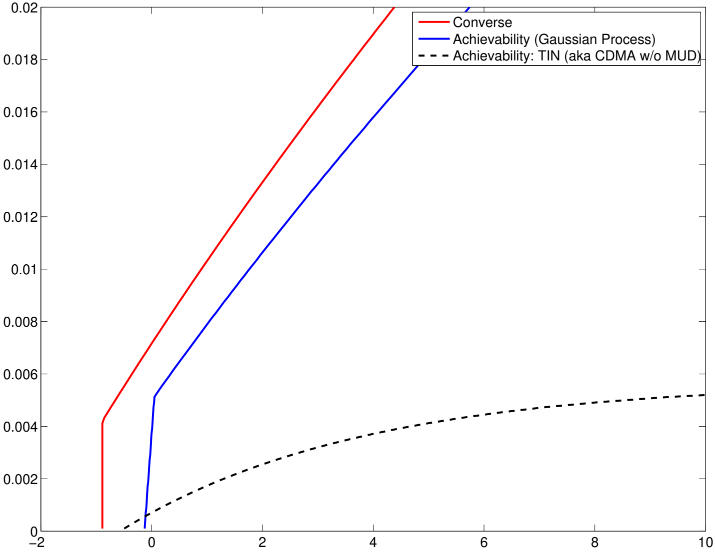
$$\min_u \max_v A_{u,v} \succeq \min_u \max_v B_{u,v}$$

Remark: 2) implies $A_u^* = \max_v A_{u,v} \succeq B_u^* = \max_v B_{u,v}$.
3) implies $\{A_u^*\}$ is "more-correlated" than $\{B_u^*\}$

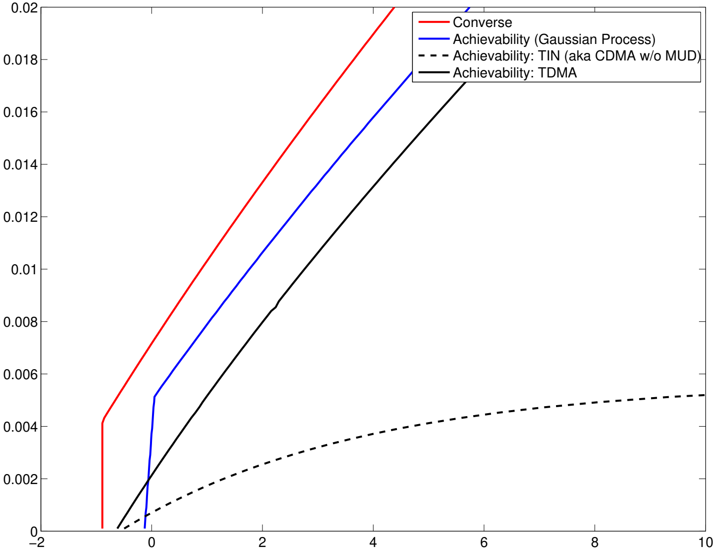
User density vs. Energy-per-bit: best bounds



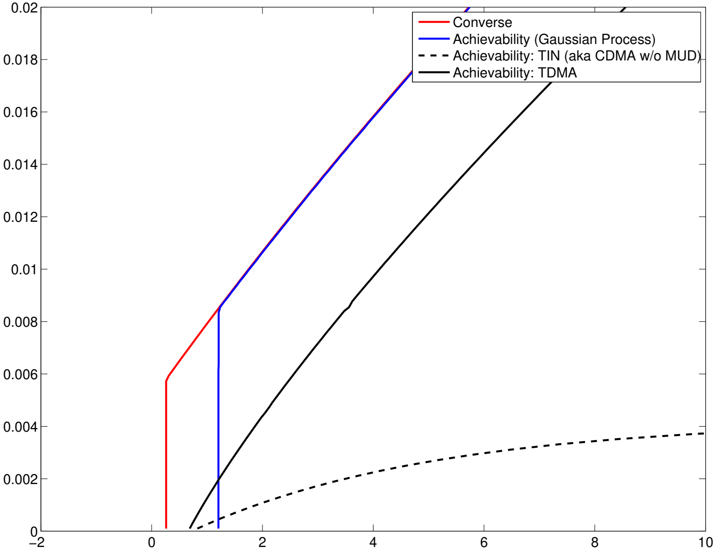
User density vs. Energy-per-bit: CDMA (w/o MUD)



User density vs. Energy-per-bit: TDMA



User density vs. Energy-per-bit: higher reliability



Problem 3: Information theory of random-access

Prior work on MAC/random-access

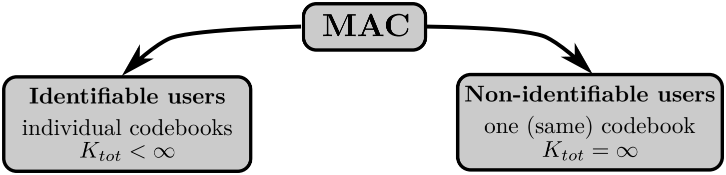
It's a mess...

Prior work on MAC/random-access

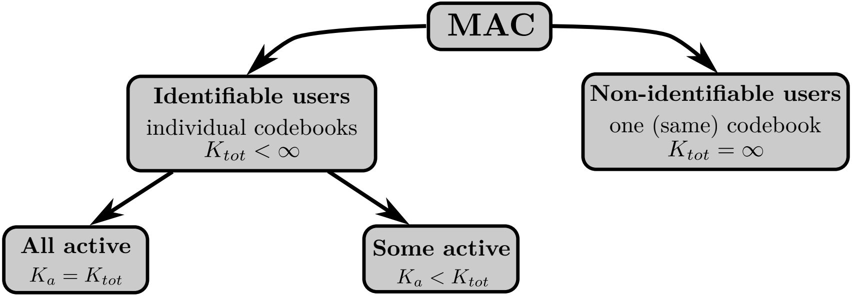
It's a mess...

- Channel model: collision vs. additive
- Noise model: noiseless, stochastic or worst-case
- Coding with or without feedback (as in CSMA)
- Probability of error: zero, vanishing or fixed > 0.
- Probability of error: per-user vs all-users
- User activity: always-on vs sporadic
- finite blocklength vs $n \to \infty$
- Various asymptotics: $K = \mathrm{const}, n \to \infty$ vs both $K, n \to \infty$

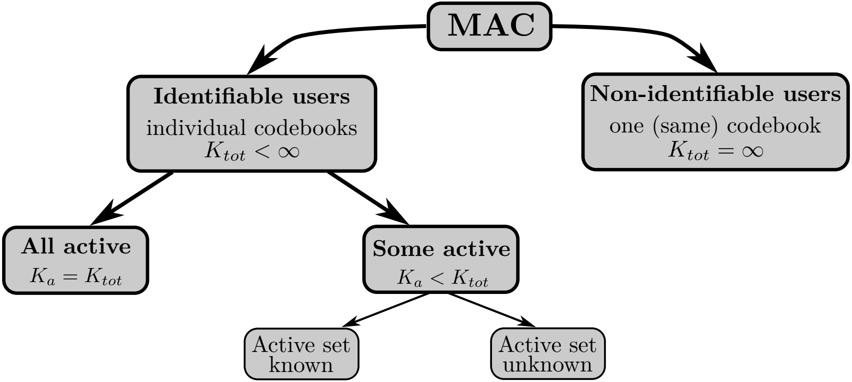
Classification by user activity



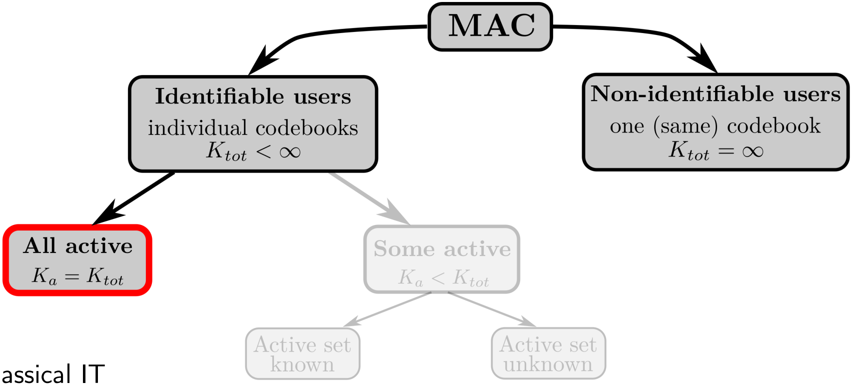
Classification by user activity



Classification by user activity



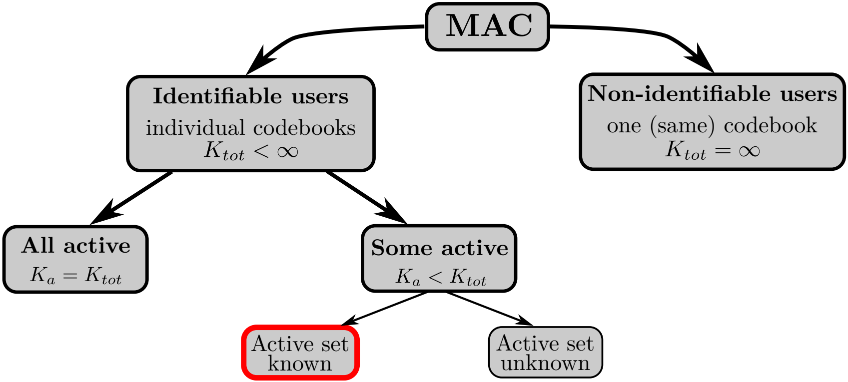
Sample of prior work



- Classical IT

  [Liao'72],[Ahlswede'73]
- Orthogonal schemes TDMA/FDMA
- Rate splitting [Rimoldi-Urbanke'99]
- Finite blocklength [MolavianJazi-Laneman'14-16]
- Many-user [Chen-Guo'14]

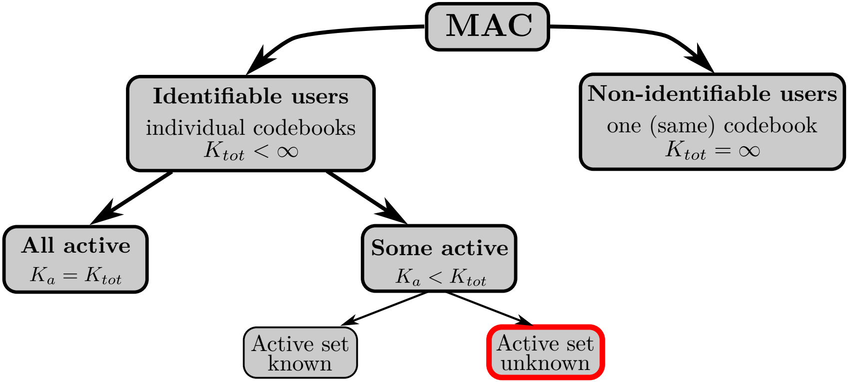
Sample of prior work



- Non-orthogonal CDMA, MUD
- Randomly-spread CDMA

  [Tse-Hanly'99], [Verdú-Shamai'99]
- [Mathys'90]
- LDS, SCMA

Sample of prior work



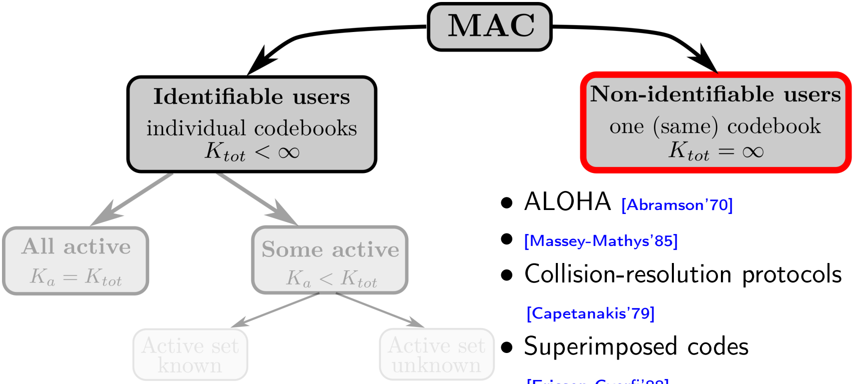
- Non-orthogonal CDMA, MUD
- Randomly-spread CDMA

  [Tse-Hanly'99], [Verdú-Shamai'99]
- [Mathys'90]
- LDS, SCMA

- Many-access [Chen-Chen-Guo'17]
- Blind-detection for CDMA
- [BarDavid-Plotnik-Rom'93]
- conflict-avoiding codes

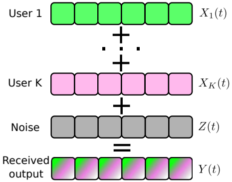
  [Bassalygo-Pinsker'83], B.Tsybakov

Sample of prior work



- ALOHA [Abramson'70]
- [Massey-Mathys'85]
- Collision-resolution protocols [Capetanakis'79]
- Superimposed codes [Ericson-Gyorfi'88] [Furedi-Ruszinkó'99]
- B_r-codes [Dyachkov-Rykov'81]
- Coded Slotted ALOHA [Casini et al'07],[Liva'11]
- Compressed sensing [Jin-Kim-Rao'11]

Key definition: random-access code



Definition (P.'17)

$f : [M] \to \mathbb{R}^n$ is a random-access code for K_a users if $\exists$ list-K_a decoder g s.t.

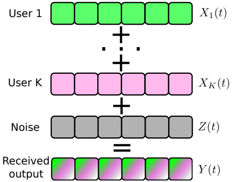
$$\mathbb{P}[W_j \notin g(f(W_1) + \cdots + f(W_{K_a}) + Z)] \le \epsilon \qquad \forall j \in [K_a]$$

where $W_i \overset{iid}{\sim} \mathrm{Unif}[M]$.

For $\epsilon = 0$ this was studied:

- Noiseless channels: B_r-codes [Dyachkhov-Rykov'81]
- Worst-case noise: superimposed codes [Ericson-Gyorfi'88, Furedi-Ruszinkó'99]

Key definition: random-access code



Definition (P.'17)

$f : [M] \to \mathbb{R}^n$ is a random-access code for K_a users if $\exists$ list-K_a decoder g s.t.

$$\mathbb{P}[W_j \notin g(f(W_1) + \cdots + f(W_{K_a}) + Z)] \le \epsilon \qquad \forall j \in [K_a]$$

where $W_i \overset{iid}{\sim} \mathrm{Unif}[M]$.

For $\epsilon > 0$ this is:

- Just compressed sensing: $Y = X\beta + Z$, X is K_a-out-of-M sparse.
- $\Rightarrow$ studied by many, but not w.r.t. $\frac{E_b}{N_0}$ and not with $M = 2^{\Theta(n)}$.

Same-codebook codes = compressed sensing

- random-access = all users share same codebook
- ... obviously decoding is upto permutation of users
- **New problems:** capacity/error-exponent/zero-error capacity
- Equivalent to compressed-sensing [Jin-Kim-Rao'11]

Same-codebook codes = compressed sensing

- random-access = all users share same codebook
- ... obviously decoding is upto permutation of users
- **New problems:** capacity/error-exponent/zero-error capacity
- Equivalent to compressed-sensing [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 comp.sensing).

Same-codebook codes = compressed sensing

- random-access = all users share same codebook
- ... obviously decoding is upto permutation of users
- **New problems:** capacity/error-exponent/zero-error capacity
- Equivalent to compressed-sensing [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:

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

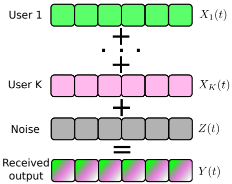
(linear regression with sparsity $\|\beta\|_0 = K_a$ aka comp.sensing).

- The famous $n \sim 2K_a \log_e M$ is just TIN:

$$\log_e M \approx \frac{n}{2} \log_e(1 + \frac{P}{1 + (K_a - 1)P}) \approx \frac{n}{2K_a}$$

So all the L_1 (LASSO) frenzy is just to achieve TIN (hehe...)

Key definition: random-access code



Definition (P.'17)

$f : [M] \to \mathbb{R}^n$ is a random-access code for K_a users if $\exists$ list-K_a decoder g s.t.

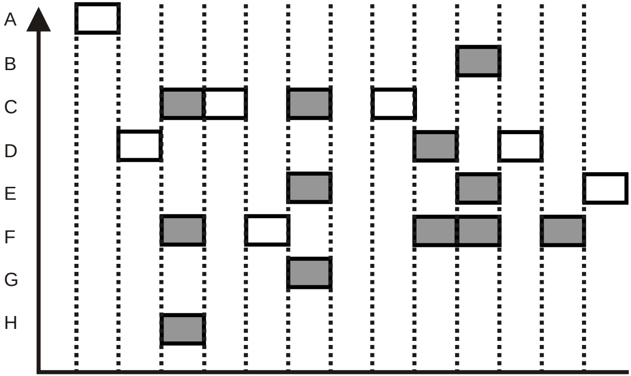
$$\mathbb{P}[W_j \notin g(f(W_1) + \cdots + f(W_{K_a}) + Z)] \le \epsilon \qquad \forall j \in [K_a]$$

where $W_i \overset{iid}{\sim} \mathrm{Unif}[M]$.

This definition is answer to many prayers, but ...

Bad news: Asymptotics of $K_a = \mu n$, $n \to \infty$ is nonsense.

Prototypical random-access code: ALOHA



Slotted ALOHA protocol (shaded slots indicate collision)

- n-frame is partitioned into $L = \frac{n}{n_1}$ subframes of length n_1
- Each of K_a users places his n_1-codeword into a random subframe.
- Per-user error: $P_e \approx \mathbb{P}[\mathrm{Bino}(K_a - 1, \frac{1}{L}) > 0] \approx \frac{K_a}{L} e^{-\frac{K_a}{L}}$

Main result 2: random-coding bound

II. Random coding bound

Theorem 1. *Fix $P' < P$. There exists an (M, n, ϵ) random-access code for K_a-user GMAC satisfying power-constraint P and*

$$\epsilon \le \sum_{t=1}^{K_a} \frac{t}{K_a} \min(p_t, q_t) + p_0\,, \qquad (3)$$

where

$$p_0 = \frac{\binom{K_a}{2}}{M} + K_a \mathbb{P}\left[\frac{1}{n}\sum_{j=1}^{n} Z_j^2 > \frac{P}{P'}\right], \qquad (4)$$

$$p_t = e^{-nE(t)}, \qquad (5)$$

$$E(t) = \max_{0\le\rho,\rho_1\le 1} -\rho\rho_1 t R_1 - \rho_1 R_2 + E_0(\rho,\rho_1)$$

$$E_0 = \rho_1 a + \frac{1}{2}\log(1 - 2b\rho_1)$$

$$a = \frac{\rho}{2}\log(1 + 2P't\lambda) + \frac{1}{2}\log(1 + 2P't\mu) \qquad (6)$$

$$b = \rho\lambda - \frac{\mu}{1 + 2P't\mu},\ \mu = \frac{\rho\lambda}{1 + 2P't\lambda} \qquad (7)$$

$$\lambda = \frac{P't - 1 + \sqrt{D}}{4(1 + \rho_1\rho)P't}, \qquad (8)$$

$$D = (P't - 1)^2 + 4P't\frac{1 + \rho\rho_1}{1 + \rho}$$

$$R_1 = \frac{1}{n}\log M - \frac{1}{n}\log(t!) \qquad (9)$$

$$R_2 = \frac{1}{n}\log\binom{K_a}{t} \qquad (10)$$

$$q_t = \inf_\gamma \mathbb{P}[I_t \le \gamma] + \exp\{n(R_1 + R_2) - \gamma\}$$

Remark: For classical regime K_a-fixed, $n \to \infty$ and $\epsilon \to 0$

$$C_{random-access}(K_a) = \frac{1}{2K_a}\log(1 + K_a P)\,.$$

Random-coding achievability bound

- Generate M codewords: $c_i \sim \mathcal{N}(0, P)^{\otimes n}$.
- WLOG, users send $c_1, c_2, \ldots, c_{K_a}$.
- Decoder sees

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

- Define sum-codewords $c(S) \triangleq \sum_{i \in S} c_i$
- ML-decoder (not optimal!)

$$\hat{S} = \arg\min_S \|c(S) - Y\| .$$

- Error-analysis:

$$P_e \le \sum_{t=1}^{K_a} \frac{t}{K_a} \mathbb{P}[t\text{–misguessed}]$$

$$\mathbb{P}[t\text{–misguessed}] \le \boxed{\mathbb{P}\left[\bigcup_{S \in \binom{K_a}{t}} \bigcup_{S' \in \binom{M-K_a}{t}} \|c(S) - c(S') + Z\| \le \|Z\|\right]}$$

- Generate M codewords: $c_i \sim \mathcal{N}(0, P)^{\otimes n}$

Analysis I:

- Condition on $Z, c_1, \ldots, c_{K_a}$
- Use Chernoff + Gallager ρ-trick for $\mathbb{P}[\cup_{S'} \cdots | c_1^{K_a}, Z]$
- Use another Gallager ρ-trick for $\mathbb{P}[\cup_{S} \cdots | Z]$
- Finally take expectation over Z

- ML-decoder (not optimal!)

$$\hat{S} = \arg\min_{S} \|c(S) - Y\| \,.$$

- Error-analysis:

$$P_e \le \sum_{t=1}^{K_a} \frac{t}{K_a} \mathbb{P}[t\text{–misguessed}]$$

$$\mathbb{P}[t\text{–misguessed}] \le \boxed{\mathbb{P}\left[\bigcup_{S \in \binom{K_a}{t}} \bigcup_{S' \in \binom{M-K_a}{t}} \|c(S) - c(S') + Z\| \le \|Z\|\right]}$$

Random-coding achievability bound

- Generate M codewords: $c_i \sim \mathcal{N}(0, P)^{\otimes n}$

Analysis I:

- Condition on $Z, c_1, \ldots, c_{K_a}$
- Use Chernoff + Gallager ρ-trick for $\mathbb{P}[\cup_{S'} \cdots | c_1^{K_a}, Z]$
- Use another Gallager ρ-trick for $\mathbb{P}[\cup_S \cdots | Z]$
- Finally take expectation over Z

- ML-decoder (not optimal!)

Analysis II:

- Define information density appropriately
- Use Feinstein's trick to bound
  $\mathbb{P}[\cup_S \cup_{S'} \cdots] \le \mathbb{P}[i_{min}(X_1^{K_a}; Y) < \gamma] + \binom{K_a}{t}\binom{M}{t} e^{-\gamma}$
  $i_{min} = \min_S i_t(c(S); Y | c(S^c))$
- $i_{min} \approx$ max of Gaussian process indexed by t-subsets of $[K_a]$

$$\mathbb{P}[t\text{–misguessed}] \le \mathbb{P}\left[\bigcup_{S \in \binom{K_a}{t}} \bigcup_{S' \in \binom{M-K_a}{t}} \|c(S) - c(S') + Z\| \le \|Z\|\right]$$

Random-coding achievability bound

- Generate M codewords: $c_i \sim \mathcal{N}(0, P)^{\otimes n}$

Analysis I:

- Condition on $Z, c_1, \ldots, c_{K_a}$
- Use Chernoff + Gallager ρ-trick for $\mathbb{P}[\cup_{S'} \cdots | c_1^{K_a}, Z]$
- Use another Gallager ρ-trick for $\mathbb{P}[\cup_S \cdots | Z]$
- Finally take expectation over Z

- ML-decoder (not optimal!)

Analysis II:

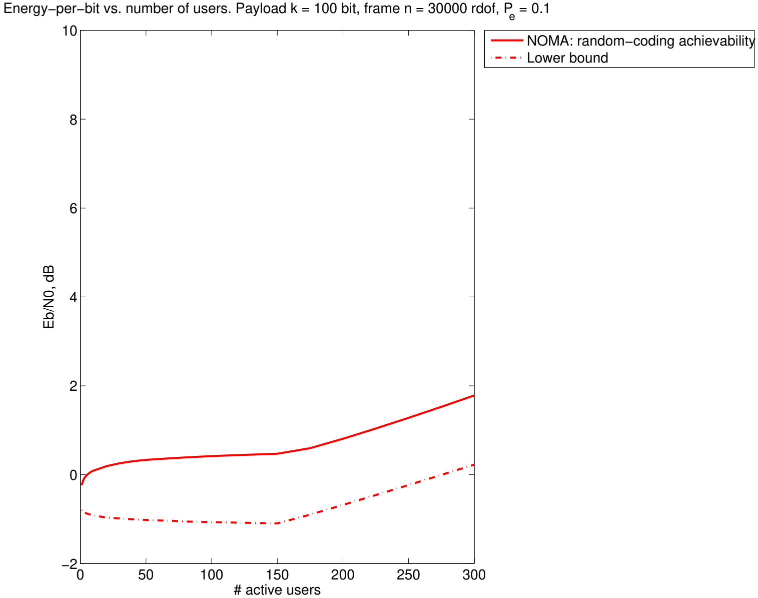
- Define information density appropriately
- Use Feinstein's trick to bound

  $\mathbb{P}[\cup_S \cup_{S'} \cdots] \le \mathbb{P}[i_{min}(X_1^{K_a}; Y) < \gamma] + \binom{K_a}{t}\binom{M}{t} e^{-\gamma}$

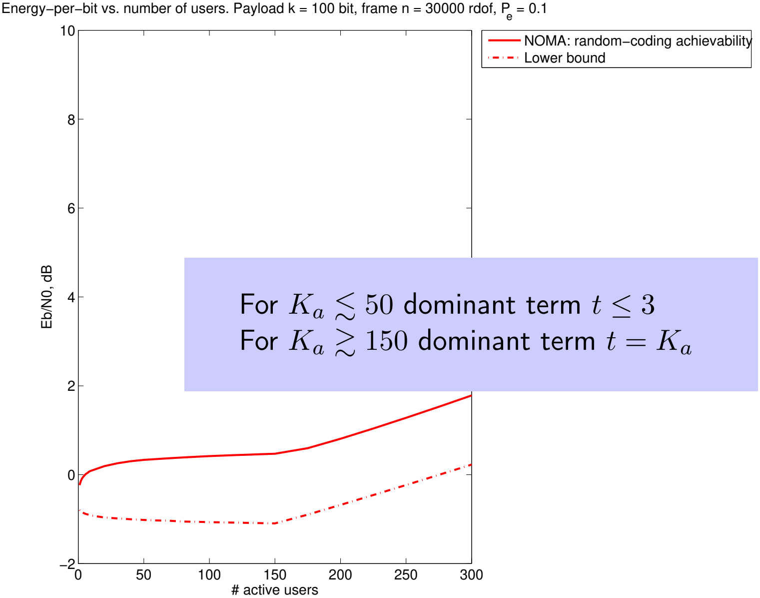
  $i_{min} = \min_S i_t(c(S); Y | c(S^c))$
- $i_{min} \approx$ max of Gaussian process indexed by t-subsets of $[K_a]$

Classical IT: term S goes $\to 0$ if $I(X_S; Y | X_{S^c}) > \sum_{i \in S} R_i$

Numerical evaluation

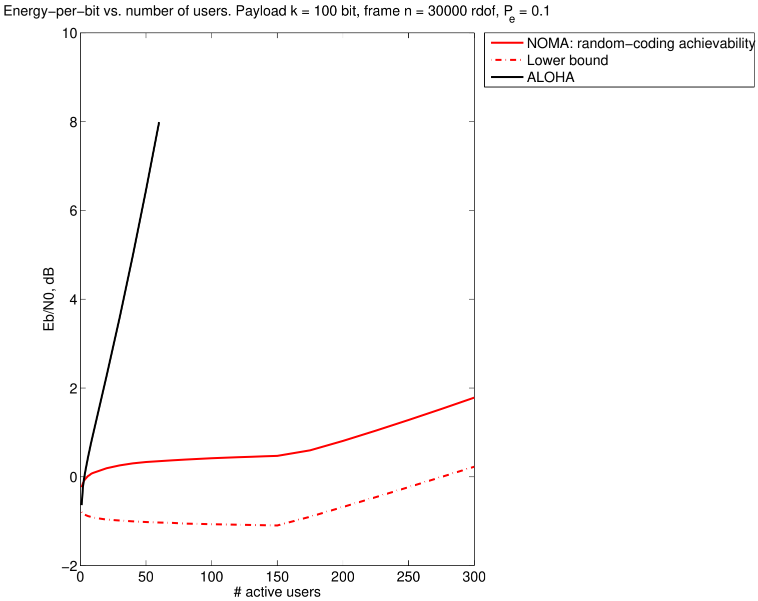


Numerical evaluation

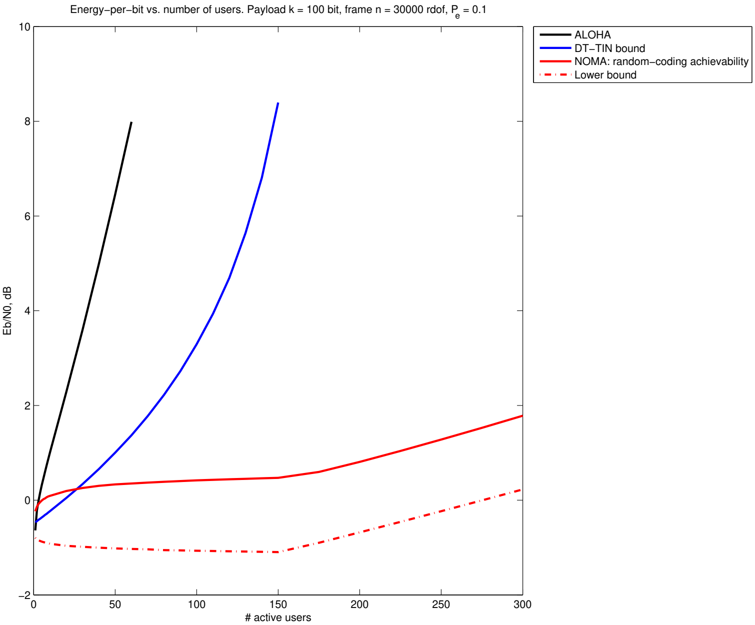


For $K_a \lesssim 50$ dominant term $t \leq 3$
For $K_a \gtrsim 150$ dominant term $t = K_a$

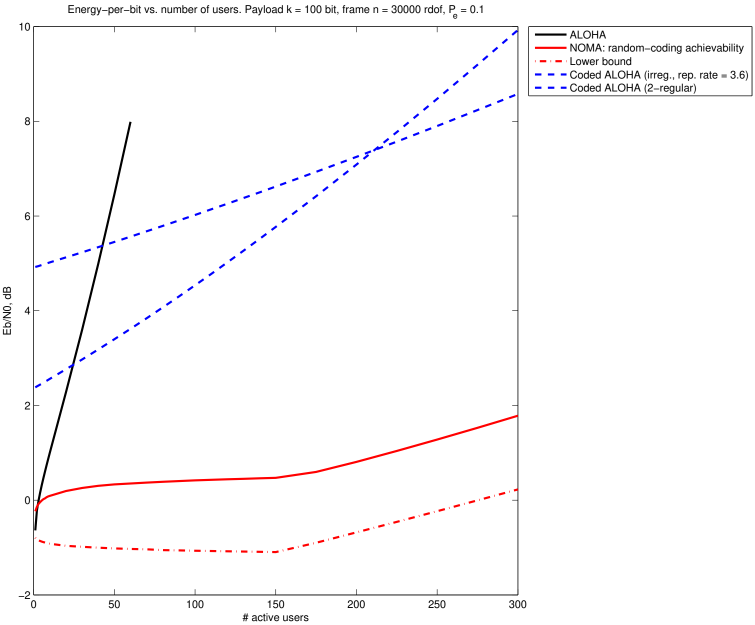
Fundamental limits vs. ALOHA



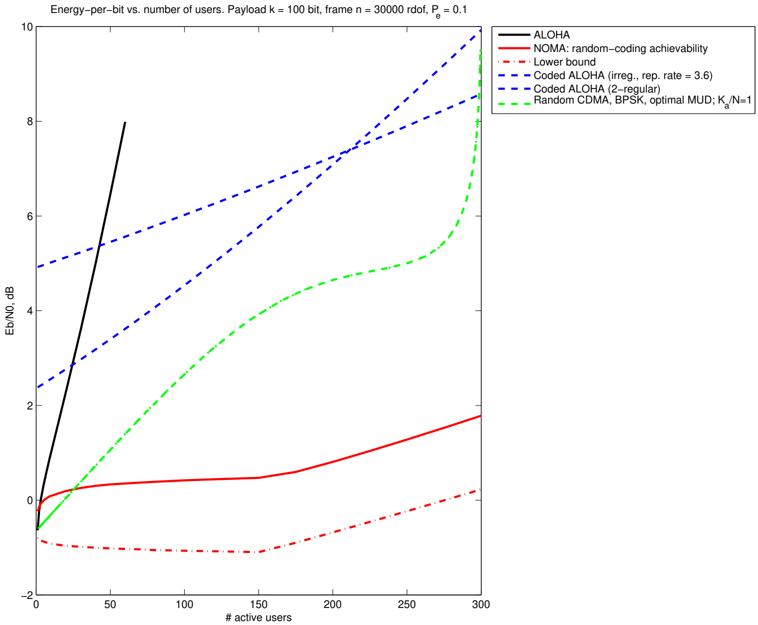
Fundamental limits vs. TIN (aka CDMA w/o MUD)



Fundamental limits vs. Coded Slotted ALOHA



. . . and randomly-spread CDMA w/ optimal MUD



New twists compared to classic MAC

Problem 1 large $K \to \infty$, fixed payload $\log_2 M$

Relevant asymptotics: $K, n \to \infty$ with $\frac{K}{n} = \mu$.

Problem 2 "user-centric" probability of error

$$P_e \triangleq \frac{1}{K} \sum_j \mathbb{P}[\hat{X}_j \neq X_j]$$

Problem 3 "random-access"

indistiguishable users (same-codebook), non-asymptotics.

Low-complexity random-access over GMAC

Key challenge:

Providing multiple-access to massive number of
UNCOORDINATED
and infrequently communicating devices

Key challenge:

Providing multiple-access to massive number of UNCOORDINATED and infrequently communicating devices

Typical scenario:

- Huge # of users $K_{\text{tot}} \approx 10^6 - 10^7$
- Still large # of active users $K_a \approx 1 - 500$
- Small data payload, e.g. $k = 100$ bits
- Blocklength $n \sim 10^4$
- $\frac{k}{n} \ll 1$, but system spectral efficiency $\rho = \frac{K_a \cdot k}{n} \sim 1$

Key challenge:

Providing multiple-access to massive number of
UNCOORDINATED
and infrequently communicating devices

Typical scenario:

- Huge # of users $K_{\text{tot}} \approx 10^6 - 10^7$
- Still large # of active users $K_a \approx 1 - 500$
- Small data payload, e.g. $k = 100$ bits
- Blocklength $n \sim 10^4$
- $\frac{k}{n} \ll 1$, but system spectral efficiency $\rho = \frac{K_a \cdot k}{n} \sim 1$

The goal is to communicate with the smallest possible energy-per-bit

Simple scheme I: Treat interference as noise (TIN)

Theorem (DT-TIN bound)

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

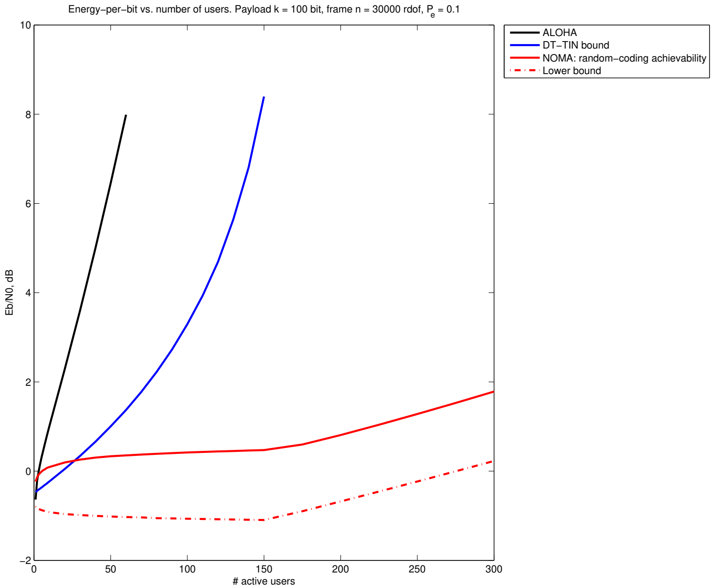
$$\mathbb{P}[X_1 \notin \{\text{top-}K_a \text{ closest c/w to } Y\}] \lesssim \mathbb{E}\left[e^{-|i(X;X+Z)-\log M|^+}\right]$$

where $Y = X_1 + \cdots + X_{K_a} + Z$, X_i – uniform on $\mathcal{C}$, $X \sim \mathcal{N}(0,P)^{\otimes n}$ and $Z \sim \mathcal{N}(0,1)^{\otimes n}$.

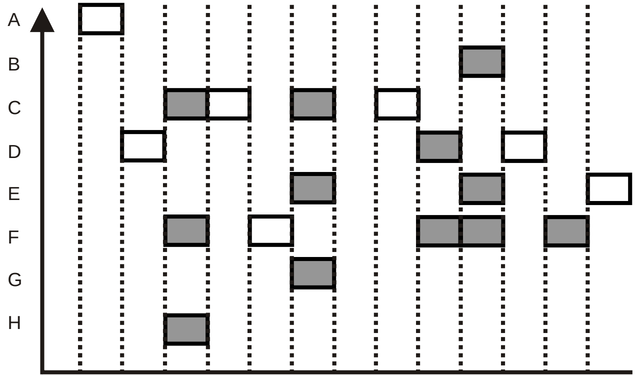
Remarks:

- Decoder searches for top-K_a closest codewords
- Achieves about $\log M \approx nC_{TIN}(P) - \sqrt{nV_{TIN}(P)}Q^{-1}(\epsilon)$
  $$C_{TIN}(P) = \frac{1}{2}\log\left(1 + \frac{P}{1+(K_a-1)P}\right), \quad V_{TIN}(P) = \frac{P\log^2 e}{1+K_a P}.$$
- Spectral efficiency as $K_a \to \infty$ is bounded by $\frac{\log_2 e}{2} \approx 0.72$ bit.

Simple scheme I: Treat interference as noise (TIN)



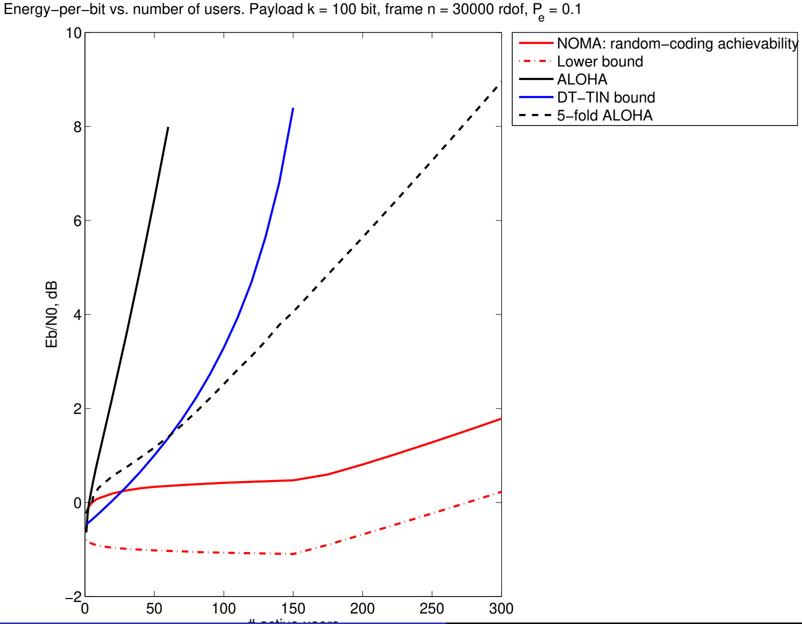
Simple scheme II: T-fold ALOHA



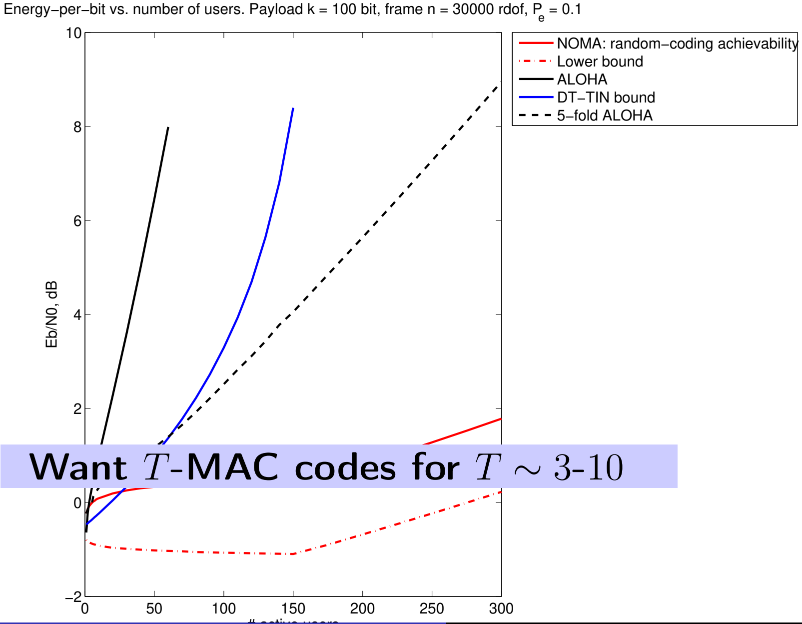
Slotted ALOHA protocol (shaded slots indicate collision)

- Each user places his n_1-codeword into one of L subframes.
- Assume any T-fold collision is resolvable
- Per-user error: $P_e \approx \mathbb{P}[\mathrm{Bino}(K_a - 1, \frac{1}{L}) > T] \approx \left(\frac{K_a}{L}\right)^T e^{-\frac{K_a}{L}}$

Simple scheme II: T-fold ALOHA

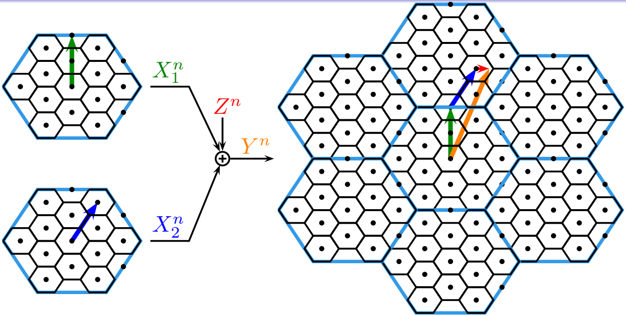


Simple scheme II: T-fold ALOHA



Want T-MAC codes for $T \sim 3$-10

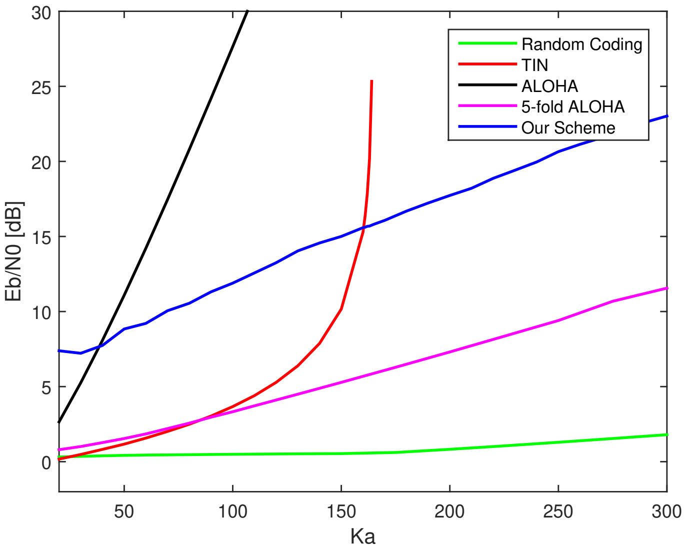
Our scheme: high-level idea



- Send lattice points
- Decode sum of codewords via single-user decoder [Nazer-Gastpar'11]
- Use a subset of points forming a Sidon set (all sums $c_1 + c_2$ distinct)
- Single-lattice (no MMSE scaling): $R \approx \frac{1}{2K} \log^+ P$
- Nested-lattice (with MMSE scaling): $R \approx \frac{1}{2K} \log^+ \left(\frac{1}{K} + P\right)$

  Warning: issues with same-dither
- Lose power-factor compared to $\frac{1}{2K} \log(1 + KP)$

Sample performance of new scheme



Related work

Many ideas appeared separately:

- Compute-and-forward [Nazer-Gastpar'11]
- Explicit codes for the modulo-2 binary adder channel [Lindström'69, Bar-David et al.'93]
- 2-user codes for $\mathbb{F}_q$-adder MAC [Dumer-Zinoviev'78, Dumer'95]
- Concatenation of codes with good minimum distance and codes for the BAC [Ericson-Levenshtein'94]
- Concatenation of CoF inner codes with syndrome decoding for compressed sensing [Lee-Hong'16]

Details of our scheme

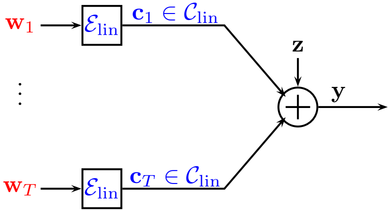
Three phases:

- Sidon set: $\{0,1\}^k \to \mathbb{F}_p^n$
- Compute-and-forward: $\mathbb{F}_p^n \to \mathbb{R}^{n_1}$
- T-fold ALOHA: Place n_1-codeword in a random subframe

Concatenation scheme

Inner code (CoF):

Convert T-user GMAC into a mod-p (noiseless) adder MAC.



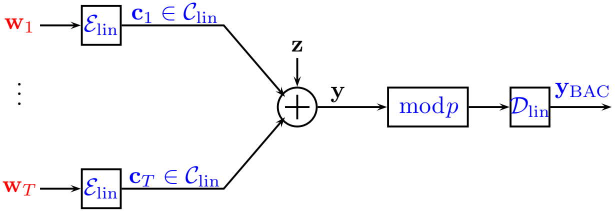
$\mathbf{w}_1, \ldots, \mathbf{w}_T$ are vectors in $\mathbb{Z}_p$

$\mathcal{C}_{\text{lin}}$ is linear code over $\mathbb{Z}_p$

Concatenation scheme

Inner code (CoF):

Convert T-user GMAC into a mod-p (noiseless) adder MAC.



$\mathbf{w}_1, \ldots, \mathbf{w}_T$ are vectors in $\mathbb{Z}_p$

$\mathcal{C}_{\text{lin}}$ is linear code over $\mathbb{Z}_p$

$$\mathbf{y}_{\text{BAC}} = \left[\sum_{i=1}^{T} \mathbf{w}_i\right] \bmod p$$

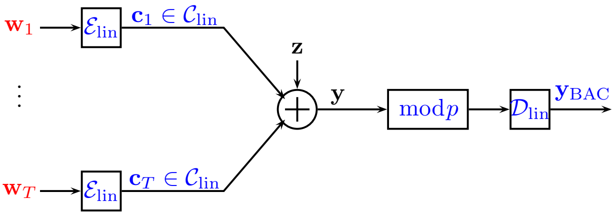
Concatenation scheme

Inner code (CoF):

Convert T-user GMAC into a mod-p (noiseless) adder MAC.

Outer code (BAC):

$\mathcal{C}_{\text{BAC}}$ code for mod-p adder T-MAC   Here: only $p = 2$



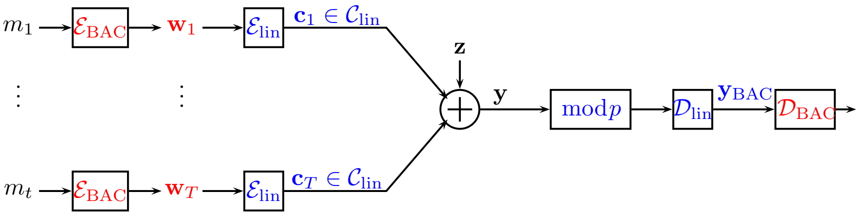
$\mathbf{w}_1, \ldots, \mathbf{w}_T$ are vectors in $\mathbb{Z}_p$

$\mathcal{C}_{\text{lin}}$ is linear code over $\mathbb{Z}_p$

$$\mathbf{y}_{\text{BAC}} = \left[\sum_{i=1}^{T} \mathbf{w}_i\right] \bmod p$$

Concatenation scheme

Inner code (CoF):
Convert T-user GMAC into a mod-p (noiseless) adder MAC.
Outer code (BAC):
$\mathcal{C}_{\text{BAC}}$ code for mod-p adder T-MAC   Here: only $p = 2$



$\mathbf{w}_1, \ldots, \mathbf{w}_T$ are vectors in $\mathbb{Z}_p$

$\mathcal{C}_{\text{lin}}$ is linear code over $\mathbb{Z}_p$

$$\mathbf{y}_{\text{BAC}} = \left[\sum_{i=1}^{T} \mathbf{w}_i\right] \bmod p$$

More on the CoF phase

- $\mathcal{C}_{\mathsf{lin}} \subset \{0,1\}^n$ is a binary linear code (shifted to $\pm\sqrt{P}$)
- Receive $\mathbf{y} = \sum_{i=1}^T \mathbf{x}_i + \mathbf{z}$, shift, rescale, take mod-2, get

$$\mathbf{y}_{\mathsf{CoF}} = [\mathbf{x} + \mathbf{z}] \bmod 2$$

  where $\mathbf{x} = [\sum_i \mathbf{x}_i] \bmod 2 \in \mathcal{C}_{\mathsf{lin}} \subset \{0,1\}^n$
- The channel from $\mathbf{x}$ to $\mathbf{y}_{\mathsf{CoF}}$ is a **BMS with folded Gsn noise**
  $\Longrightarrow$ Designing $\mathcal{C}_{\mathsf{lin}}$ is a standard coding task
  Normal approximation: $\log|\mathcal{C}_{\mathsf{lin}}| \approx nC - \sqrt{nV}Q^{-1}(\epsilon_{code})$

More on the CoF phase

- $\mathcal{C}_{\mathsf{lin}} \subset \{0,1\}^n$ is a binary linear code (shifted to $\pm\sqrt{P}$)
- Receive $\mathbf{y} = \sum_{i=1}^{T} \mathbf{x}_i + \mathbf{z}$, shift, rescale, take mod-2, get

$$\mathbf{y}_{\mathsf{CoF}} = [\mathbf{x} + \mathbf{z}] \bmod 2$$

  where $\mathbf{x} = [\sum_i \mathbf{x}_i] \bmod 2 \in \mathcal{C}_{\mathsf{lin}} \subset \{0,1\}^n$
- The channel from $\mathbf{x}$ to $\mathbf{y}_{\mathsf{CoF}}$ is a **BMS with folded Gsn noise**
  $\implies$ Designing $\mathcal{C}_{\mathsf{lin}}$ is a standard coding task
  Normal approximation: $\log|\mathcal{C}_{\mathsf{lin}}| \approx nC - \sqrt{nV}Q^{-1}(\epsilon_{code})$

What is lost in the conversion $\mathbf{y} \mapsto \mathbf{y}_{\mathsf{CoF}}$?

Sum-capacity of $\mathbf{y}$ grows like $\log(T \cdot P)$
Capacity of $\mathbf{y}_{\mathsf{CoF}}$ only grows like $\log(P)$

More on the CoF phase

- $\mathcal{C}_{\text{lin}} \subset \{0,1\}^n$ is a binary linear code (shifted to $\pm\sqrt{P}$)
- Receive $\mathbf{y} = \sum_{i=1}^T \mathbf{x}_i + \mathbf{z}$, shift, rescale, take mod-2, get

$$\mathbf{y}_{\text{CoF}} = [\mathbf{x} + \mathbf{z}] \bmod 2$$

  where $\mathbf{x} = [\sum_i \mathbf{x}_i] \bmod 2 \in \mathcal{C}_{\text{lin}} \subset \{0,1\}^n$
- The channel from $\mathbf{x}$ to $\mathbf{y}_{\text{CoF}}$ is a **BMS with folded Gsn noise**
  $\implies$ Designing $\mathcal{C}_{\text{lin}}$ is a standard coding task
  Normal approximation: $\log|\mathcal{C}_{\text{lin}}| \approx nC - \sqrt{nV}Q^{-1}(\epsilon_{code})$

What is lost in the conversion $\mathbf{y} \mapsto \mathbf{y}_{\text{CoF}}$?

Sum-capacity of $\mathbf{y}$ grows like $\log(T \cdot P)$
Capacity of $\mathbf{y}_{\text{CoF}}$ only grows like $\log(P)$

T-fold ALOHA reduces "power-loss" to $1/T$ instead of $1/K_a$

More on the BAC Phase

$$\mathbf{y}_{\mathsf{BAC}} = \left[\sum_{i=1}^{T} \mathbf{w}_i\right] \bmod 2, \quad \mathbf{w}_1, \ldots, \mathbf{w}_T \in \mathcal{C}_{\mathsf{BAC}}$$

Need to decode a list $\{\mathbf{w}_1, \ldots, \mathbf{w}_T\}$

Symmetric-capacity: $C_{sym} = \frac{1}{T}$

More on the BAC Phase

$$\mathbf{y}_{\mathsf{BAC}} = \left[\sum_{i=1}^{T} \mathbf{w}_i\right] \bmod 2, \quad \mathbf{w}_1, \ldots, \mathbf{w}_T \in \mathcal{C}_{\mathsf{BAC}}$$

Need to decode a list $\{\mathbf{w}_1, \ldots, \mathbf{w}_T\}$

Symmetric-capacity: $C_{sym} = \frac{1}{T}$

How to construct explicit codes?

- Let $H = [\mathbf{h}_1 | \cdots | \mathbf{h}_N]$ be the **parity-check matrix** of a T-error correcting code
- $\Rightarrow$ all T-sums of columns are distinct
- Set $\mathcal{C}_{\mathsf{BAC}} = \{\mathbf{h}_1, \ldots, \mathbf{h}_N\}$
- BCH parity check matrix: $R_{\mathsf{BAC}} = \frac{1}{T}$ (optimal!)
- Encoding: easy (just compute $\alpha, \alpha^3, \cdots, \alpha^{2T-1}$)

More on the BAC Phase

$$\mathbf{y}_{\mathsf{BAC}} = \left[\sum_{i=1}^{T} \mathbf{w}_i\right] \bmod 2, \quad \mathbf{w}_1, \ldots, \mathbf{w}_T \in \mathcal{C}_{\mathsf{BAC}}$$

Need to decode a list $\{\mathbf{w}_1, \ldots, \mathbf{w}_T\}$

Symmetric-capacity: $C_{sym} = \frac{1}{T}$

How to construct explicit codes?

- Let $H = [\mathbf{h}_1 | \cdots | \mathbf{h}_N]$ be the **parity-check matrix** of a T-error correcting code
- $\Rightarrow$ all T-sums of columns are distinct
- Set $\mathcal{C}_{\mathsf{BAC}} = \{\mathbf{h}_1, \ldots, \mathbf{h}_N\}$
- BCH parity check matrix: $R_{\mathsf{BAC}} = \frac{1}{T}$ (optimal!)
- Encoding: easy (just compute $\alpha, \alpha^3, \cdots, \alpha^{2T-1}$)

Problem: decoding complexity of BCH linear in $n = 2^k - 1$

More on the BAC Phase: Decoding BCH

Decoding:

- $\alpha_1, \ldots, \alpha_T \in \mathbb{F}_{2^k}$ are messages
- $\mathbf{y}_{\mathsf{BAC}} = He'$ – syndrome (!) $\Rightarrow$ we know $\sum_i (\alpha_i)^s, s \le 2T$
- *Error locator:* Berlekamp-Massey yields coeffs of

$$\sigma(z) = \prod_{i=1}^{T} (1 + \alpha_i z)$$

- *Find roots of* $\sigma(\cdot)$ e.g. via [Rabin'80]
- *Invert roots:* using the identity $\alpha^{-1} = \alpha^{2^k} - 1$

Total complexity: $\mathcal{O}(kT^2 \log^2(T) \log\log(T))$ operations in $\mathbb{F}_{2^k}$

Spectral Efficiency > 1

The spectral efficiency $\rho = \frac{K_a \cdot k}{n}$ of our scheme is at most R_{lin}
What if $\rho > 1$?

Solution: - work with $p > 2$

- CoF phase requires good linear codes over $\mathbb{F}_p$
- BAC phase can be implemented using $H = [\mathbf{h}_1 | \cdots | \mathbf{h}_n]$ of a $[n = p^s - 1, n - k = 2T]$ Reed-Solomon code over $\mathbb{F}_{p^s}$ with

$$\mathcal{C}_{\text{BAC}} = \{\alpha \mathbf{h}_i \ : \ \alpha \in \mathbb{F}_{p^s} \setminus \{0\}, i = 1, \ldots, p^s - 1\}$$

- Can use nested lattice to achieve the 1.53dB shaping gain
- Drawback: hard to analyze finite blocklength

Approximate performance

Asymptotic optimum: $\left(\frac{E_b}{N_0}\right)^* = \frac{2^{2\rho}-1}{2\rho}$, with $\rho = \frac{K_a \cdot k}{n}$.

Let $L = \frac{K_a}{\alpha T}$ for $\alpha \in (0,1]$ be number of subframes

$P_e \approx \mathbb{P}[T\text{-collision}] = \Pr\left(\text{Binomial}\left(K_a - 1, \frac{\alpha T}{K_a}\right) \geq T\right)$

Linear code rate $R_{\text{lin}} = \frac{\rho}{\alpha}$

$$\begin{aligned}\Delta &= \left(\frac{E_b}{N_0}\right)\text{dB} - \left(\frac{E_b}{N_0}\right)^*\text{dB}\\ &\approx 6\rho\frac{1-\alpha}{\alpha} + 10\log_{10}(\alpha)\end{aligned}$$

T-Collision avoidance loss due to a $1/\alpha$ increase in spectral efficiency

Approximate performance

Asymptotic optimum: $\left(\frac{E_b}{N_0}\right)^* = \frac{2^{2\rho}-1}{2\rho}$, with $\rho = \frac{K_a \cdot k}{n}$.

Let $L = \frac{K_a}{\alpha T}$ for $\alpha \in (0, 1]$ be number of subframes

$P_e \approx \mathbb{P}[T\text{-collision}] = \Pr\left(\text{Binomial}\left(K_a - 1, \frac{\alpha T}{K_a}\right) \geq T\right)$

Linear code rate $R_{\text{lin}} = \frac{\rho}{\alpha}$

$$\Delta = \left(\frac{E_b}{N_0}\right) \text{dB} - \left(\frac{E_b}{N_0}\right)^* \text{dB}$$

$$\approx 6\rho\frac{1-\alpha}{\alpha} + 10\log_{10}(\alpha) + 10\log_{10}(T)$$

CoF loss from the reduction $\mathbf{y} \mapsto \mathbf{y}_{\text{CoF}}$

Approximate performance

Asymptotic optimum: $\left(\frac{E_b}{N_0}\right)^* = \frac{2^{2\rho}-1}{2\rho}$, with $\rho = \frac{K_a \cdot k}{n}$.

Let $L = \frac{K_a}{\alpha T}$ for $\alpha \in (0,1]$ be number of subframes

$P_e \approx \mathbb{P}[T\text{-collision}] = \Pr\left(\mathrm{Binomial}\left(K_a - 1, \frac{\alpha T}{K_a}\right) \ge T\right)$

Linear code rate $R_{\text{lin}} = \frac{\rho}{\alpha}$

$$\Delta = \left(\frac{E_b}{N_0}\right) \mathrm{dB} - \left(\frac{E_b}{N_0}\right)^* \mathrm{dB}$$
$$\approx 6\rho\frac{1-\alpha}{\alpha} + 10\log_{10}(\alpha) + 10\log_{10}(T) - 10\log_{10}(1 - 2^{-2\rho})$$

Loss of $+1$ in computation rate

Approximate performance

Asymptotic optimum: $\left(\frac{E_b}{N_0}\right)^* = \frac{2^{2\rho}-1}{2\rho}$, with $\rho = \frac{K_a \cdot k}{n}$.

Let $L = \frac{K_a}{\alpha T}$ for $\alpha \in (0,1]$ be number of subframes

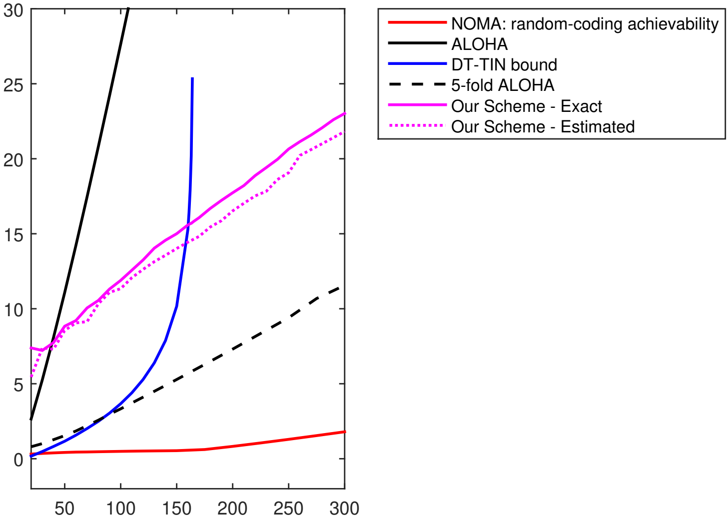
$P_e \approx \mathbb{P}[T\text{-collision}] = \Pr\left(\mathrm{Binomial}\left(K_a - 1, \frac{\alpha T}{K_a}\right) \geq T\right)$

Linear code rate $R_{\text{lin}} = \frac{\rho}{\alpha}$

$$\Delta = \left(\frac{E_b}{N_0}\right)\mathrm{dB} - \left(\frac{E_b}{N_0}\right)^*\mathrm{dB}$$
$$\approx 6\rho\frac{1-\alpha}{\alpha} + 10\log_{10}(\alpha) + 10\log_{10}(T) - 10\log_{10}(1-2^{-2\rho}) + 1.53$$

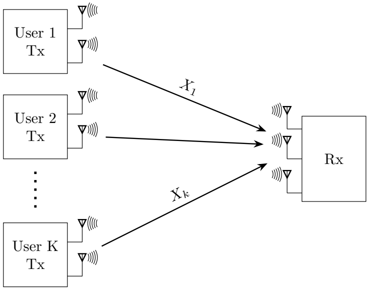
Shaping loss

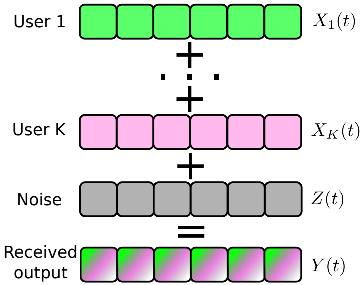
Low-complexity schemes: summary



MAC with random path-loss

K-user GMAC with random **path-loss**



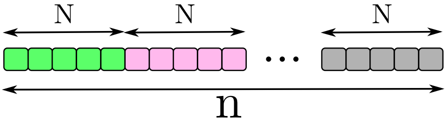


$$Y(t) = H_1 X_1(t) + \cdots + H_K X_K(t) + Z(t)$$

- More realistic model: waveforms added with random gains
- Standard work-around: use pilots
- Impossible without coordination!

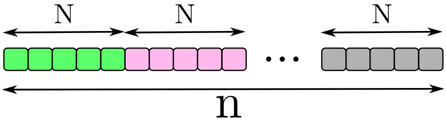
New multi-access protocol (2018): idea

- **Step 1:** Partition entire frame into subframes of length N



New multi-access protocol (2018): idea

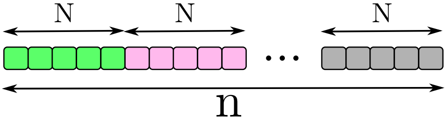
- **Step 1:** Partition entire frame into subframes of length N



- **Step 2:** Each user randomly selects a subframe for communication. Important: K and $\frac{n}{N}$ are chosen so that $> T$-**fold** collisions are improbable.

New multi-access protocol (2018): idea

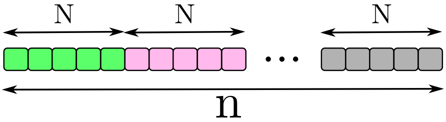
- **Step 1:** Partition entire frame into subframes of length N



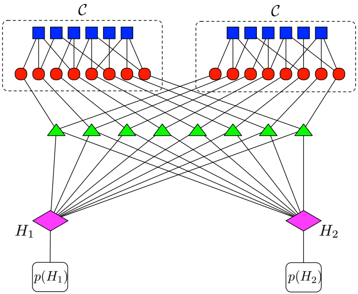
- **Step 2:** Each user randomly selects a subframe for communication.
- **Step 3:** Users encode their data via sparse-graph (LDPC) codes

New multi-access protocol (2018): idea

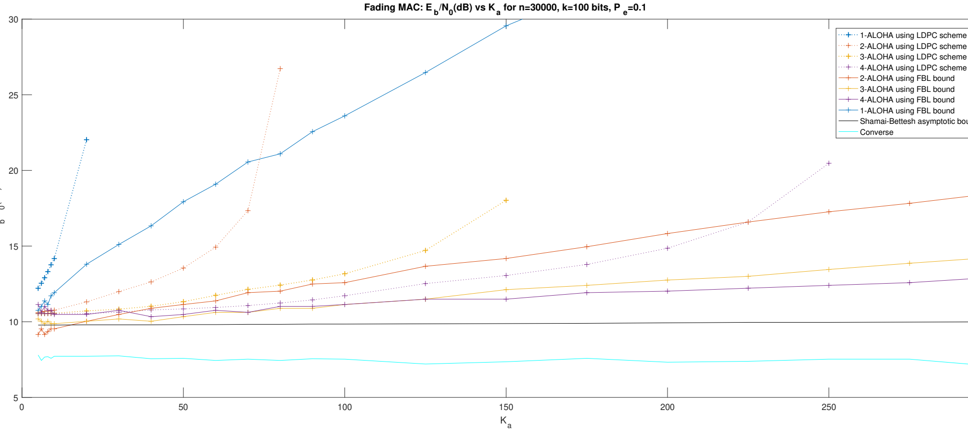
- **Step 1:** Partition entire frame into subframes of length N



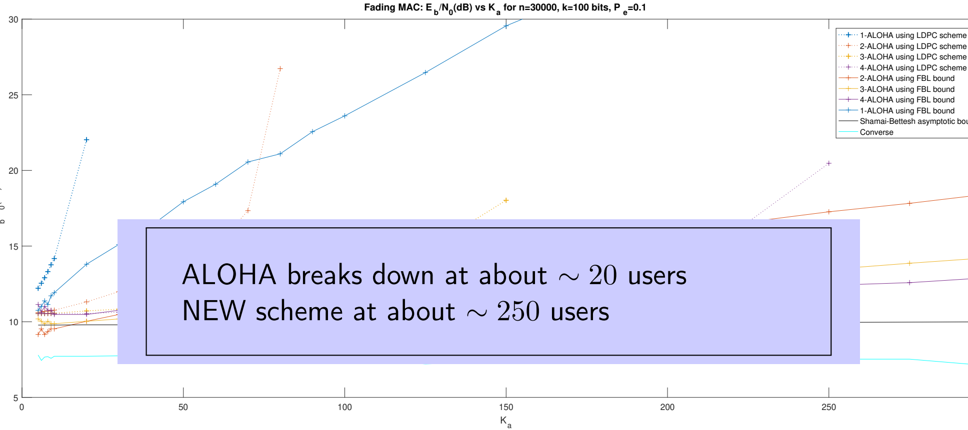
- **Step 2:** Each user randomly selects a subframe for communication.
- **Step 3:** Users encode their data via sparse-graph (LDPC) codes
- **Step 4:** Decoder uses joint Tanner graph (LDPC+LDGM structure) to iteratively decode data and learn the channel gains!



New multi-access protocol (2018): results



New multi-access protocol (2018): results



ALOHA breaks down at about ~ 20 users

NEW scheme at about ~ 250 users

Other ideas for low-complexity schemes

- Work in progress by several groups
  - Narayanan-Chamberland
  - P.-Frolov
  - Durisi-Dalai
  - Popovski-Liva
  - ... (sorry to those I forgot)
- Methods we did not cover:
  - Coded Slotted ALOHA
  - ... including with MPR capability
  - iterative decoding same-codebook LDPCs
  - super-imposed codes
- Problem is even more interesting with fading
  - Random channel gains H_j help distinguish users.
  - With many users, order statistics of H_j's becomes deterministic.

Outline – revisited

Envisioned solution:

- To save battery: sensors sleep all the time, except transmissions.
- ... uncoordinated transmissions.
- ... they wake up, blast the packet, go back to sleep.
- Focus on low-energy (low E_b/N_0)
- Focus on fundamental limits
- ... but with low-complexity solutions (single-user-only decoding).

Outline – revisited

Envisioned solution:

- To save battery: sensors sleep all the time, except transmissions.
- ... uncoordinated transmissions.
- ... they wake up, blast the packet, go back to sleep.
- Focus on low-energy (low E_b/N_0)
- Focus on fundamental limits
- ... but with low-complexity solutions (single-user-only decoding).

Issues we need to understand:

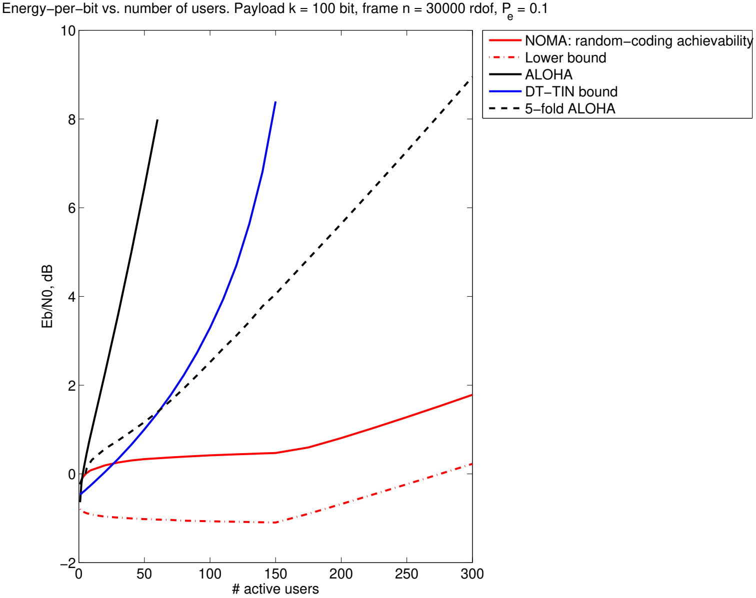
1. packets are short: finite-blocklength (FBL) info theory
2. multiple-access channel: Classical MAC
3. low-complexity MAC: modulation, CDMA, multi-user detection
4. massive random-access: many users, same-codebook codes **(NEW)**

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

$\boxed{\text{Thank you!}}$

Extra: More plots

ALOHA + codes repairing 5-fold collisions



Other schemes...