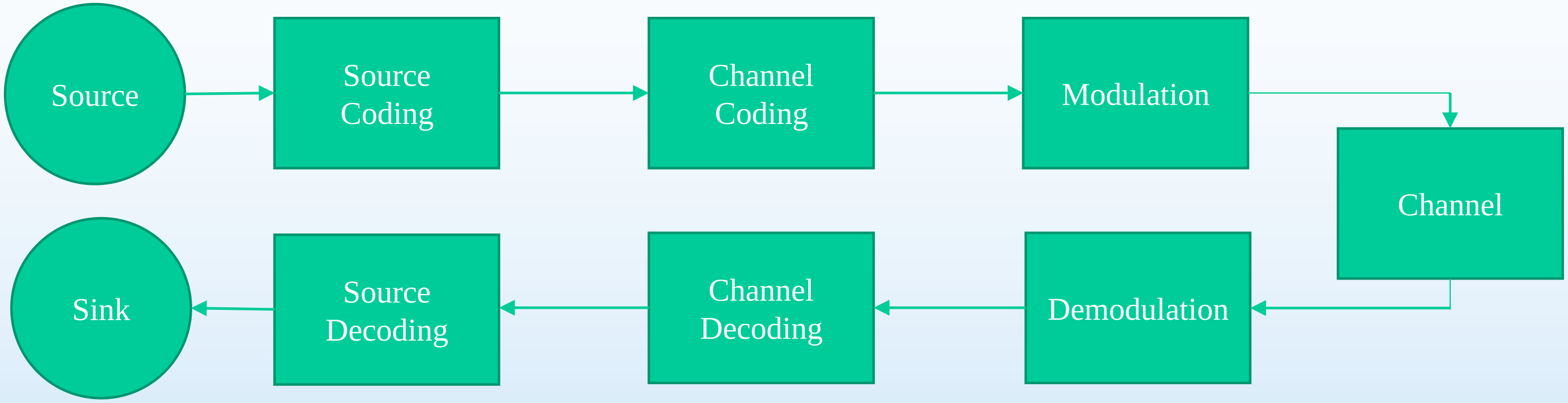
COMPUTER & ELECTRONIC
ENGINEERING

Digital Communication Techniques

Block diagram digital connection



- Source coding: compression.
- Channel coding: Redundancy is added in a systematic way, for protection against errors (coding rate: R_c (information bits/coded bits)).
- Modulation: Operation by which a continuous time signal is associated with the symbols (numbers) emitted by the source, after compression and channel coding.

Digital signal

- Message: discrete succession of numbers the values of which are chosen in set of dimension $M=2^b$. Each value represents b bits.

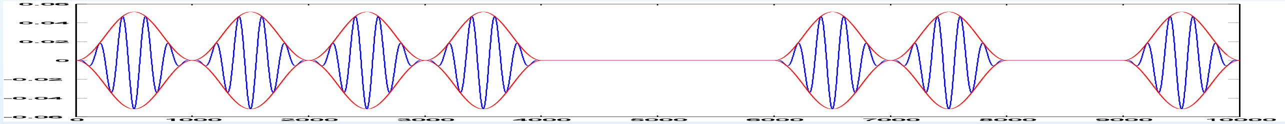
- Modulation: operation that associates each of the possible values with a waveform. A continuous time signal, continuous amplitude is obtained, used to transfer the numeric information.

- A modified signal reaches the receiver (delay, attenuation, distortion, noise, interference) and can be misinterpreted by the receiver (detection/decoding error).

- The performance of a digital transmission system is measured in terms of error rate or, more in general, by error statistic.

Digital Modulator

- It associates to the sequence of numbers to transmit a succession of selected waveforms in a finite dimensional set $M=2^b$, being b a integer. Each waveform is associated with a group of b consecutive bits (*mapping*).



- Modulation operates baseband, if the produced signal has Fourier transform in baseband, or it is defined passband otherwise.

- Parameters:
  - T [s]: Signaling (or symbol) interval, (waveform emission period).
  - $1/T$ [baud] o [symbol/s]: transmission rate in symbols per second.
  - W [Hz]: bandwidth. It is proportional to $1/T$.

A mathematical model

- Consider a set of *N* orthonormal functions, $\psi_1(t), \psi_2(t), \ldots, \psi_N(t)$, .
$$\langle \psi_i \psi_j \rangle = \int_{<T>} \psi_i(t) \psi_j(t) \mathrm{d}t = \delta_{ij}$$
- A generic waveform can be expressed as a linear combination of the orthonormal functions $\Psi_i(t)$, $x(t) = \sum_{i=1}^{N} x_i \psi_i(t)$, being $x_i = \langle x(t) \psi_i(t) \rangle$.
- A generic waveform, $x(t)$, is, therefore, represented by a vector **x**$=(x_1, x_2, \ldots, x_N)$, whose components can be used to determine some quantities of interest.
  - Norm: $\|x(t)\|^2 = \sum_{i=1}^{N} |x_i|^2$.
  - Scalar product: $\langle x(t) y(t) \rangle = \sum_{i=1}^{N} x_i y_i^*$
  - Distance: $d^2(x, y) = \sum_{i=1}^{N} |x_i - y_i|^2$.
- It is $N \leq M$.

Passband Amplitude modulation: N=1, $\psi(t) = g(t) \cos(2\pi f_c t) \sqrt{2/E_g}$

Phase, amplitude and phase modulation: N=2, $\psi_1(t) = g(t) \cos(2\pi f_c t) \sqrt{2/E_g}$

$$\psi_2(t) = g(t) \sin(2\pi f_c t) \sqrt{2/E_g}$$

Linear Passband Modulations

- Amplitude modulation: N=1. $$\psi_1(t)=\sqrt{\frac{2}{E_g}}g(t)\cos(2\pi f_c t)$$

  Complex envelope: $$\tilde{x}(t)=\sum_{n=-\infty}^{\infty}a_n g(t-nT), \qquad a_n\in\{2m-1-M\}, m=1,\ldots,M=2^b$$

- Phase, amplitude and phase modulations: N=2.

$$\psi_1(t)=\sqrt{\frac{2}{E_g}}g(t)\cos(2\pi f_c t) \qquad \psi_2(t)=\sqrt{\frac{2}{E_g}}g(t)\sin(2\pi f_c t)$$

Complex envelope: $$\boxed{\tilde{x}(t)=\sum_{n=-\infty}^{\infty}\alpha_n e^{j\theta_n}g(t-nT),}$$

PSK (M≥4): $$\alpha_n=\alpha=\text{constant}, \quad \theta_n\in\left\{\frac{2\pi}{M}m+\frac{\pi}{M}\right\}, m=0,\ldots,M-1$$

QAM (M=2^{2b}): $$a_n=\alpha_n\cos(\vartheta_n)\in\{2m-1-\sqrt{M}\}, m=1,\ldots,\sqrt{M}=2^b$$

$$b_n=\alpha_n\sin(\vartheta_n)\in\{2m-1-\sqrt{M}\}, m=1,\ldots,\sqrt{M}=2^b$$

Power spectrum

- Linear modulation

  Complex envelope: $v(t)=\sum_{n=-\infty}^{+\infty} a_n g(t-nT,)$ (*)

  where the complex coefficients a_n (information) are random variables belonging to a stationary process, characterized by the mean value $\mu_a = E[a_n]$ and the autocorrelation function $R_a(m-n)=E\left[a_n^* a_m\right]$.

- Autocorrelation function of the complex envelope (cyclostationary process of period T):

$$R_v(t,t+\tau)=E\left[\sum_{n=-\infty}^{+\infty} a_n^* g^*(t-nT)\sum_{m=-\infty}^{+\infty} a_m g(t-mT+\tau)\right]=\sum_n\sum_m R_a(m-n)g^*(t-nT)g(t-mT+\tau)$$

- Wiener-Khintchine theorem: the power spectrum is given by the Fourier transform of the average autocorrelation function $S_v(f)=\mathrm{F}\{\overline{R_v}(\tau)\}$. We have $\overline{R_v}(\tau)=\frac{1}{T}\int_{-T/2}^{T/2} R_v(t,t+\tau)dt=\frac{1}{T}\sum_{m=-\infty}^{\infty} R_a(m)R_g(\tau-mT)$ being $R_g(t)=g(t)\otimes g(-t)$.

(*) The actual signal is $s(t)=\mathrm{Re}\left\{v(t)e^{j2\pi f_c t}\right\}$, being f_c the carrier frequency.

Power Spectrum

- A useful equality : $\int_{-\infty}^{\infty}\sum_n \delta(t-nT)e^{-j2\pi ft}\mathrm{d}t=\sum_n e^{-j2\pi fnT}=\frac{1}{T}\sum_n \delta\left(f-\frac{n}{T}\right)$
- Power spectrum (general expression):

$$\boxed{S_v(f)=\frac{|G(f)|^2}{T}\left(\sigma_a^2+2\,\mathrm{Re}\left[\sum_{m=1}^{\infty}\left(R_a(m)-|\mu_a|^2\right)e^{-j2\pi fmT}\right]\right)+\frac{|\mu_a|^2}{T^2}\sum_{m=-\infty}^{\infty}\left|G\left(\frac{m}{T}\right)\right|^2\delta\left(f-\frac{m}{T}\right)}$$

- Independent data, with non-zero mean value:

$R_a(m)=\begin{cases}|\mu_a|^2+\sigma_a^2 & m=0\\ |\mu_a|^2 & m\neq 0\end{cases}$ being $\sigma_a^2=E\left[|a_m|^2\right]-|\mu_a|^2$ the data variance.

- $$\boxed{S_v(f)=\frac{|G(f)|^2}{T}\sigma_a^2+\frac{|\mu_a|^2}{T^2}\sum_{m=-\infty}^{\infty}\left|G\left(\frac{m}{T}\right)\right|^2\delta\left(f-\frac{m}{T}\right)}$$

- Independent data, with zero mean value

$$R_a(m)=\begin{cases}\sigma_a^2 & m=0\\ 0 & m\neq 0\end{cases}$$

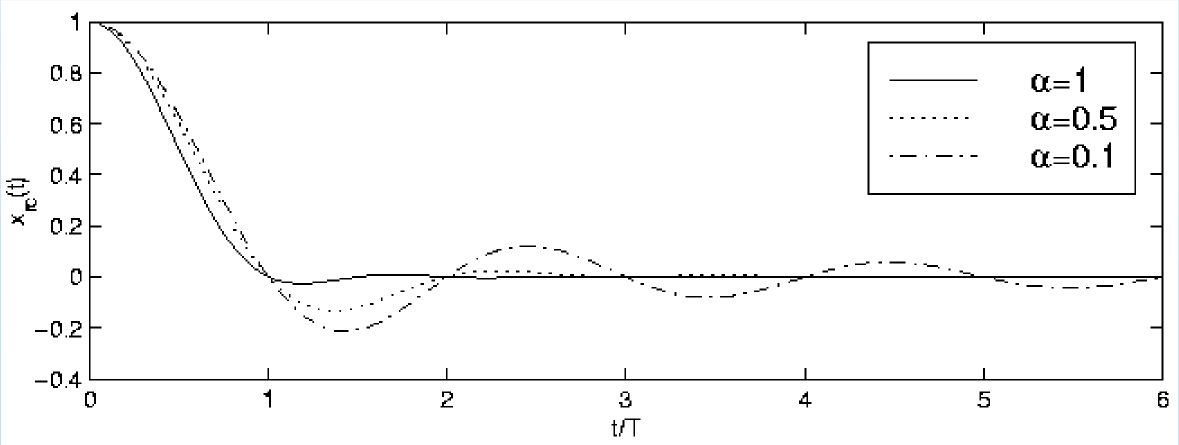
$$\boxed{S_v(f)=\frac{|G(f)|^2}{T}\sigma_a^2}$$

Raised Cosine

- The Raised Cosine pulse allows one to obtain a strictly limited bandwidth.

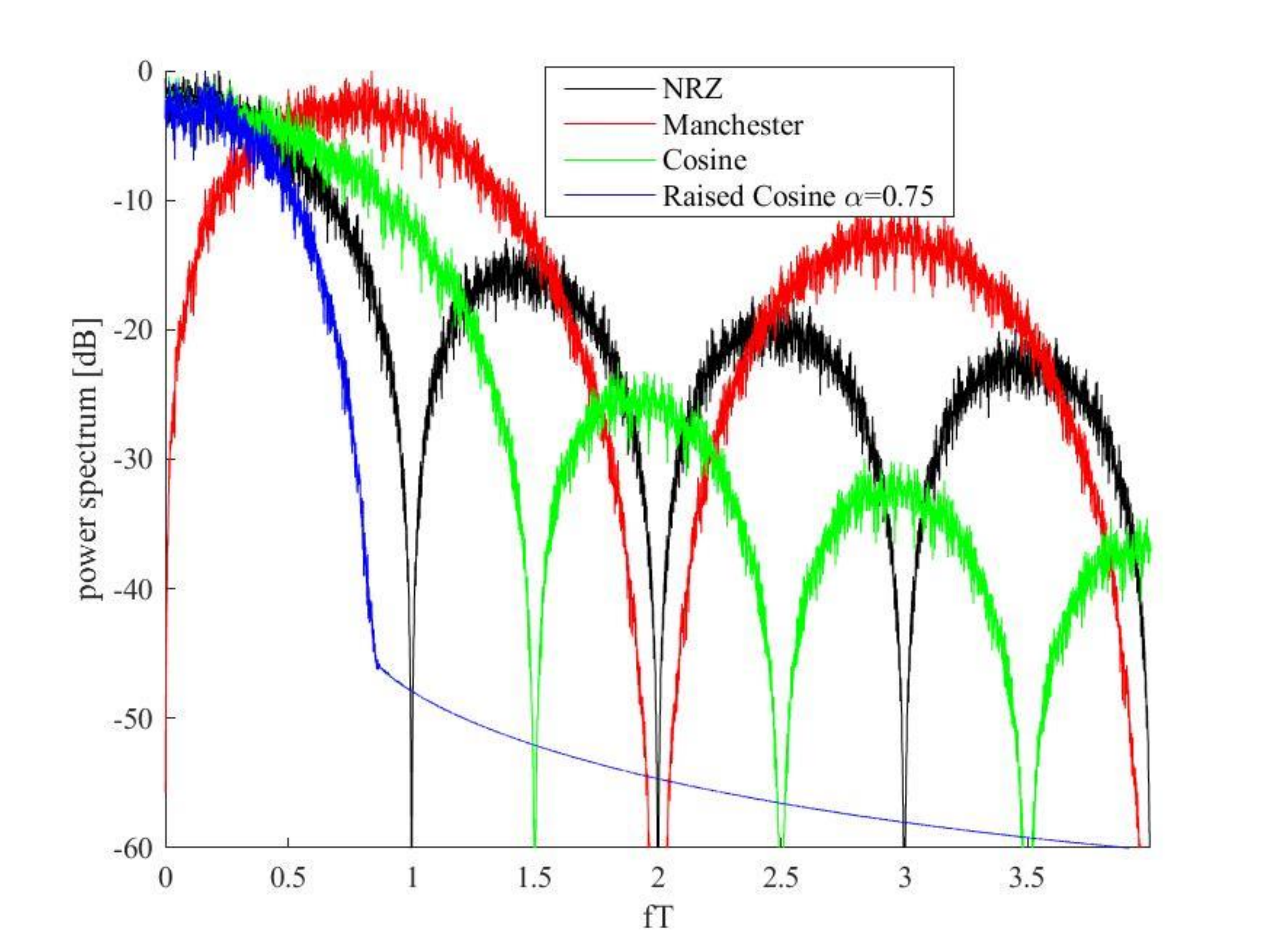
$$H_{RC}(f)=\begin{cases} T & |f|\leq (1-\alpha)/2T \\ \frac{T}{2}\left(1+\cos\left(\pi\frac{T}{\alpha}\left(|f|-\frac{1-\alpha}{2T}\right)\right)\right) & (1-\alpha)/2T\leq |f|\leq (1+\alpha)/2T \\ 0 & |f|\geq (1+\alpha)/2T \end{cases}$$

- As α increases, the bandwidth increases, but also the speed with which the impulse response is damped increases (and therefore the problems arising from imperfect synchronism decrease). It is commonly adopted $0.5\leq\alpha\leq 1$.



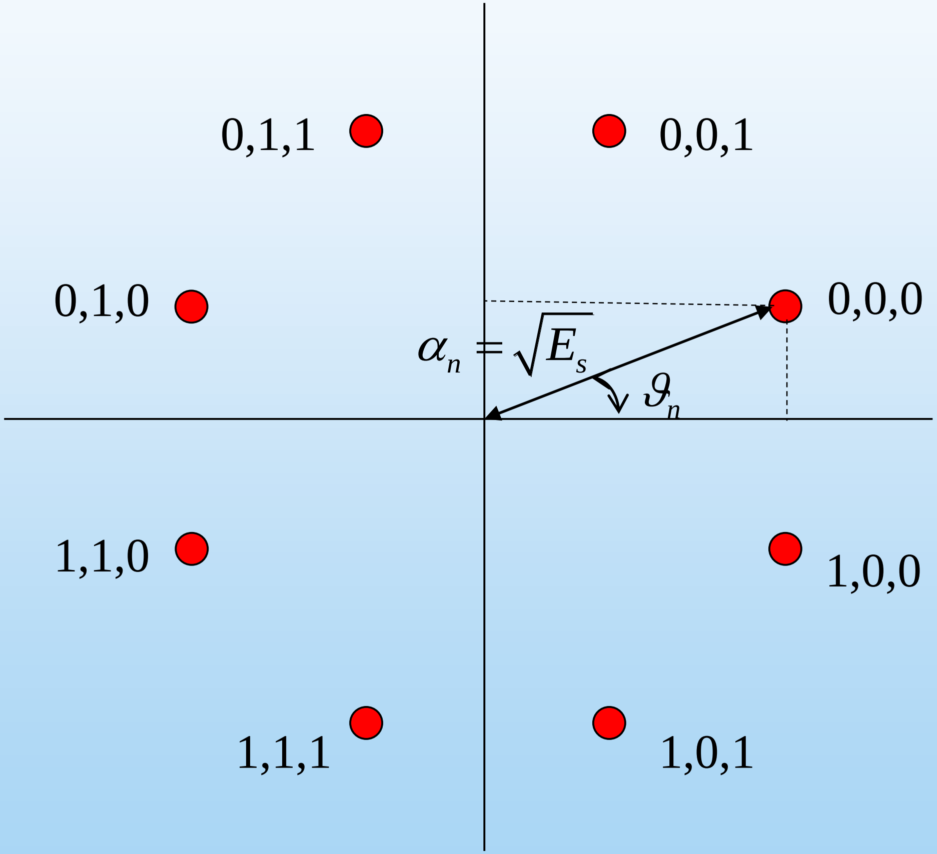
$$h_{RC}(t)=\frac{\sin(\pi t/T)}{\pi t/T}\frac{\cos(\alpha\pi t/T)}{1-(2\pi t/T)^2}$$

Power spectra



Phase Shift Keying example (M-PSK)

- Scattering diagram (8-PSK, with Gray *mapping*)

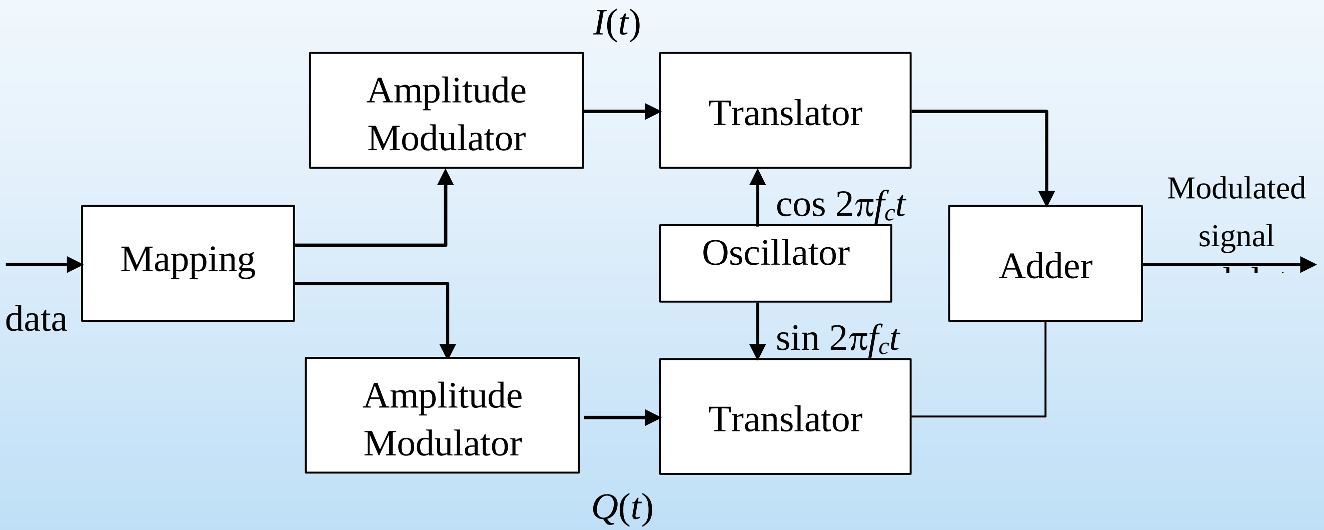


Gray mapping

Bit					Ordine						Ordine
0	0	0	0	0	0	1	1	0	0	0	16
0	0	0	0	1	1	1	1	0	0	1	17
0	0	0	1	1	2	1	1	0	1	1	18
0	0	0	1	0	3	1	1	0	1	0	19
0	0	1	1	0	4	1	1	1	1	0	20
0	0	1	1	1	5	1	1	1	1	1	21
0	0	1	0	1	6	1	1	1	0	1	22
0	0	1	0	0	7	1	1	1	0	0	23
0	1	1	0	0	8	1	0	1	0	0	24
0	1	1	0	1	9	1	0	1	0	1	25
0	1	1	1	1	10	1	0	1	1	1	26
0	1	1	1	0	11	1	0	1	1	0	27
0	1	0	1	0	12	1	0	0	1	0	28
0	1	0	1	1	13	1	0	0	1	1	29
0	1	0	0	1	14	1	0	0	0	1	30
0	1	0	0	0	15	1	0	0	0	0	31

Phase, Amplitude-Phase modulator

- Modulator PSK, QAM



- The phase component, $I(t)$, and the quadrature components, $Q(t)$, are baseband, amplitude modulated signals.
- f_c is the carrier frequency (specified by the standard).
- Complex envelope: $v(t)=I(t)+jQ(t)=\sum_{n=-\infty}^{+\infty} a_n g(t-nT)$

Correlator (coherent) demodulator

- Determines the received vector components.
  Assume to transmit, in a generic signaling interval of duration *T* the waveform $s_m(t)$, and to receive $r(t)= s_m(t)+ n(t)$, being $n(t)$ the receiver noise (Additive White Gaussain AWGN model), with power N_0W watts. The correlator evaluates:

$$r_n = \langle r(t)\psi_n(t)\rangle = \int_0^T s_m(t)\psi_n(t)dt + \int_0^T n(t)\psi_n(t)dt = s_{mn} + n_n, \ \ n = 1,\dots, N$$

- The noise vector components, n_n, are Gaussian, independent, zero mean random variables, having variance $N_0/2$.

$$E[n_k n_h] = E\left[\int_0^T n(t)\psi_k(t)dt \int_0^T n(t)\psi_h(t)dt\right] = \int_0^T\int_0^T E[n(t)n(\tau)]\psi_k(t)\psi_h(\tau)dtd\tau = \frac{N_0}{2}\delta_{hk}$$

- Therefore, the received vector components, r_n, are Gaussian, independent, random variables with mean s_{mn} (corresponding to the transmitted symbol vector components), and variance $N_0/2$

$$f(\mathbf{r}|\mathbf{s}_m) = \frac{1}{(\pi N_0)^{N/2}} \exp\left(-\frac{1}{N_0}\sum_{n=1}^{N}(r_n - s_{mn})^2\right)$$

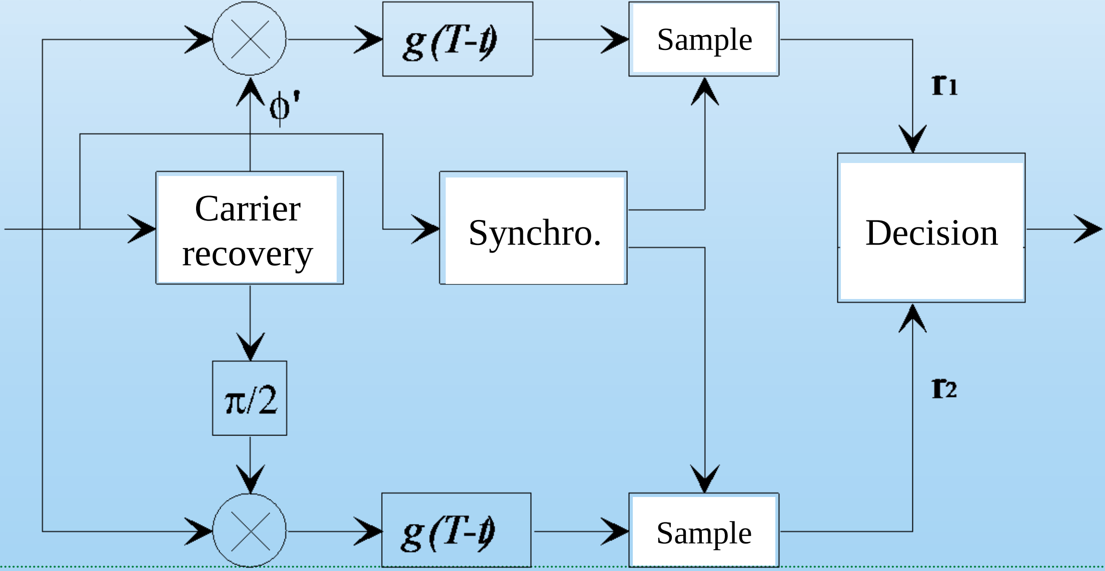
Matched filter demodulator

- Coherent matched filter demodulator

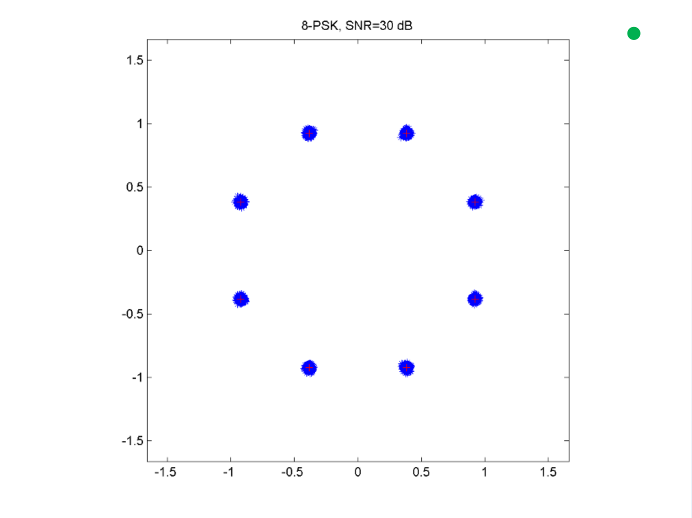
The channel introduces delay, t_d. Neglecting the effects of the distortion introduced by the channel and of the noise introduced by the receiver we have

$$x_R(t) = x_T(t - t_d) = \mathrm{Re}\left\{ \sum_{n=-\infty}^{\infty} \alpha_n \mathrm{e}^{j\theta_n} g(t - t_d - nT) e^{j2\pi f_c (t - t_d)} \right\},$$

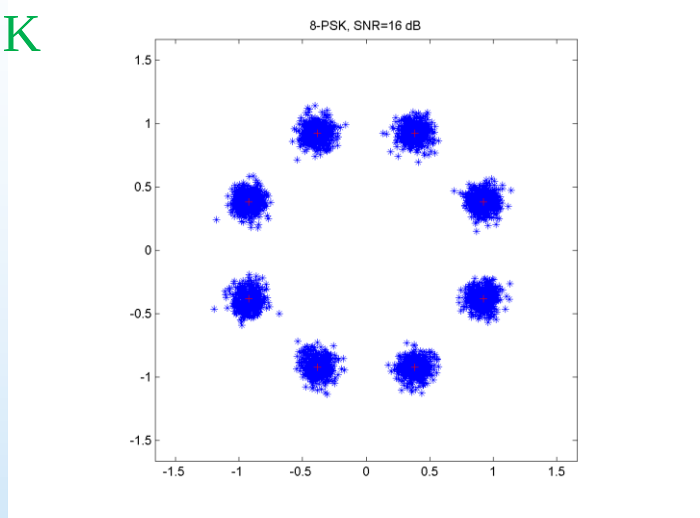
therefore the delay introduces a phase shift $\phi = 2\pi f_c t_d$ which must be compensated.

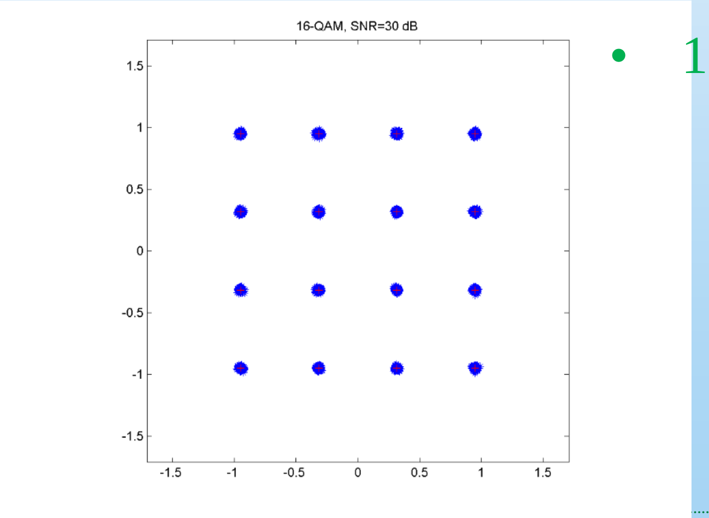


Received scattering diagrams

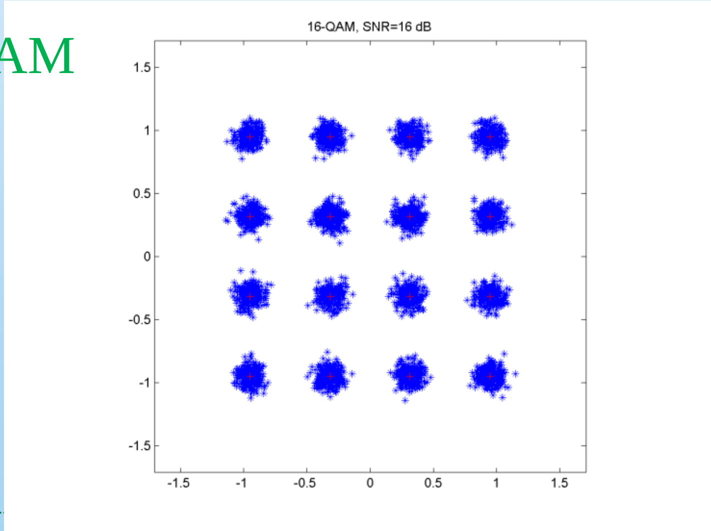


- 8 PSK





- 16 QAM



Hard Detection Theory

- The N-dimensional vector space to which $\mathbf{r}$ belongs, it is divided into M disjointed regions, A_m (Voronoi regions).

- Correct decision probability:

$$p_c = \sum_m p(\mathbf{r} \in A_m | \mathbf{s}_m) p(\mathbf{s}_m) = \sum_m p(\mathbf{s}_m) \int_{\mathbf{r} \in A_m} f(\mathbf{r} | \mathbf{s}_m) d\mathbf{r}$$

- MAP criterion: $\mathbf{r} \in A_m : m = \arg\max_{j} \left(f(\mathbf{r} | \mathbf{s}_j) p(\mathbf{s}_j) \right)$

- Binary choice $r \in A_0 : f(r|s_0) p(s_0) > f(r|s_1) p(s_1)$, equivalent to :
(the relationship between the conditional probabilities is called the likelihood ratio).

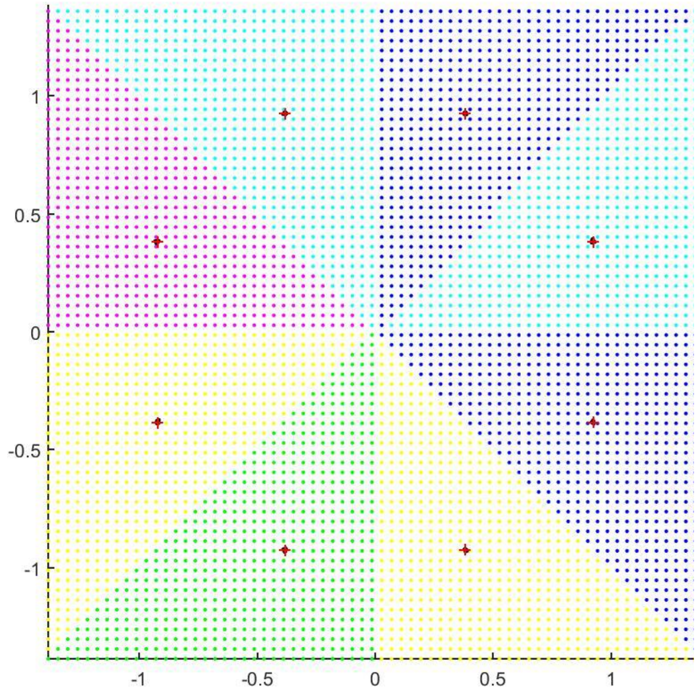
$$r \in A_0 : \frac{f(r|s_1)}{f(r|s_0)} > \frac{p(s_0)}{p(s_1)}$$

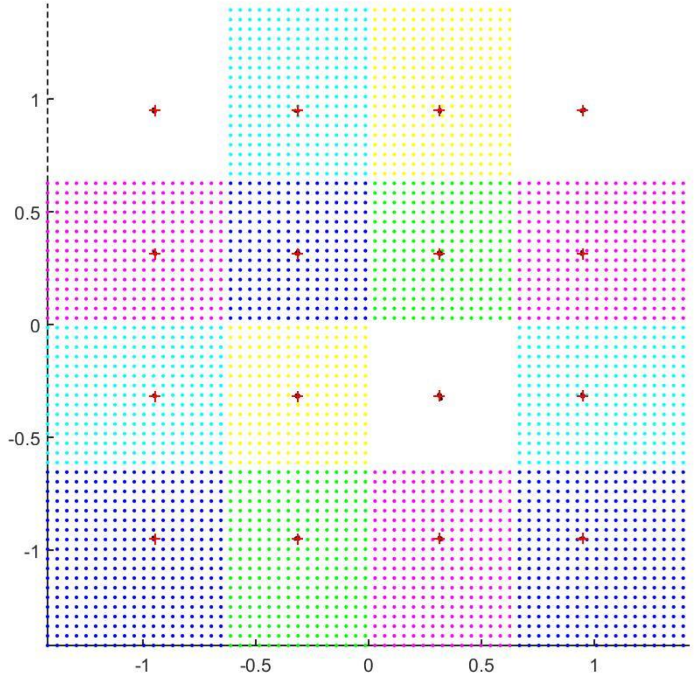
- If $p(s_0)=p(s_1)$, or if the two probabilities are not known, the ML (Maximum Likelihood or Minimum Distance) criterion is adopted:

$$r \in A_0 : \frac{f(r|s_0^{\cdot})}{f(r|s_1)} > 1$$

Minimum distance detection regions

- 8-PSK 16 QAM





Hard detection performance (AWGN)

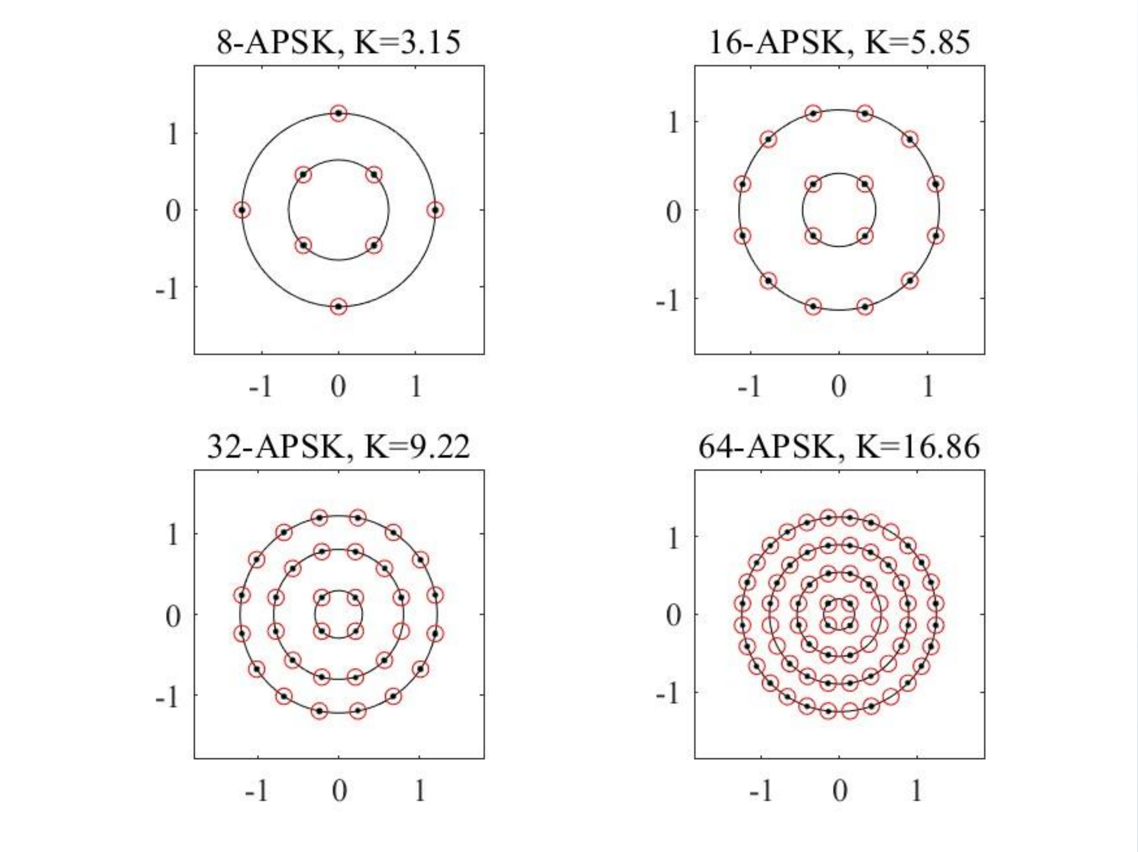
- Binary antipodal modulation (BPSK): $P_e = Q\left(\sqrt{2R_c \frac{E_b}{N_0}}\right) = \frac{1}{2}\mathrm{erfc}\left(\sqrt{R_c \frac{E_b}{N_0}}\right)$
- M-ASK: $p_E = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{R_c \log_2 M}{M^2-1}\frac{E_b}{N_0}}\right)$ $\quad W_{\min} = f_N = \frac{1}{2T}$
- M-QAM: $p_{E_{M\cdot\mathrm{QAM}}} = 1 - p_{C_{M\cdot\mathrm{QAM}}} \approx 2 p_{E_{\sqrt{M}-\mathrm{ASK}}} < 4Q\left(\sqrt{\frac{3R_c \log_2 M}{M-1}\frac{E_b}{N_0}}\right)$

$W_{\min} = \frac{1}{T},\ r_{\max} = \log_2 M = b$

- M-PSK $P_{E_{M-\mathrm{PSK}}} \approx 2Q\left(\sqrt{2R_c \log_2 M \cdot E_b/N_0}\,\sin(\pi/M)\right)$
- M-FSK: $p_E \le (M-1)Q\left(\sqrt{\frac{E_s}{N_0}}\right) = (M-1)Q\left(\sqrt{R_c \frac{E_b}{N_0}\log_2 M}\right)$ $\quad W_{\min} = \frac{M}{2T},\ r_{\max} = \frac{b}{2^{b-1}}$

M-APSK

- Define R_i the radium of the i-th circle (square root of energy) and δ_{ii} the distance between two adjacent points on the same energy level.
- Constellation examples



4-12 APSK

$$d_{\min} = \delta_{11} = \delta_{22} = R_1\sqrt{2}$$

$$R_2 = R_1 \Big/ \left(\sqrt{2}\sin(\pi/12)\right)$$

$$E_s = KR_1^2$$

$$K = \frac{1}{4}\left(1+\frac{3}{2\sin^2(\pi/12)}\right) \approx 5.85$$

$$p_E \approx 2Q\left(\sqrt{\frac{4R_c}{K}\frac{E_b}{N_0}}\right)$$

16-QAM comparison

2 energy levels instead of 3; the required average energy is 0.68 dB higher.

A comparison

- E_b/N_0 and E_{av}/N_0 required for obtaining a symbol error rate P_e=10^{-6}, as a function of the number of bits b, for different modulations with relevant energy levels /observe that, for PSK we have a single energy level). For all the modulation in the table r_{max}=*b*.

b	PSK		QAM			APSK		
	E_b/N_0	E_{av}/N_0	E_b/N_0	E_{av}/N_0	Levels	E_b/N_0	E_{av}/N_0	Levels
1	10.53	10.53						
2	10.78	13.78	10.78	13.79	1			
3	14.36	19.13						
4	18.96	24.98	15.00	21.02	3	15.68	21.70	2
5	23.97	30.96				17.65	24.64	3
6	29.19	36.97	19.47	27.25	6	20.28	28.06	4

Shannon's theorem on capacity

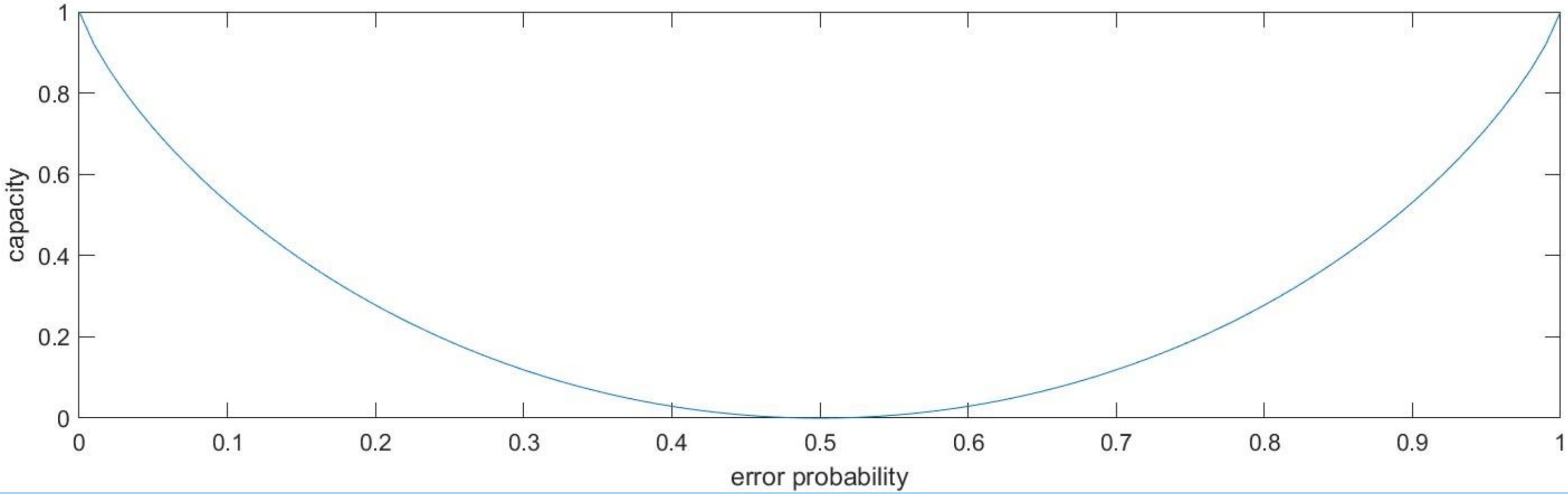
- Channel capacity, C, limits the maximum rate, R (in information bits per channel use), at which information can be reliably transmitted over the channel. Consider an encoded source that emits information at a rate R.
- Negative statement: if $R>C$, considered a sufficiently long sequence of bits of length N, $M=2^{NR}$ equiprobable messages are obtained. Whichever way the associated waveforms are chosen, the probability of error tends towards 1.
- Positive statement: if $R<C$, considered a sufficiently high sequence of bits of length N, it is possible to identify a set of waveforms that allows obtaining an arbitrarily small error probability.
- Alternative formulation: If $R<C$, given ε, there exists a code of sufficiently long length for which $p_e< \varepsilon$, p_e being the error probability conditioned on the transmission of the i-th message.

Hard capacity: symmetric binary channel

- Binary discrete input (symbols ‘0_T’ e ‘1_T’, with probability *p* and 1-*p*).
- Discrete binary output (‘0_R’ e ‘1_R’) (hard detection).
- Error probability: ε=P(‘0_R’| ‘1_T’)=P(‘1_R’| ‘0_T’).

$$\boxed{C = 1 + \varepsilon \log_2 \varepsilon + (1-\varepsilon)\log_2(1-\varepsilon)}$$

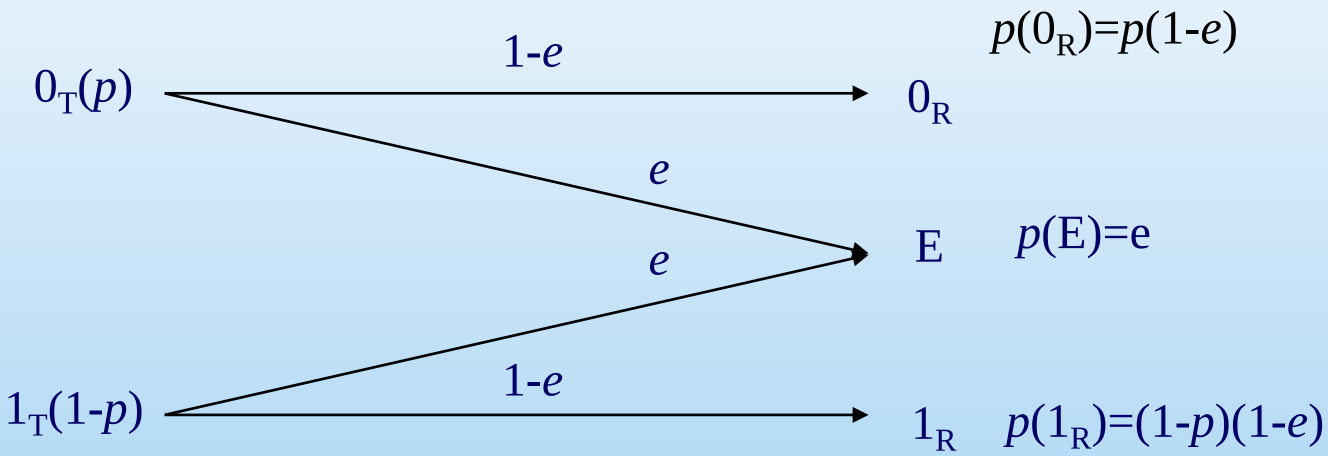
Antipodal binary modulation: $\varepsilon = Q\left(\sqrt{2\frac{E_s}{N_0}}\right)$ being $Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^{\infty} \exp\left(-\frac{u^2}{2}\right)\mathrm{d}u$



The Erasure Channel

- Binary discrete input (symbols ‘0_T’ e ‘1_T’, with probability *p* and 1-*p*).

- An input value may be received correctly or it may be erased with probability e.



- Capacity:

$$C = 1-e \; \frac{\text{information bits}}{\text{channel use}}$$

Soft detection

- Square root energy signal space: $f\left(\mathbf{r}\middle|\mathbf{s}_j\right)=\frac{1}{\left(\pi N_0\right)^{N/2}}\exp\left(-\frac{1}{N_0}\sum_{h=1}^{N}\left(r_h-s_{jh}\right)^2\right)$
-
- Change of variable: $y_h=\frac{r_h}{\sqrt{N_0}},\qquad x_{jh}=\frac{s_{jh}}{\sqrt{N_0}}$
- Square root SNR signal space: $f\left(\mathbf{y}\middle|\mathbf{x}_j\right)=\frac{1}{\left(\pi\right)^{N/2}}\exp\left(-\sum_{h=1}^{N}\left(y_h-x_{jh}\right)^2\right)$
- Log-Likelihood Ratio antipodal binary modulation: (to be used as the input of the channel decoder) $\log\frac{f\left(y\middle|x_0\right)}{f\left(y\middle|x_1\right)}=4y\sqrt{\frac{E_s}{N_0}}$

Soft demodulation

- Pragmatic approach
- Consider a one(two)-dimensional modulation with b bits, being M=2^{b}.
- Let **r** be the value of the observed vector. The likelihood ratio of the i-th bit, with i=1,…,b is given by:

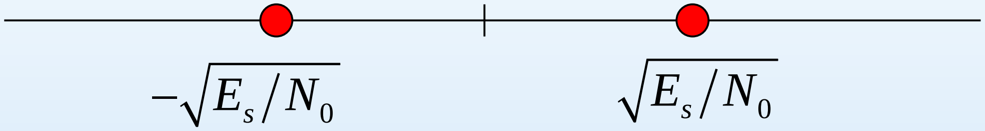
$$\Lambda(x_i) = \log\left(\frac{P[\mathbf{r}|x_i = 1]}{P[\mathbf{r}|x_i = 0]}\right).$$

- Applying Bayes, in the hypothesis of equiprobability of the sequences of b bits present on the channel, let $\mathbf{s}_{j,1}$ be the j-th sequence with i-th bit equal to 1, and $\mathbf{s}_{j,0}$ the one with i-th bit equal to 0, we obtain:

$$\Lambda(x_i) = \log\left(\frac{\sum_{j=1}^{2^{b-1}} \exp\left(-\frac{\left\|\mathbf{r}-\mathbf{s}_{j,1}\right\|^2}{2\sigma^2}\right)}{\sum_{j=1}^{2^{b-1}} \exp\left(-\frac{\left\|\mathbf{r}-\mathbf{s}_{j,0}\right\|^2}{2\sigma^2}\right)}\right).$$

Antipodal modulation soft capacity

- Consider now a non-quantized output, and the Square root SNR signal space.



$$f(x)=\frac{1}{2\sqrt{\pi}}\exp\left(-\left(x-\sqrt{E_s/N_0}\right)^2\right)+\frac{1}{2\sqrt{\pi}}\exp\left(-\left(x+\sqrt{E_s/N_0}\right)^2\right)$$
$$=\frac{1}{\sqrt{\pi}}\exp\left(-x^2-E_s/N_0\right)\cosh\left(2x\sqrt{E_s/N_0}\right).$$

$$f(x|s)=\frac{1}{\sqrt{\pi}}\exp\left(-x^2\right)$$

$$C=-\int_{-\infty}^{\infty} f(x)\log_2 f(x)\mathrm{d}x-\frac{1}{2}\log_2(\pi e)\quad\frac{\text{information bits}}{\text{channel use}}$$

BPSK-QPSK soft capacity

- BPSK

$$C = 1 - \exp\left(-a_1 \left(\frac{E_{av}}{N_0} \right)^{a_2} + a_3 \right) \quad \frac{\text{bit infor.}}{\text{simbolo}}$$

$$a_1 \approx 1.286,\ a_2 \approx 0.931,\ a_3 \approx 0.010$$

- QPSK

$$C = 2\left[1 - \exp\left(-a_1 \left(\frac{1}{2}\frac{E_{av}}{N_0} \right)^{a_2} + a_3 \right) \right] \quad \frac{\text{bit infor.}}{\text{simbolo}}$$

F. Babich, A. Soranzo, and F. Vatta, “Useful mathematical tools for capacity approaching codes design,” IEEE Commun. Lett., vol. 21, no. 9, pp. 1949–1952, Sep. 2017.

AWGN Shannon bound

- Shannon Theorem: for a *N*-dimensional channel, for a reliable transmission, the coded modulation rate (for every modulation) is bounded as follows:

$$R < C = \frac{N}{2}\log_2\left(1+\frac{2\left(E_s/N\right)}{N_0}\right) \quad \frac{\text{information bits}}{\text{channel use}}$$

- If follows that the source rate, R_s=R/T, is bounded as follows:

$$R_s < \frac{C}{T} = \frac{N}{2T}\log_2\left(1+\frac{2T}{N}\frac{RE_b}{TN_0}\right) = W\log_2\left(1+\frac{1}{W}R_s\frac{E_b}{N_0}\right)\frac{\text{information bits}}{\text{second}}$$

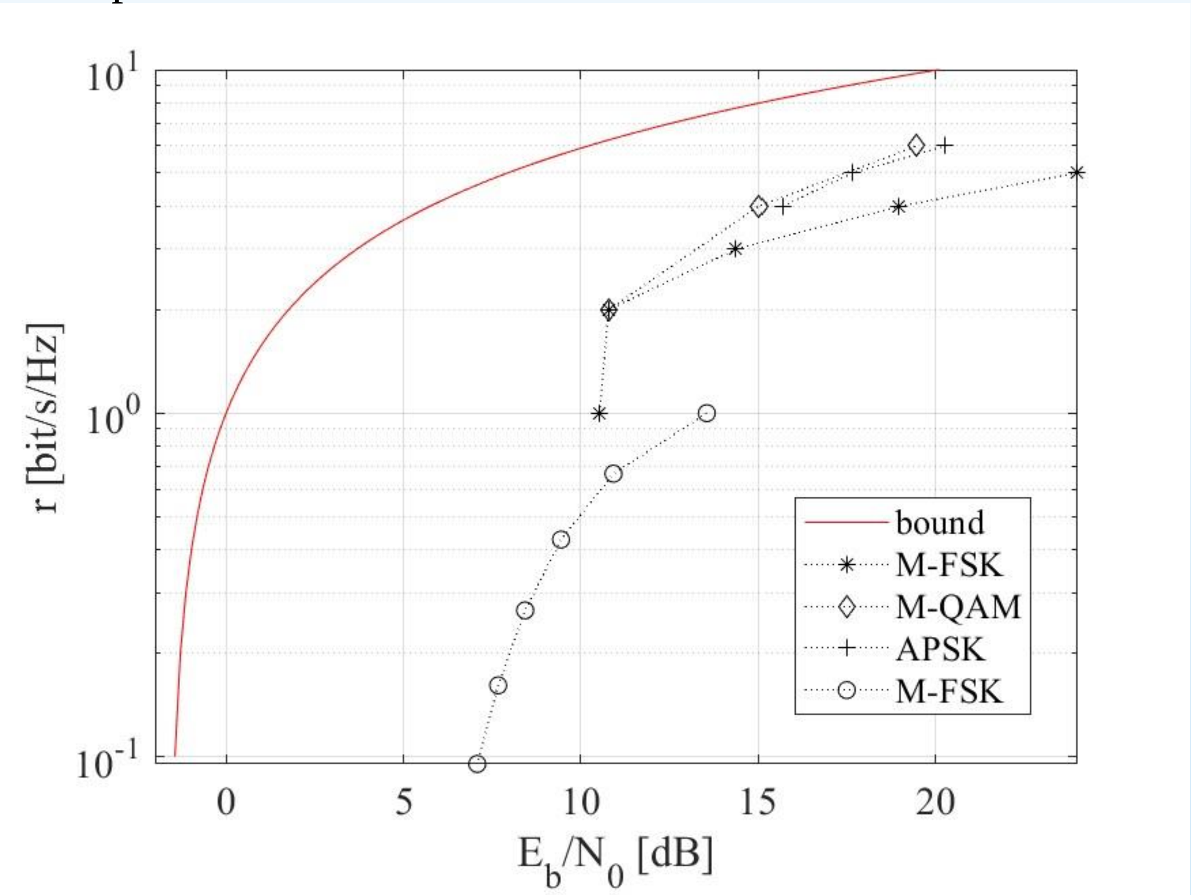
- The spectral efficiency is bounded as follows

$$\boxed{r_{\max} = \frac{R_{s_{\max}}}{W} = \log_2\left(1+r_{\max}\frac{E_b}{N_0}\right)\frac{\text{information bits}}{\text{second}\cdot\text{Hz}}}$$

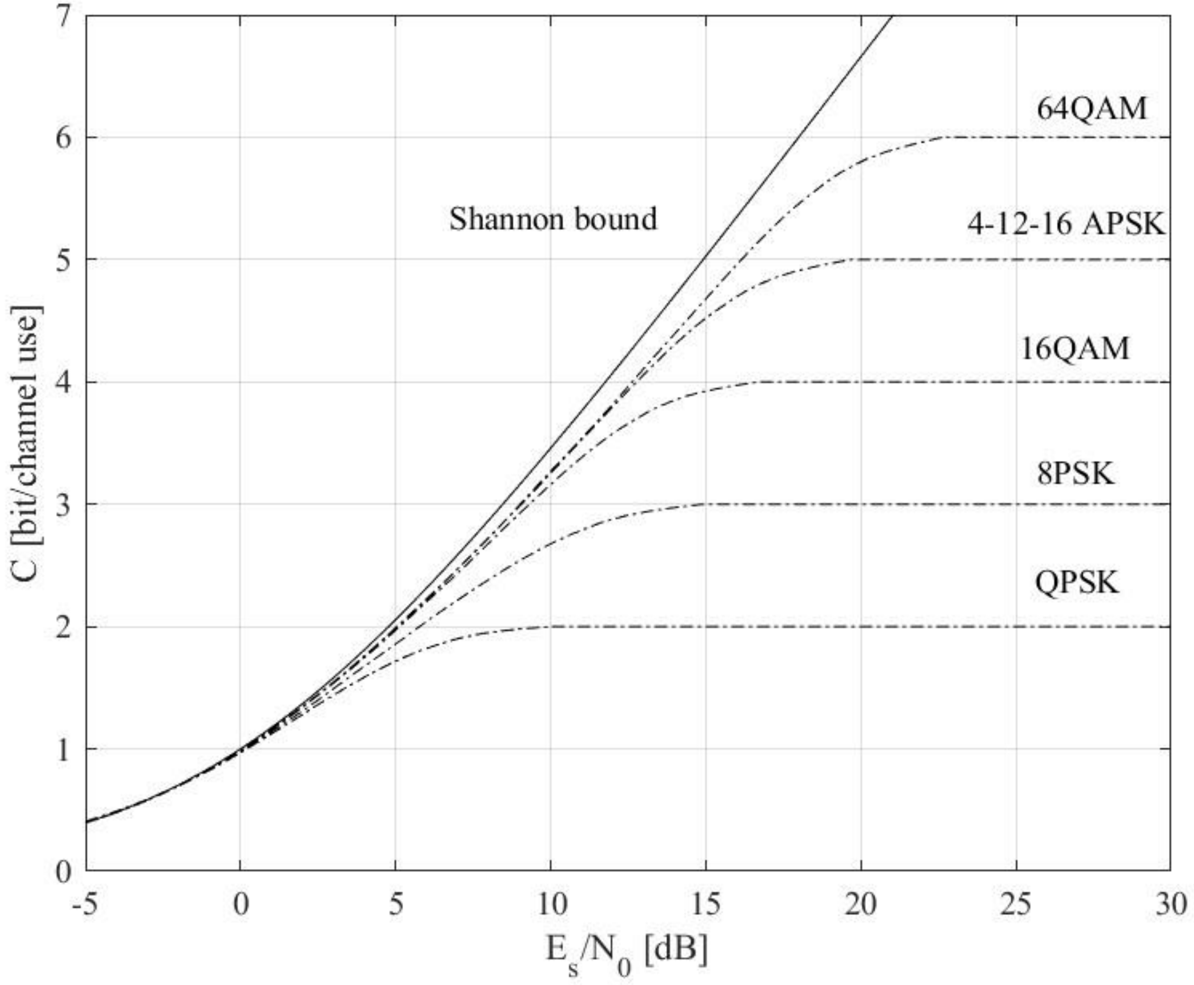
- From which we obtain $\boxed{\left(\frac{E_b}{N_0}\right)_{\min} = \frac{2^r-1}{r}}$ (Shannon bound)

Comparison with Shannon bound

- E_b/N_0 required for obtaining a symbol error rate P_e=10^{-6}. With hard detection without channel coding there is a large gap between the actual and the achievable performance.



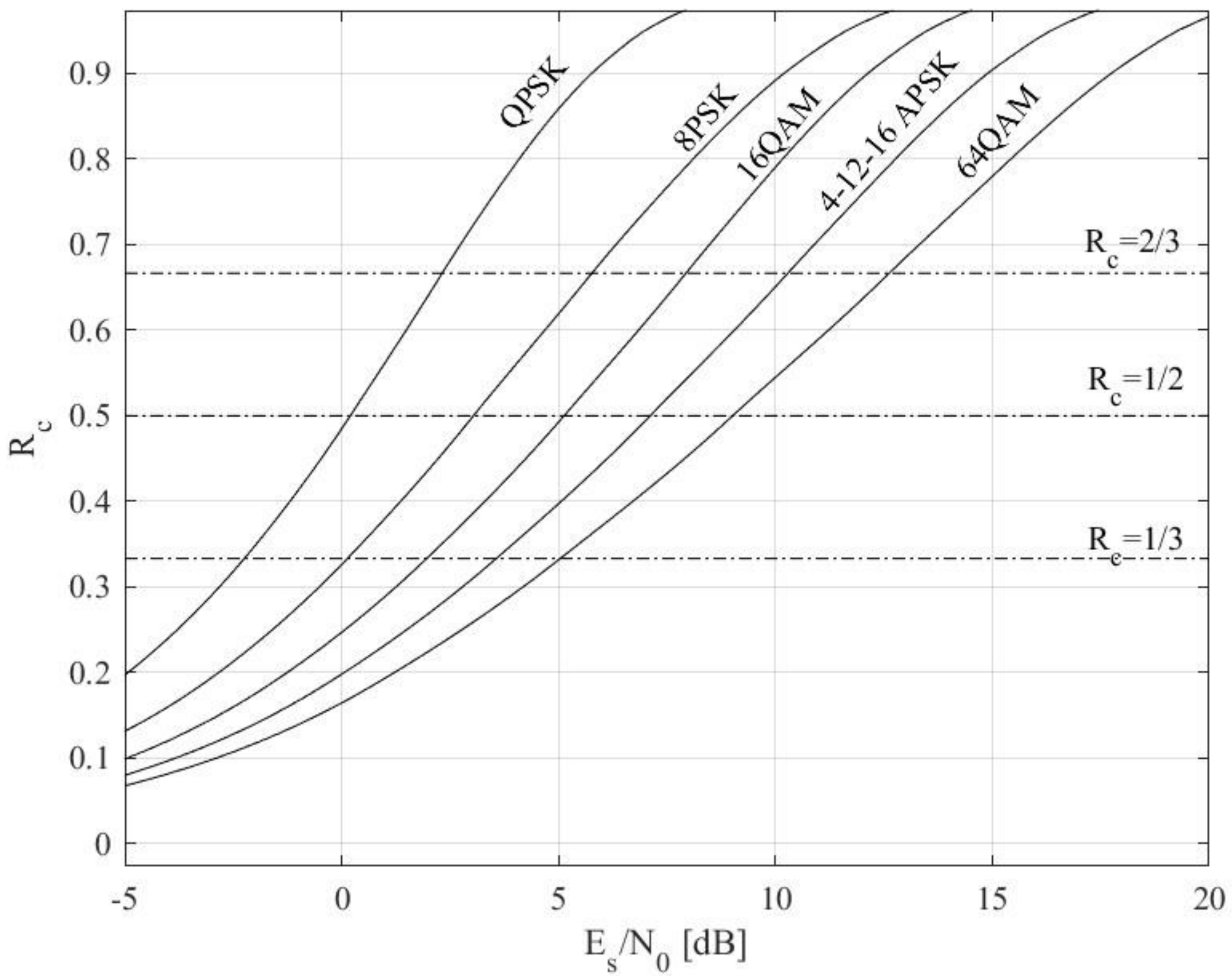
Capacity of bidimensional modulations



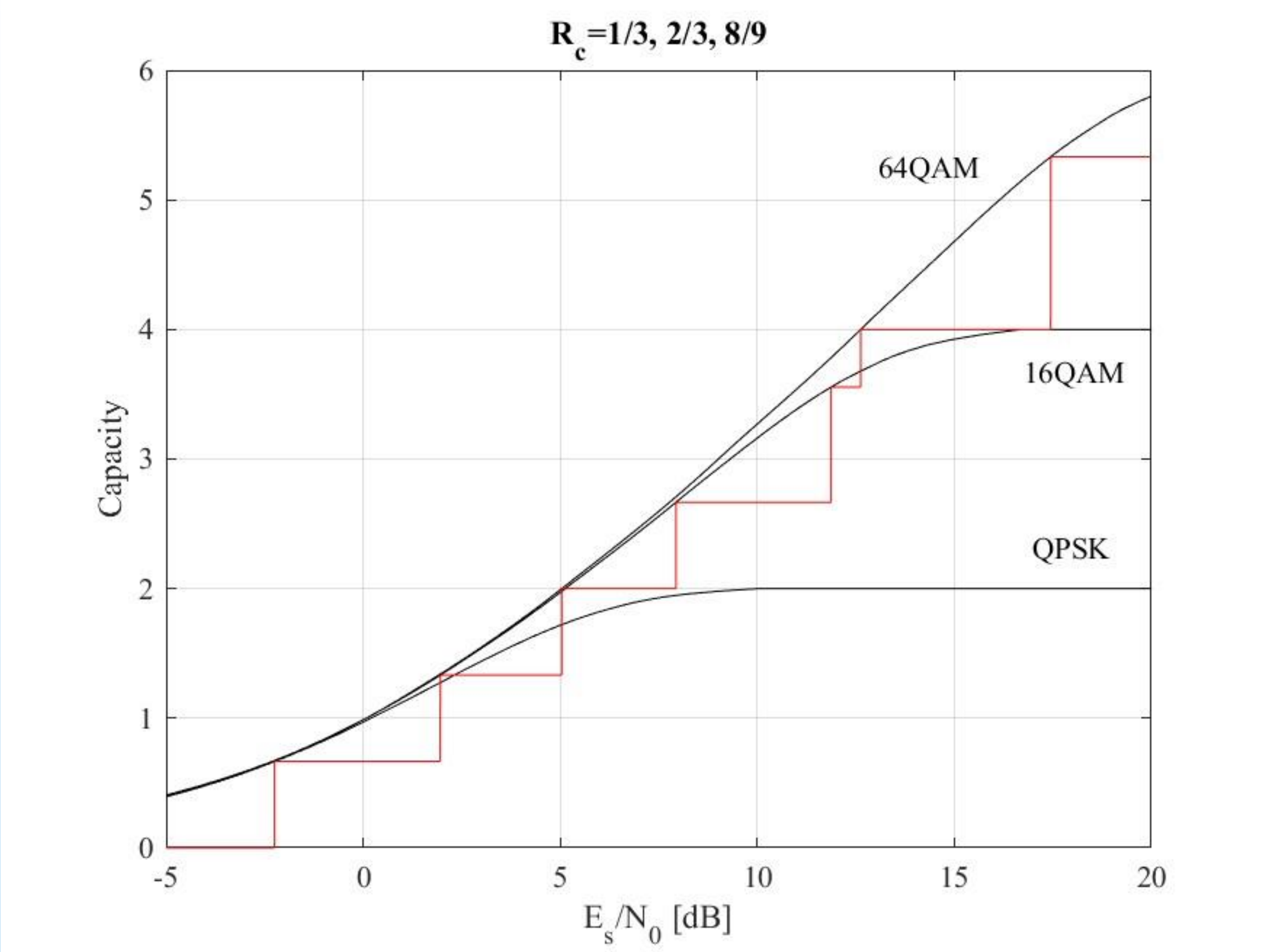
Channel encoding

- Channel encoding introduces redundancy to protect the digital signal from errors.
- It may be used to detect the presence of errors in a message (associated with retransmission algorithms, Automatic Repeat reQuest – ARQ).
- It may be also be used to prevent the occurrence of errors (Forward Error Correction - FEC).
- Encoding: k compressed source bits are associated with n bits for transmission on the channel, with $n>k$; redundancy is thus systematically introduced that can be used in reception to detect and/or prevent errors.
- Code rate: $R_c=k/n<1$. Let E_b be the energy used to transmit a bit of information. Therefore, the energy used to transmit a bit on the channel is equal to $E_b R_c$.
- From the bound on $R=R_c b=R_c\log_2(M)$, we may determine a bound on R_c.

Determining required Code Rate

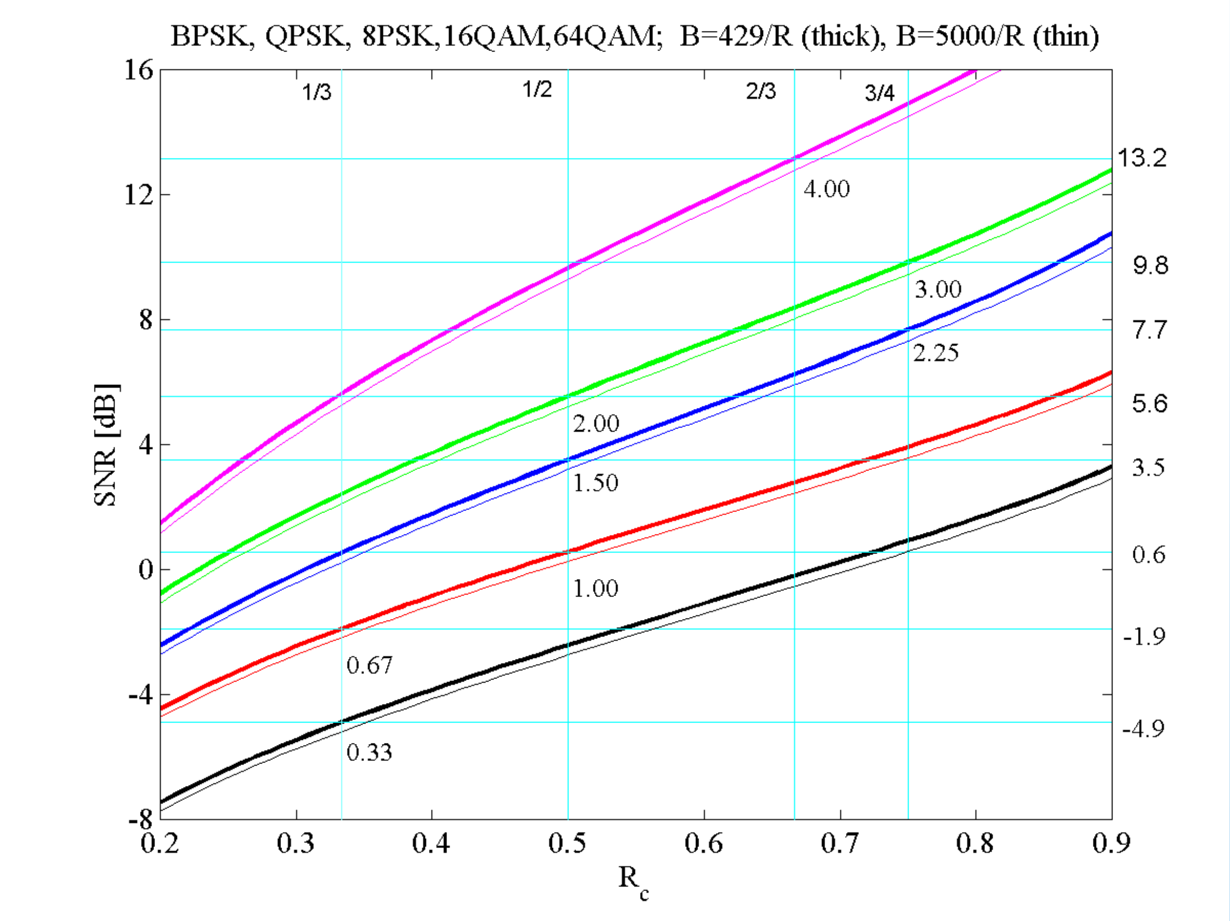


Adaptive System example

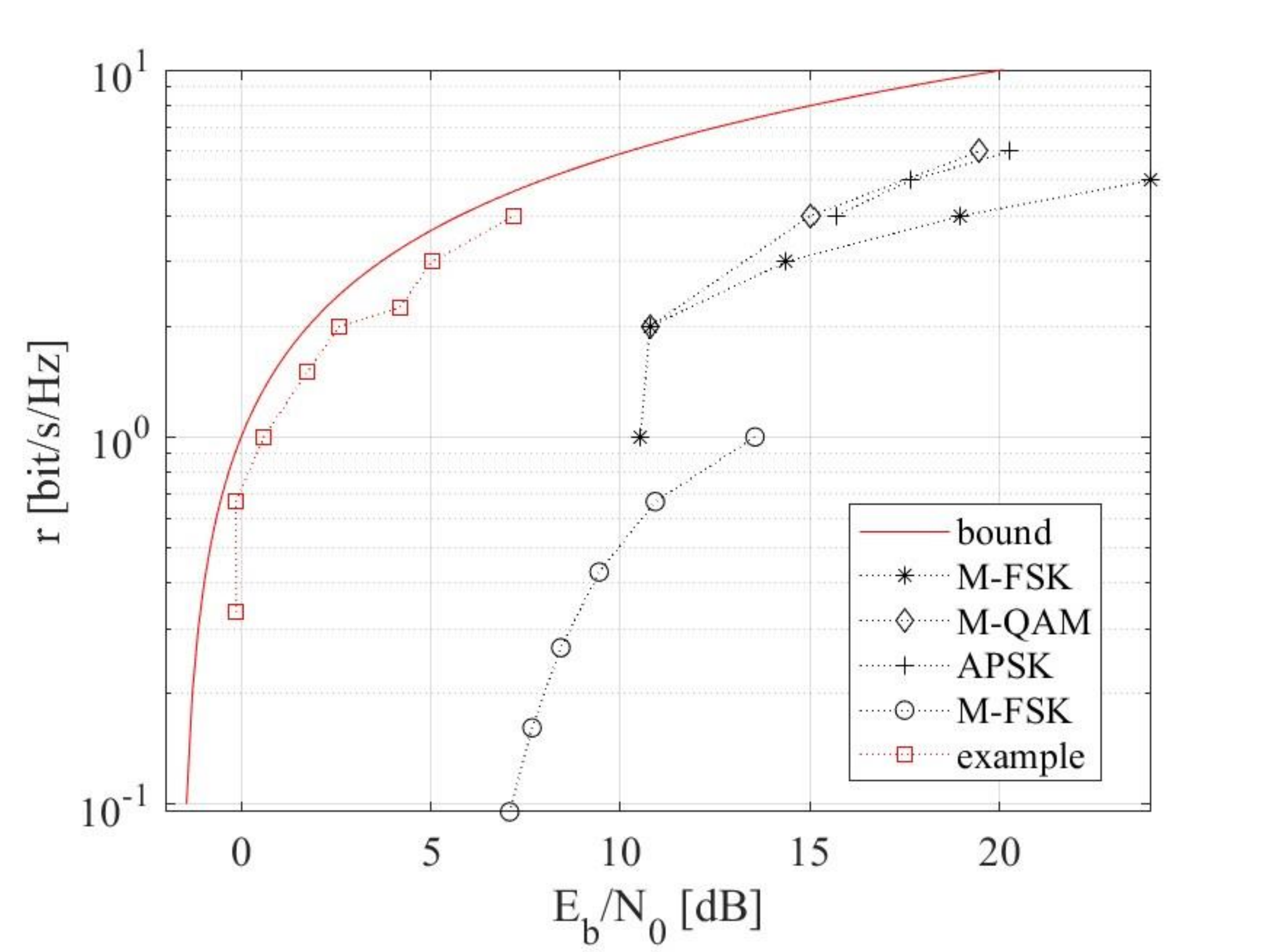


SNR threshold design (ON/OFF model)

- A possible set of thresholds, for block length N=429/R and N=5000/R, for R_c=1/3, 1/2, 2/3, 3/4.



Shannon bound comparison



Error Protection Techniques

Error protection techniques

- Error detection codes and retransmission algorithms (Automatic Repeat reQuest - ARQ).
- Self-correcting error codes (Forward Error Correction - FEC).
- Coding: *k* bits of information are associated with *n* bits for transmission on the channel, with *n*>*k*; redundancy is thus systematically introduced which can be used in reception to detect and/or correct errors.
- In the time *k* user bits are generated, *n* channel bits must be transmitted; therefore the time for the transmission of a bit is reduced and, consequently, the used bandwidth increases.
- Block codes: The *n*-*k* redundancy bits depend only on the current *k* user bits; the coding operation is, therefore, without memory.
- Convolutional codes: The *n*-*k* bits of redundancy depend on *k* current user bits and on (*N*-1)*k* previous user bits (*N* is called constraint length); the coding operation is, therefore, with memory.

Linear binary block codes

- The *n*-bit sequences produced by the encoder are called code words.
- The codes we will consider are linear codes, in which the sum (modulo 2) of two code words is still a code word.
- Weight of a code word: it is the number of bits other than 0 of the code word.
- Hamming distance between two code words: it is the number of bits in which they differ; it is also the weight of the word obtained by adding the code words modulo 2. Thus, the weight of the least weight word represents the minimum distance, d_{min}, between two codewords.

Performance of linear binary block codes

- Assume hard detection. The performance depend on the minimum distance. The decoding operation generally takes place in maximum likelihood, i.e. associating the closest code word to the n-tuple received.

- Consequently, a code with minimum distance $d_{\min}$ can correct at most $t = \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor$ weight errors, and reveal at most $d_{\min}$ weight errors.

- A code is used to correct $t_c \le \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor$ weight errors at most, and detect l weight errors, being $t_c + 1 \le l \le d_{\min} - 1 - t_c$

- Perfect codes: each n-tuple has at least one code word at distance $\left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor$. For them the relationship holds: $2^n = 2^k \sum_{i=0}^{t} \binom{n}{i}$

- The repetition codes (n,1), with n odd ($d_{\min}$ =n), the Hamming codes ($d_{\min}$ =3), and the Golay code ((23,12), $d_{\min}$ =7) are perfect codes.

Cyclic codes

- They are linear codes.
- A cyclic permutation of a codeword produces a codeword.
- Each binary sequence is associated with a polynomial whose coefficients are the bits that make up the sequence.
- Redundancy bits are found using a division algorithm. The divisor polynomial is denoted by $g(x)$, and has degree n-k. Therefore a cyclic code is characterized by n, k, $g(x)$.
- Systematic encoding.
  - The k-ple to be encoded is associated with a polynomial $u(x)$ of degree n-1, whose k most significant coefficients coincide with the user bits, while the other n-k bits are set to 0. Calculate the remainder, $r(x)$, of dividing u(x) by $g(x)$. The coefficients of $r(x)$ are the redundancy. The n-ple thus obtained is associated with a polynomial $c(x)$, divisible by g(x).
- The received sequence is associated with the polynomial $c_r(x)$. We divide by $g(x)$. If the remainder (syndrome) is null it is assumed that there have been no errors.

Cyclic codes for error detection

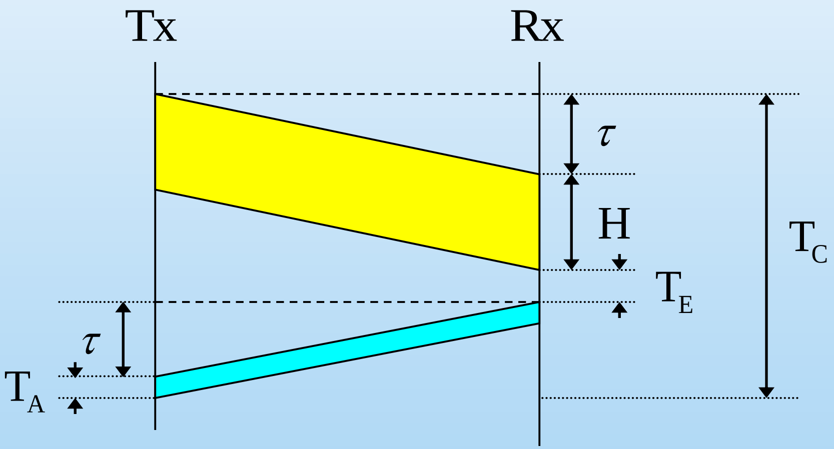
- Cyclic Redundancy Check (CRC)
- The generating polynomial is of the type $g(x)=g_1(x)(x+1)$, where $g_1(x)(x+1)$ is a polynomial of degree $m=n-k-1$ chosen so as to be a divisor of $x^{2^m-1}+1$, but not to be a divisor of any polynomial of the type x^h+1, with $h<2m-1$.
- With such choices, the code allows revealing all errors of odd weight (no polynomial with an odd number of terms can be divided by $x+1$) and all errors of weight 2 if $n<2^m-1$.
- The maximum code rate is $R_c=\left(2^m-m-3\right)/\left(2^m-2\right)$
- CRC is used by modern coding techniques to verify the correct decoding of a block.

CRC - Examples

n - k [bit]	Generator	n [bit]	k [bit]	$R_c = \frac{k}{n}$	Org.
5	$g(x)=x^5+x^3+x+1$	<15	<10	<2/3	ITU-T
8	$g(x)=x^8+x^7+x^3+x^2+1$	<127	<119	<119/127	ITU-T
16	$g(x)=x^{16}+x^{12}+x^5+1$	<32767	<32751	≅1	
32	$g(x)=x^{32}+x^{26}+x^{23}+x^{16}+x^{12}+x^{11}$ $+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$	$\cong 10^9$	$\cong 10^9$	≅1	IEEE
64	$g(x)=x^{64}+x^4+x^3+x+1$	$\cong 10^{18}$	$\cong 10^{18}$	≅1	ISO

Automatic Repeat reQuest

- Error detection.
- Bidirectional communication: availability of a return channel (feedback) on which ACKnowledgement (ACK) packets can be sent.
- Service tolerance for delay.



τ : propagation delay

2τ: round trip delay

H : packet transmission time

T_E: Elaboration time

T_A: ACK transmission time

T_C: cycle duration

$$T_C = 2\tau + H + T_E + T_A \quad \Rightarrow \quad \boxed{T_C \cong 2\tau + H}$$

$T_E,\ T_A << \tau,\ H$

The sustainable rate

- Assume that a codeblock of size *N* is transmitted in changing channel conditions.
- Assume that the codeblock may be divided into segments, each of which experiences a constant channel condition. With this assumption, the code performance may be determined as follows.
- Name with $[n_1, n_2, \ldots, n_L]$ the sequence of segment lengths and with $[\gamma_1, \gamma_2, \ldots, \gamma_L]$ the sequence of the relevant SNRs.
- Define sustainable code rate as the weighted average value of the equivalent code rates, that is given by

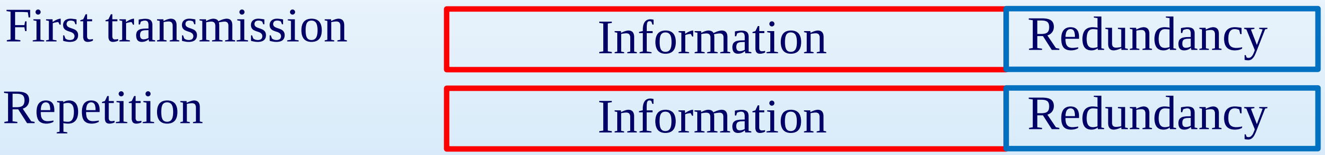
$$R_s = \sum_{j=1}^{L} \frac{R_j n_j}{N}$$

  in which R_j is the equivalent rate of the *j*-th segments, which, assuming the ON-OFF model, represents the maximum rate which can be used with the SNR γ_j and block size *N*.
- The sustainable rate may be seen as the maximum rate that can be successfully used, given the block size the segment length sequence, and the SNR sequence.

F. Babich, “On the Performance of Efficient Coding Techniques Over Fading Channels,” IEEE Transactions on Wireless Communications, vol. 3, no. 1, pp. 290–299, Jan. 2004.

Hybrid ARQ

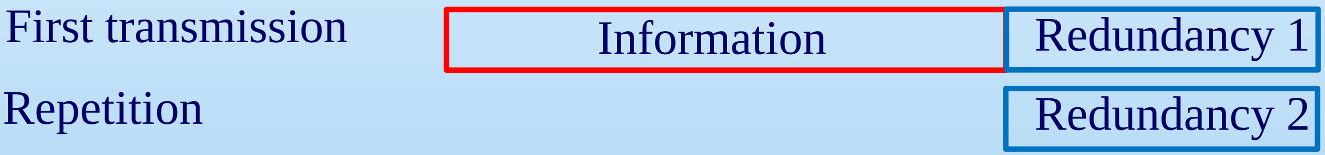
- Channel coding and repetition algorithm interact.
  - *Repetition Hybrid ARQ with soft (Chase) combining*: assume systematic encoding. The soft values of the received replicas are stored and combined to increase the probability of correct decoding.



First transmission | Information | Redundancy

Repetition | Information | Redundancy

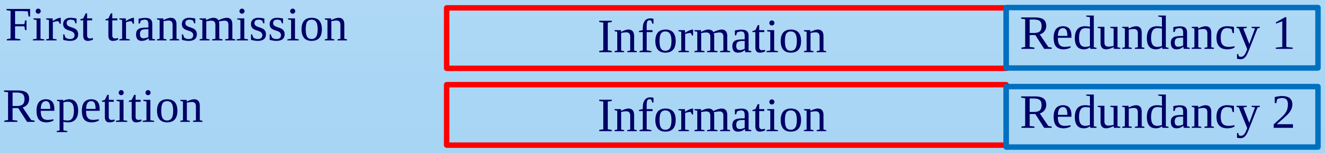
  - *Incremental Hybrid* ARQ: Repetitions only include additional redundancy.



First transmission | Information | Redundancy 1

Repetition | Redundancy 2

  - *Complementary Hybrid* ARQ: replications include information (systematic coding) and additional redundancy



First transmission | Information | Redundancy 1

Repetition | Information | Redundancy 2

Hybrid ARQ performance (I)

- Parameters
  - b: bit per symbol.
  - k: information bits
  - p_1, p_2 : redundancy bit first, second transmission.
  - $R_{c1}=k/(k+ p_1)$: code rate first transmission.
  - $R_{c2}=k/(k+ p_1+p_2)$: code rate second transmission.
  - γ_1, γ_2 : SNR first, second transmission.
  - $R_1=R_{c1}b$, $R_2=R_{c2}b$: rate first, second transmission.
- Limiting performance
- First transmission success condition:
  $R_1 < \log_2(1+\gamma_1)$, from which $\boxed{\gamma_1 > 2^{R_1} - 1}$
- Repetition ARQ (Chase combining): in case of failure of the first transmission, the limiting condition for the success of the second is :
  $R_1 < \log_2(1+\gamma_1+\gamma_2)$, from which $\boxed{\gamma_2 > 2^{R_1} - 1 - \gamma_1}$

Hybrid ARQ performance (II)

- Incremental ARQ: in case of failure of the first transmission be
  - $\alpha = \dfrac{k + p_1}{k + p_1 + p_2} = \dfrac{R_{c_2}}{R_{c_1}}$: fraction of bits sent in the first transmission;
  - limiting condition for the success of the second is :

$R_2 < \alpha \log_2(1+\gamma_1) + (1-\alpha)\log_2(1+\gamma_2)$ from which: $\boxed{\gamma_2 > 2^{\frac{R_2 - \alpha\log_2(1+\gamma_1)}{(1-\alpha)}} - 1}$

if $p_2 = p_1$, $R_{c2}=R_{c1}/(2-R_{c1})$, we have: $\boxed{\gamma_2 > 2^{\frac{bR_{c_1} - \log_2(1+\gamma_1)}{(1-R_{c_1})}} \cdot - 1}$

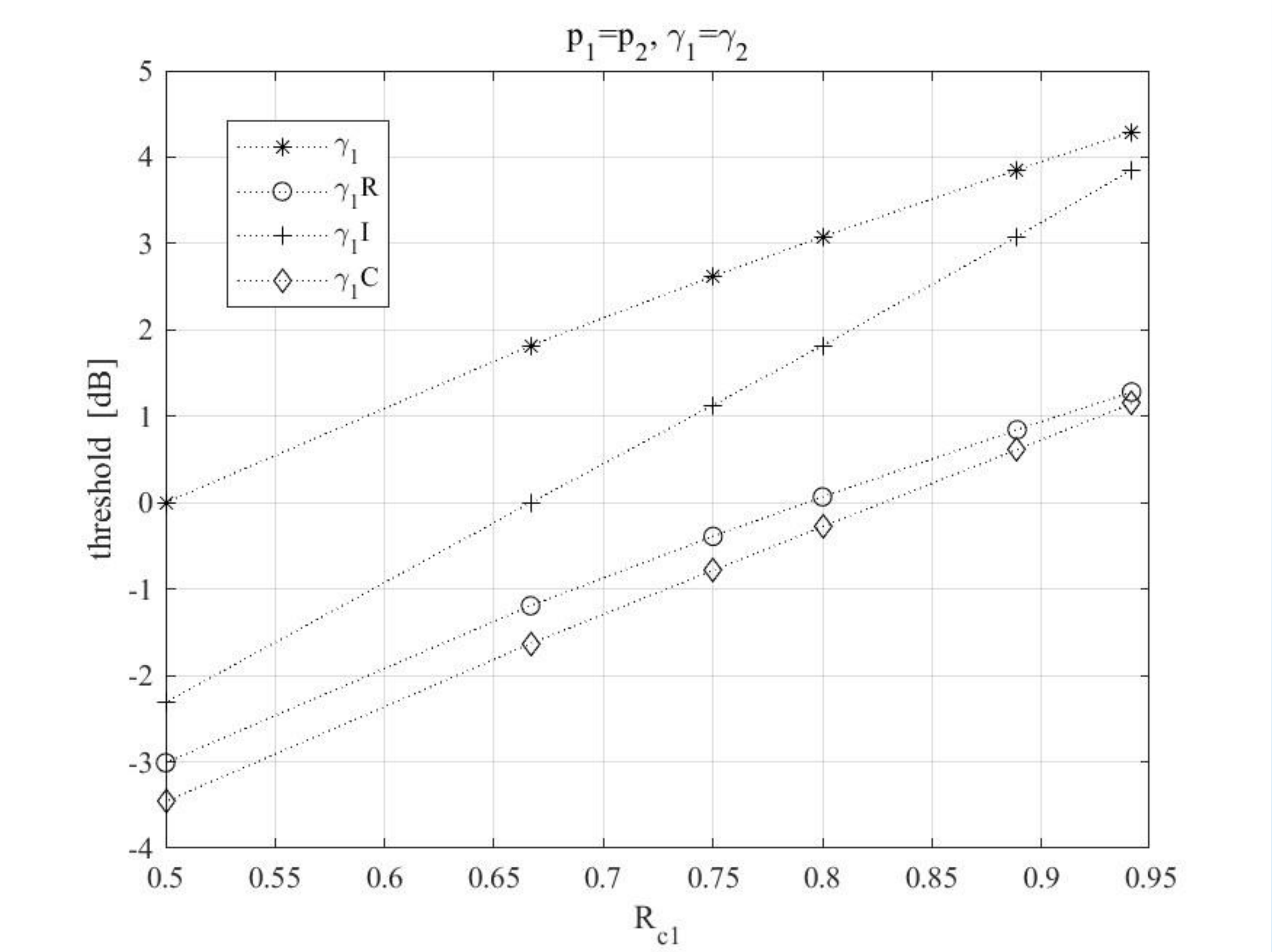
- *Complementary* ARQ : in case of failure of the first transmission be

$\alpha = R_{c_2}$ the fraction of bits of information, received with $\gamma_T = \gamma_1 + \gamma_2$;

  - hypothesizing $p_1 = p_2$, be β=(1-α)/2 the fraction of parity received with both γ_1 and γ_1; the limiting condition for the success of the second is:

$$\boxed{R_2 = b\alpha < \alpha\log_2(1+\gamma_1+\gamma_2) + \frac{(1-\alpha)}{2}\left[\log_2(1+\gamma_1) + \log_2(1+\gamma_2)\right]}$$

Hybrid ARQ performance (III)



Channel coding techniques

Parity check codes

- They are linear codes.

- The n-k bits of redundancy come from parity operations.

- Coding: it is done using the generating matrix $\mathbf{G}$ [k x n]; the encoded vector $\mathbf{x}$ is obtained from the unencoded vector $\mathbf{u}$ by means of the operation $\mathbf{x}=\mathbf{uG}$. If the first k bits of $\mathbf{x}$ coincide with $\mathbf{u}$, the code is said to be systematic.

- Error detection (parity check): this is done using the parity check matrix $\mathbf{H}$ [(n-k) x n]; if the received vector is $\mathbf{y}=\mathbf{x}\oplus\mathbf{e}$, where $\mathbf{e}$ is the error vector, $\mathbf{s}=\mathbf{yH}'=(\mathbf{x}\oplus\mathbf{e})\,\mathbf{H}'=\mathbf{eH}'$ is calculated (where $\oplus$ it indicates the operation of addition module 2 and ' it indicates the transposition operation). If there were no errors, the vector $\mathbf{s}$ is the null vector. It turns out to be $\mathbf{GH}'=\mathbf{0}$. The vector $\mathbf{s}$ is called syndrome.

- Error correction (hard decoding): to each vector $\mathbf{s}$ it is associated a vector $\mathbf{e}$, so that the decoded vector is $\mathbf{v}=\mathbf{y}\oplus\mathbf{e}$. The association takes place at a minimum distance (to each syndrome it is associated the minimum weight error sequence that produces that given syndrome). The non-zero syndromes (and therefore the correctable error sequences) are $2^{n-k}-1$.

Hamming codes

- Parity check codes with the following parameters: n=2^m-1, n-k=m, $d_{\min}$=3, being m≥3 an integer.

- The columns of the parity check matrix consist of all the non zero m-length sequences.

- Example, m=3, n=7, k=4, systematic version

$$\mathbf{H} = \left[\begin{array}{cccc|ccc} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array}\right] = \left[\mathbf{P} \middle| \mathbf{I}_{n-k}\right]$$

- The hard decoding consists of identifying the single error position by comparing the syndrome with the columns of **H**.

$$\mathbf{G} = \left[\begin{array}{cccc|ccc} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{array}\right] = \left[\mathbf{I}_k \middle| \mathbf{P}'\right]$$

- If more than one error occur the decoding is wrong.

Coding gain

- The performance of a channel code may be determined through the coding gain, which is difference between the SNR (E_b/N$_0$) required for achieving a given BER, without and with code.
- Consider a binary antipodal modulation, for which the BER before decoding is given by

$$p_b = Q\left(\sqrt{2R_c \frac{E_b}{N_0}}\right)$$

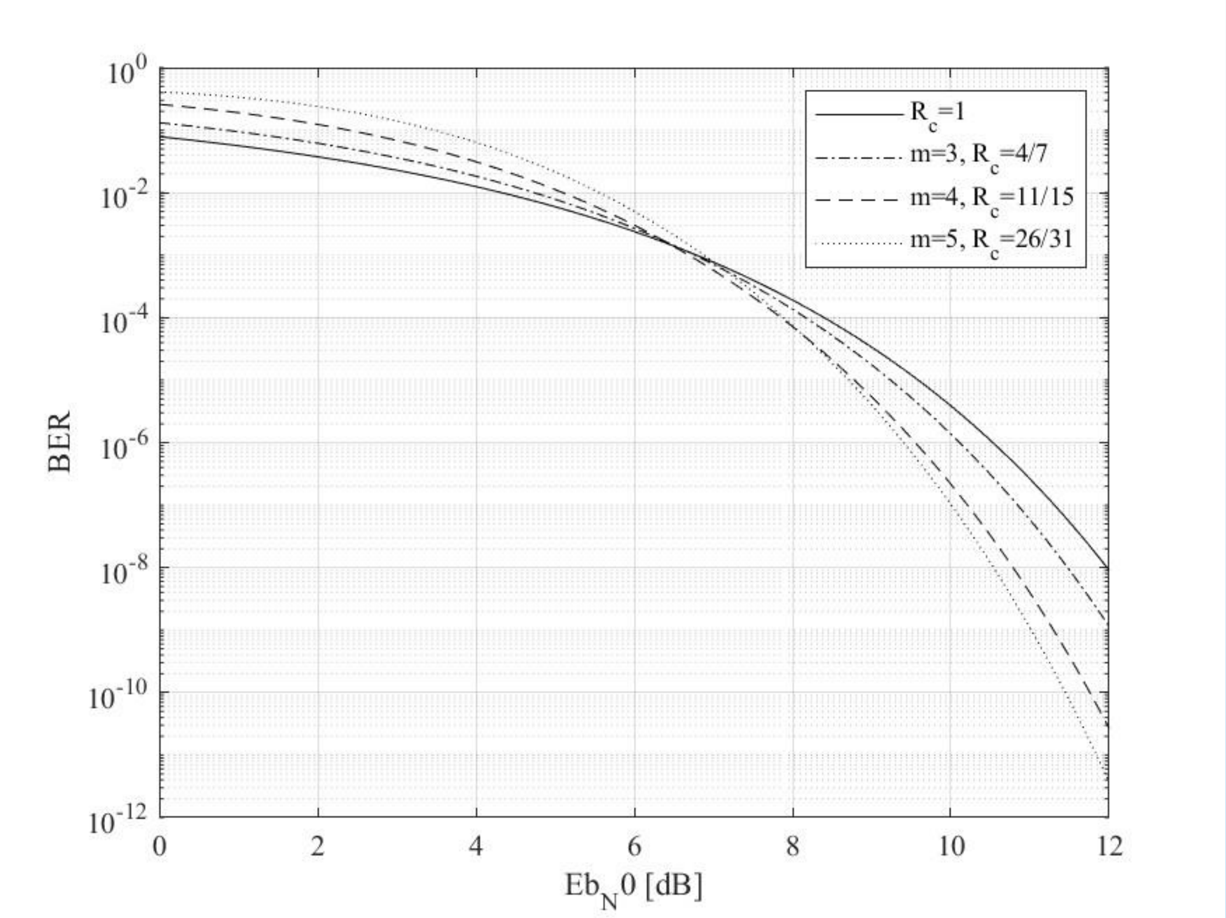
- Without code R_c=1, while with a code of error correction capability t_c, the probability of correct word reception is given by:

$$P_c = \sum_{i=0}^{t_c} \binom{n}{i} p_b^i \left(1-p_b\right)^{n-i}$$

- If the full error correction is used, t_c=t, and the probability of wrong word reception is given by 1- P_c.

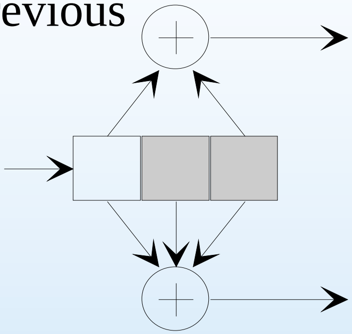
Example: Hamming codes

- t_c=1. By increasing *m*, the block size increases, the code rate approaches 1, and the coding gain, for low error rates, slightly increases.

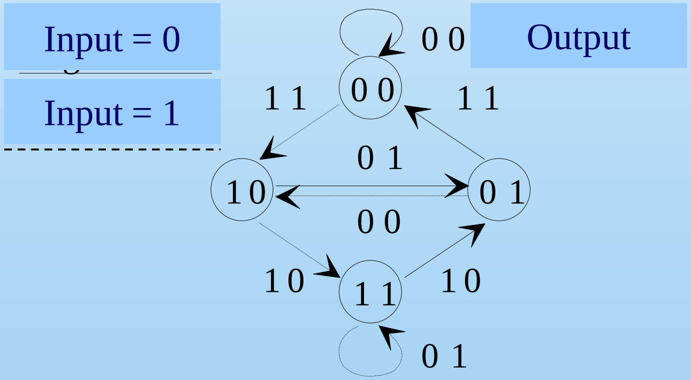


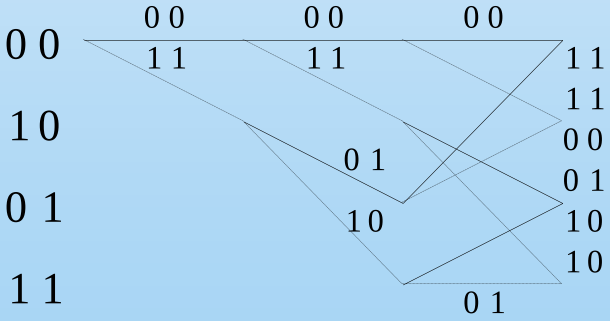
Convolutional Codes

- The coded sequences depends on the current and on the previous inputs.
- Example: R_c=1/2. The previous values of the input that have an influence on the current value of the output are highlighted (these values define the state of the system). The constraint length, L, is the size of the register (in the example L=3). The input/output relations define the generators which are indicated in octal notation. In the example g_1=(1 0 1) (indicated by 5), g_2=(1 1 1) (indicated by 7).



- Evolution is described by state diagram or trellis diagram.





Convolutional codes

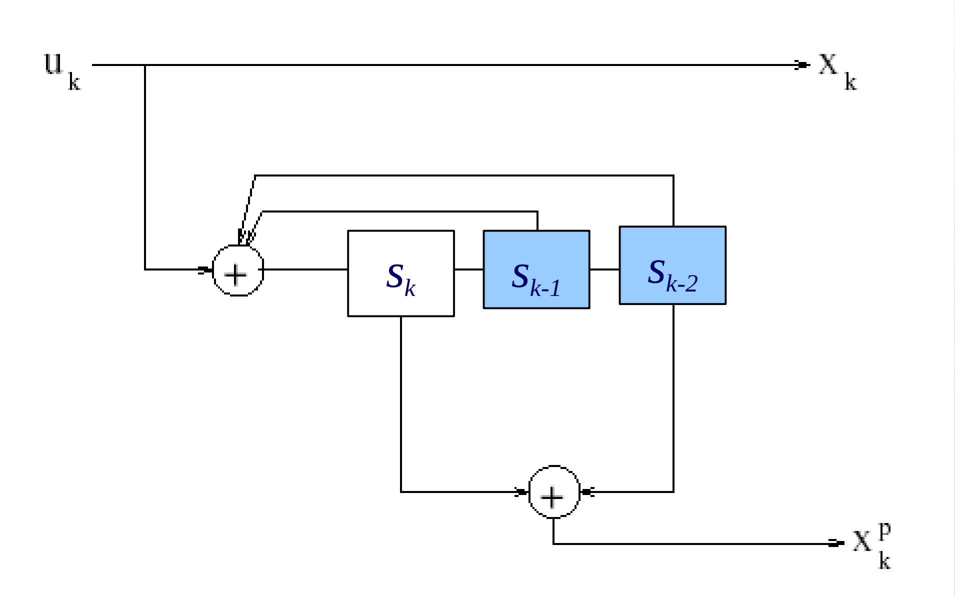
- It is a linear code.
- The performance depends on the free distance, d_{free}, which is the weight of the lowest weight non zero sequence. It may me determined through the transfer function.
- In our example we obtain the transfer function $\frac{D^5}{1-2D} = D^5 + 2D^6 + 4D^7 + \ldots$

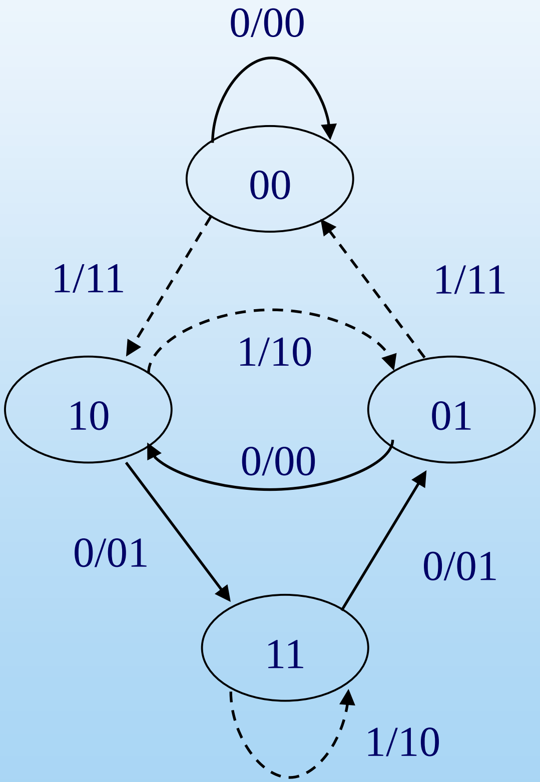
  There is a sequence of weight 5 (input 100 which produces output 110111), two sequences of weight 6 (input 1100 which produces output 11101011 and input 10100 which produces output 1101000111), 4 weighing 7, and so on.

- Decoding takes place at a minimum distance, using the Viterbi algorithm.
- The algorithm does not change if the soft decision is used. Euclidean distance is used instead of Hamming distance.
- Complexity grows linearly with length.

Recursive Convolutional Codes

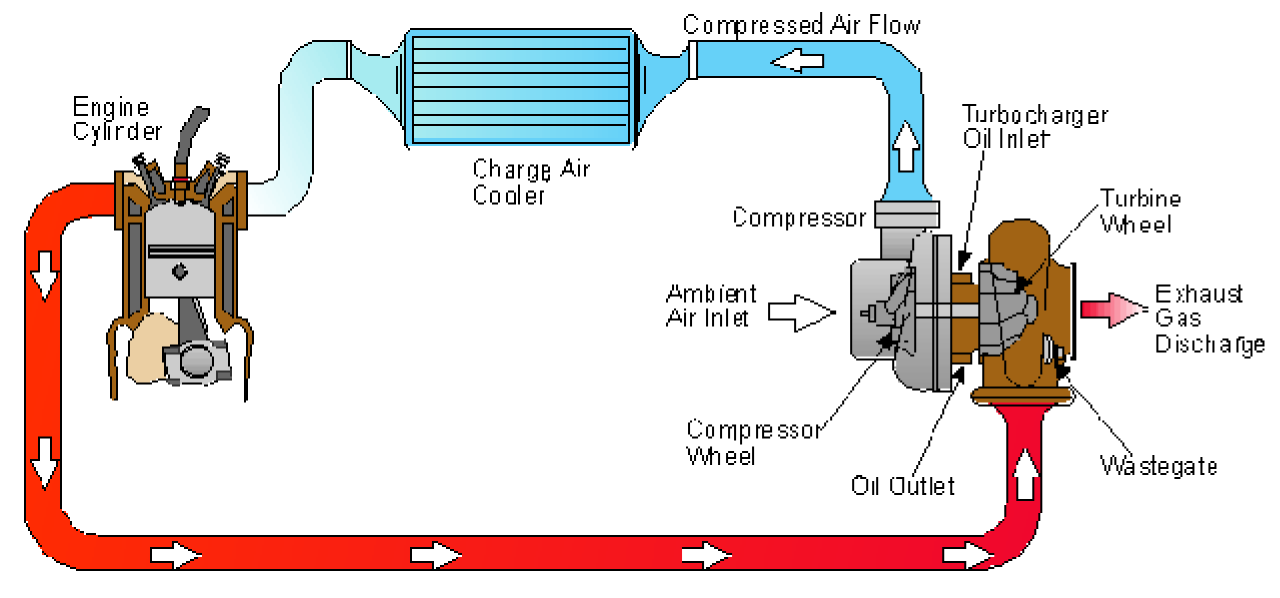
- It may be systematic (the output sequence contains the input sequence)
- It may be recursive.
- Example of systematic and recursive encoder with the relevant state diagram.
- It is the key component of the turbo encoder.





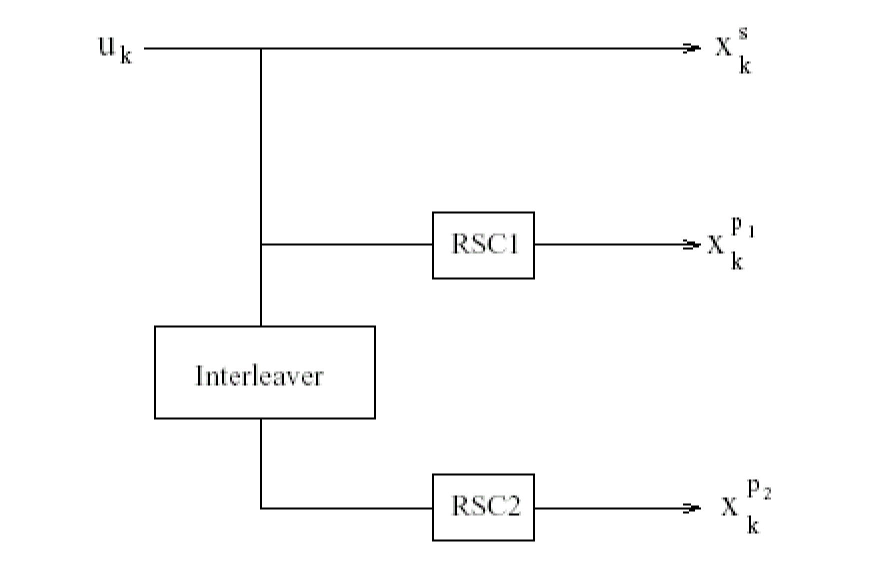
Iterative (turbo) decoding

- In 1993 Berrou and Glavieux proposed the concatenate iterative decoding (turbo decoding).
- Turbo codes owe their name to the decoding algorithm, which uses feedback, like the turbo engine: this algorithm is called iterative decoding.

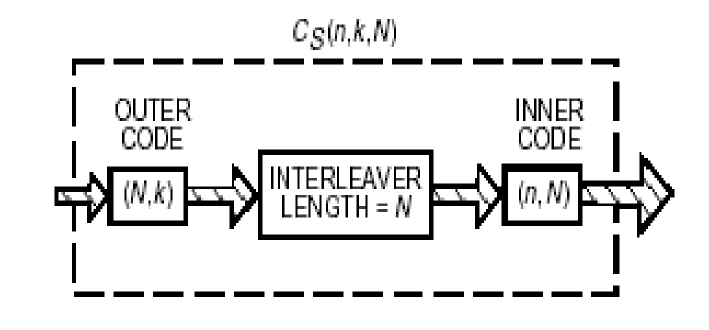


Parallel Concatenated Convolutional Codes

- A turbo encoder consists of the parallel concatenation of two recursive convolutional codes (PCCC: Parallel Concatenated Convolutional Codes) by means of a interleaver.

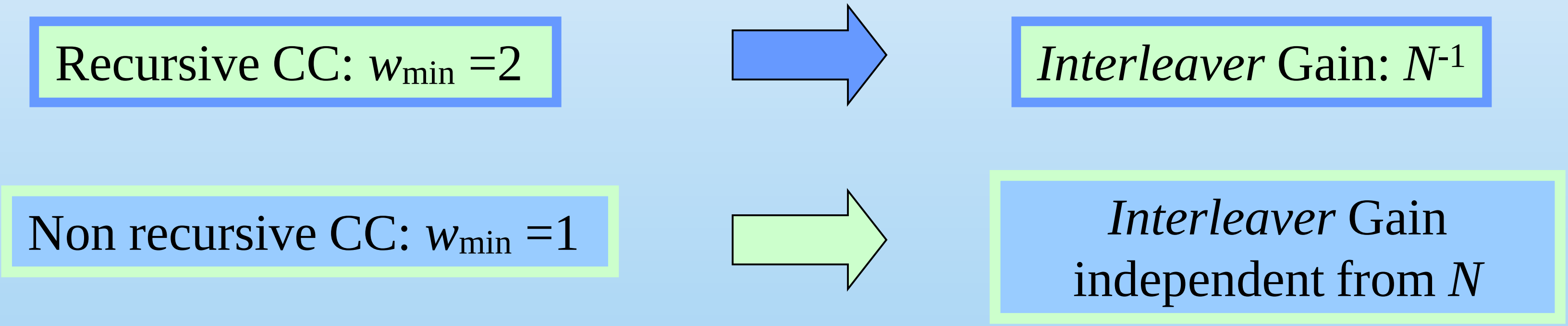


- A serial concatenation is also possible (and even preferable) with different properties and performance (SCCC)



The role of the interleaver

- Name w the weight of an input sequence that may produce a low weight output sequence.
- For non recursive convolutional encoder w_{min}=1 (an input sequence consisting of single 1 produces a low weight output sequence).
- For recursive convolutional encoder w_{min}=2 (an input sequence consisting of single 1 produces an oscillating output sequence). The interleaver may be carefully designed for counteract the effects of w=2 sequences, providing the interleaver gain.



Recursive CC: w_{min} =2 → *Interleaver* Gain: N^{-1}

Non recursive CC: w_{min} =1 → *Interleaver* Gain independent from N

Design parameters

- Parallel versus serial concatenation
- Number of concatenated codes
- Memory (# of states) of each code
- Code rate and generating polynomials for each constituent code
- Block length
- Excellent versus suboptimal decoding algorithms
- Trellis termination techniques
- Interleaver design
- Early iteration stopping techniques
- Using external block codes to lower the error floor

Design considerations

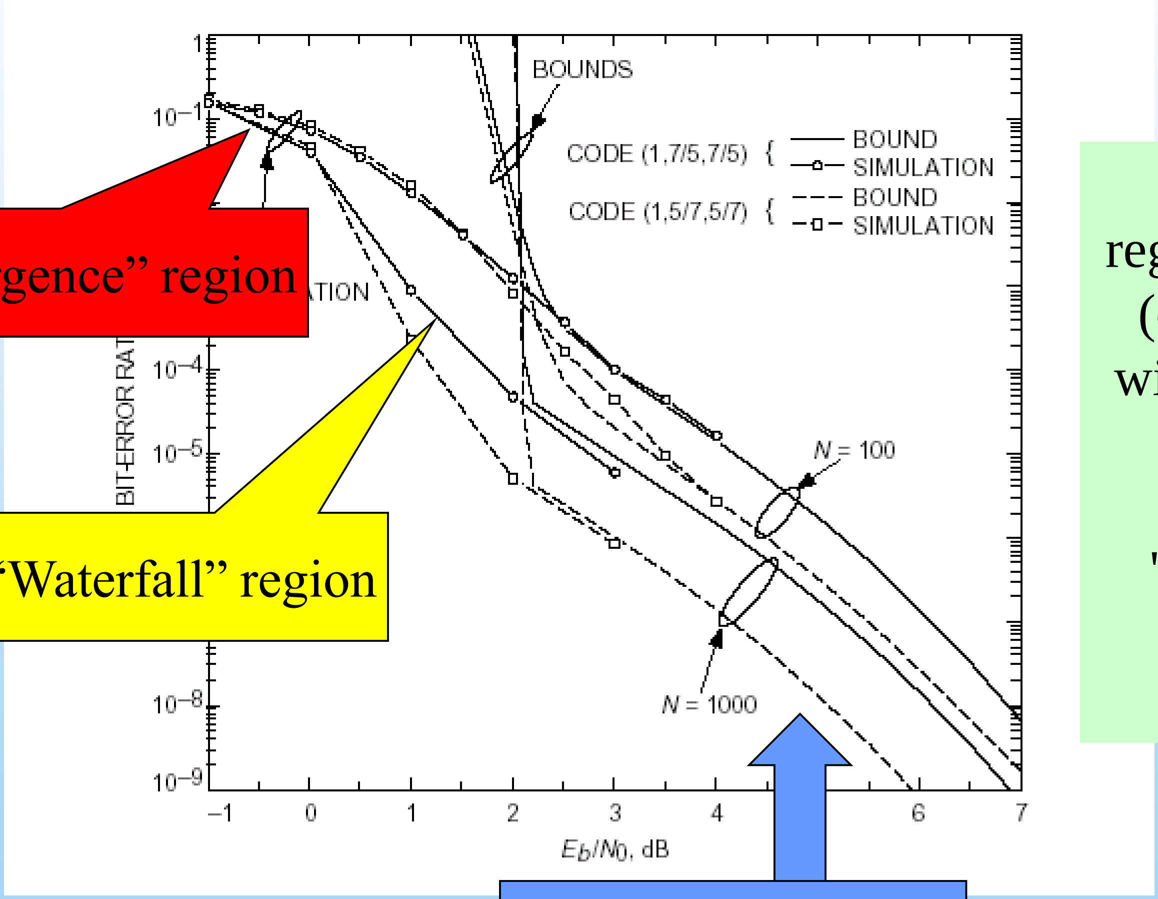
- Assume that the minimum weight of the input sequences used for leaving and going back to the state 0 is 2. The asymptotic expression (with respect to the interleaver length *N*) of the bit error probability is:

$$P_b(e) \approx \frac{1}{2} N^{-1} N_{\mathrm{C_1 eff}} N_{\mathrm{C_2 eff}} \operatorname{erfc}\left(\sqrt{\frac{d_{\mathrm{Peff}} R_c E_b}{N_0}} \right)$$

- being d_{Peff} is the effective free distance of the PCCC, i.e. the minimum weight of the PCCC sequences generated from input sequences of weight 2, and $N_{\mathrm{C}i\mathrm{eff}}$ (i=1,2) is the multiplicity of error events of the two CCs with weight equal to the effective free distance.

- It follows that, to improve the turbo code performance is necessary to increase d_{Peff} and to decrease $N_{\mathrm{C}i\mathrm{eff}}$.

Performance



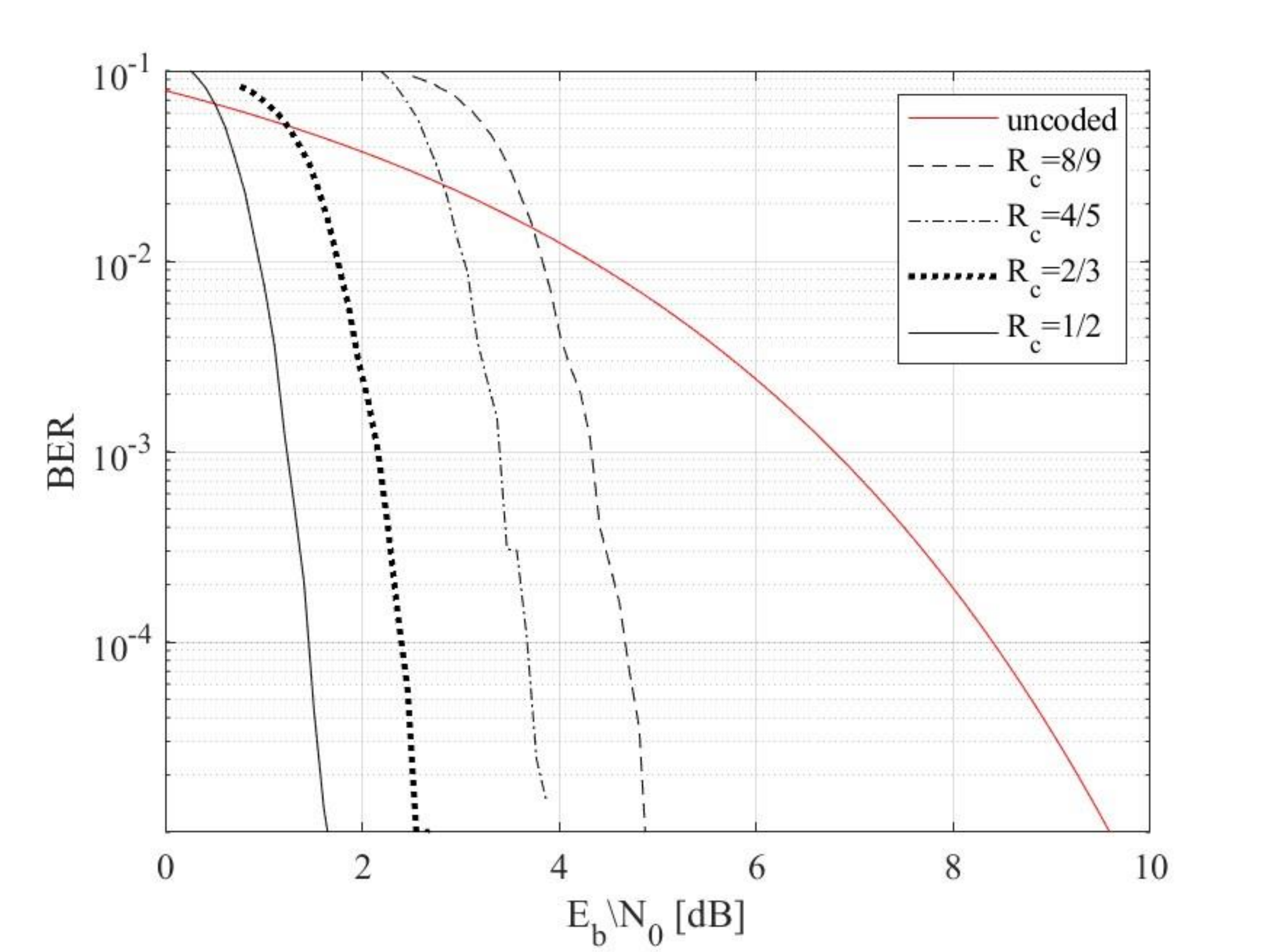
"Non convergence" region

"Waterfall" region

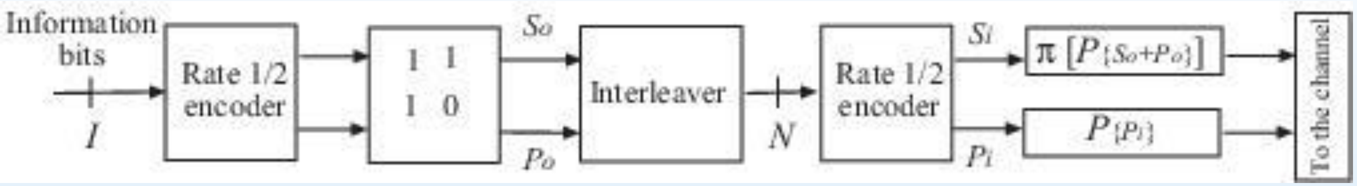
"Error floor" region

Three distinct regions of the BER (or FER) curves with respect to the SNR: the non-convergence, "waterfall" and "error floor" regions

PCCC coding gain

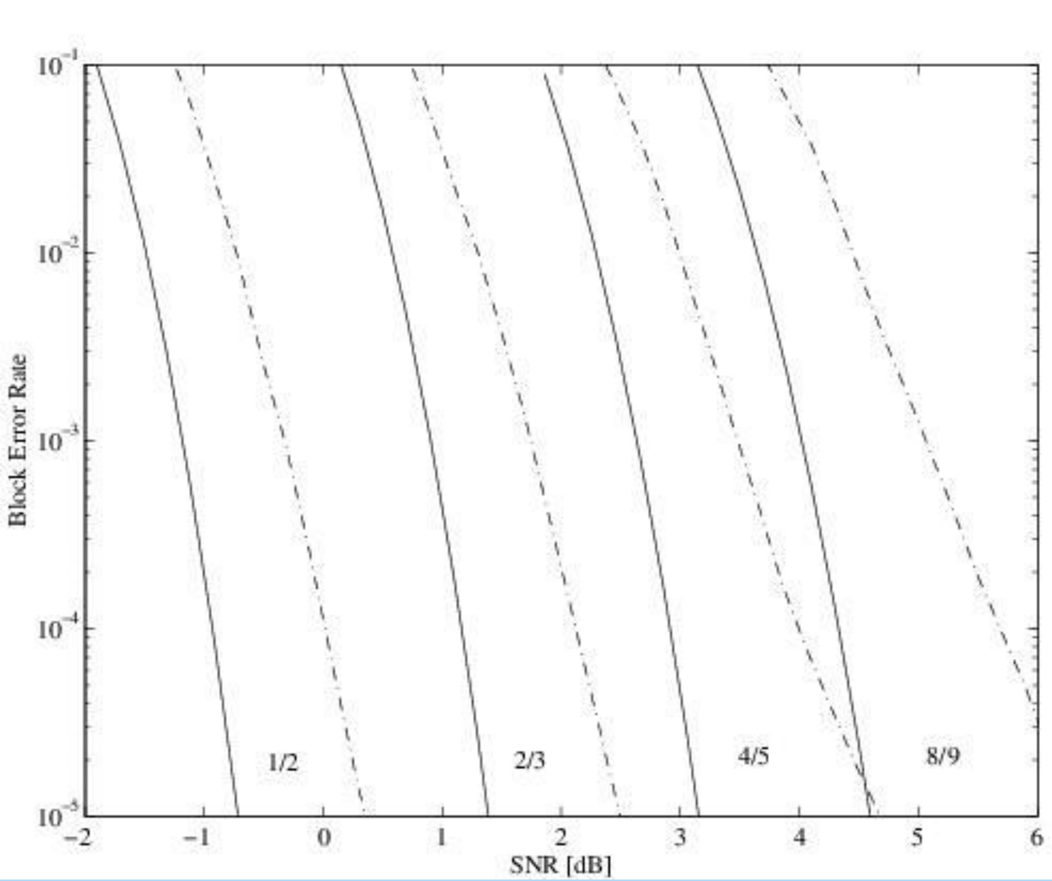


SCCC example



- Comparison of some rate compatible puncturing schemes, obtained starting from a rate 1/3 SCCC mother code formed by serially concatenating a 4-states rate 2/3 outer systematic recursive convolutional code (SRCC) and a 4-states rate 1/2 inner SRCC. A non periodic puncturing has been applied on inner output bits only. The frame length, measured in terms of input information bits, has been set to 200 using a random interleaver.

SCCC performance



- N=200, random interleaver
- The actual performance is compared with the limiting one

Low Density Parity Check Codes

- LDPC codes are parity check codes with sparse (low density) check matrix. They were first proposed by Gallagher in his 1960 doctoral dissertation.
- Most standards adopt LDPC codes instead of Turbo Codes. this derives partly from some patent issues related to Turbo Codes.
- But the LDPC have demonstrated to achieve a performance that is very close to the Shannon bound with simpler decoding algorithms.
- The decoding algorithms are independent from the coding rate (this is also true for the Turbo Codes).
- Both LDPC and Turbo Codes have a poor performance for short packets.

PhD in Industrial and Information Engineering

Recursive decoding for Parity Check Codes

MAP in the presence of a code

- Assume that **y** is a sequence of received noisy values, obtained by transmitted a sequence **x** of antipodal, binomial values. The MAP criterion may be applied as follows.

$$\log\frac{p\left(x_i=+\middle|\mathbf{y}\right)}{p\left(x_i=-\middle|\mathbf{y}\right)}=\log\frac{p\left(\mathbf{y}\middle|x_i=+\right)p\left(x_i=+\right)}{p\left(\mathbf{y}\middle|x_i=-\right)p\left(x_i=-\right)}=\log\frac{p\left(y_1,y_2,...,y_n\middle|x_i=+\right)p\left(x_i=+\right)}{p\left(y_1,y_2,...,y_n\middle|x_i=-\right)p\left(x_i=-\right)}=$$

$$=\log\frac{p\left(y_{1,}y_{2,}..,y_{i-1,}y_{i+1,}..,y_n\middle|x_i=+\right)p\left(y_{i,}\middle|x_i=+\right)p\left(x_i=+\right)}{p\left(y_{1,}y_{2,}..,y_{i-1,}y_{i+1,}..,y_n\middle|x_i=-\right)p\left(y_{i,}\middle|x_i=-\right)p\left(x_i=-\right)}$$

$$=\log\frac{p\left(y_{1,}y_{2,}..,y_{i-1,}y_{i+1,}..,y_n\middle|x_i=+\right)}{p\left(y_{1,}y_{2,}..,y_{i-1,}y_{i+1,}..,y_n\middle|x_i=-\right)}$$

Code

$$+\log\frac{p\left(y_{i,}\middle|x_i=+\right)}{p\left(y_{i,}\middle|x_i=-\right)}$$

Channel

$$+\log\frac{p\left(x_i=+\right)}{p\left(x_i=-\right)}$$

A priori

Parity codes: APP math

- Name $L=\log\frac{p_+}{p_-}=\log\frac{p_+}{1-p_+}$. We have $p_+=\frac{e^L}{1+e^L}$.

- Consider a pair of (antipodal) bits, u_1, u_2. u_1u_2=+ se u_1=u_2.

$$p_+(u_1u_2)=p(u_1^+)p(u_2^+)+(1-p(u_1^+))(1-p(u_2^+))=1-p(u_1^+)-p(u_2^+)+2p(u_1^+)p(u_2^+)$$

$$p_-(u_1u_2)=p(u_1^+)+p(u_2^+)-2p(u_1^+)p(u_2^+)$$

- Substituting, after some simple calculations we get

$$p_+(u_1u_2)=\frac{1+e^{L(u_1)}e^{L(u_2)}}{\left(1+e^{L(u_1)}\right)\left(1+e^{L(u_2)}\right)}$$

$$p_-(u_1u_2)=\frac{e^{L(u_1)}+e^{L(u_2)}}{\left(1+e^{L(u_1)}\right)\left(1+e^{L(u_2)}\right)}$$

$$L(u_1u_2)=\log\left[\frac{1+e^{L(u_1)}e^{L(u_2)}}{e^{L(u_1)}+e^{L(u_2)}}\right]$$

APP math II

- By induction it is straightforward to demonstrate that

$$L\left(u_1u_2...u_J\right)=\log\left[\frac{\prod_{j=1}^{J}\left(e^{L\left(u_j\right)}+1\right)+\prod_{j=1}^{J}\left(e^{L\left(u_j\right)}-1\right)}{\prod_{j=1}^{J}\left(e^{L\left(u_j\right)}+1\right)-\prod_{j=1}^{J}\left(e^{L\left(u_j\right)}-1\right)}\right]$$

define $\alpha=\frac{\prod_{j=1}^{J}\left(e^{L\left(u_j\right)}-1\right)}{\prod_{j=1}^{J}\left(e^{L\left(u_j\right)}+1\right)}=\frac{\prod_{j=1}^{J}\left(e^{L\left(u_j\right)/2}-e^{L\left(u_j\right)/2}\right)}{\prod_{j=1}^{J}\left(e^{L\left(u_j\right)/2}+e^{L\left(u_j\right)/2}\right)}=\prod_{j=1}^{J}\tanh\left(L\left(u_j\right)/2\right)$

we have $L\left(u_1u_2...u_J\right)=\log\left[\frac{1+\alpha}{1-\alpha}\right]$

from which $\alpha=\frac{e^{L\left(u_1u_2...u_J\right)}-1}{e^{L\left(u_1u_2...u_J\right)}+1}=\tanh\left(L\left(u_1u_2...u_J\right)/2\right)$

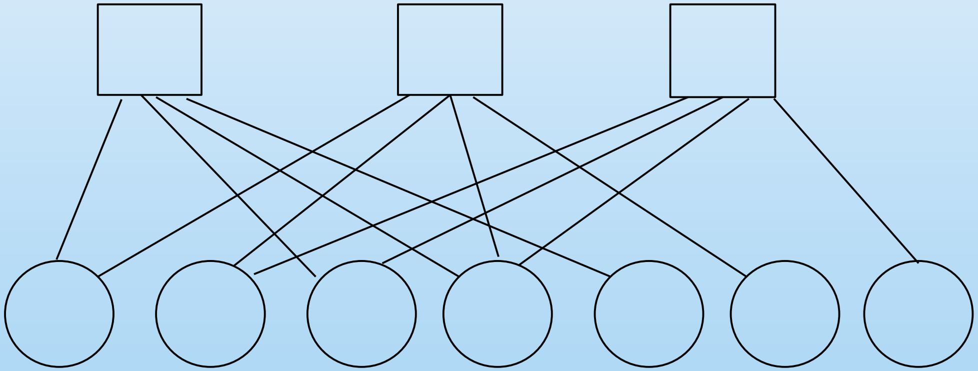
We finally obtain $L\left(u_1u_2...u_J\right)=2\ \text{arctanh}\left(\alpha\right)=2\ \text{arctanh}\left(\prod_{j=1}^{J}\tanh\left(L\left(u_j\right)/2\right)\right)$

Graph Representations of Codes
Tanner Graph

- A Tanner Graph is a bipartite graph which represents the parity check matrix of an error correcting code.
- **H** is the (*n*–*k*)-by-*n* parity check matrix. The Tanner graph has:
- *n* bit nodes (or variable nodes), represented by circles.
- *n*–*k* check nodes, represented by squares.
- There is an edge between bit node *i* and check node *j* if there is a one in row *i* and column *j* of **H**.

$$\mathbf{H} = \left[\begin{array}{cccc|ccc} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array}\right]$$

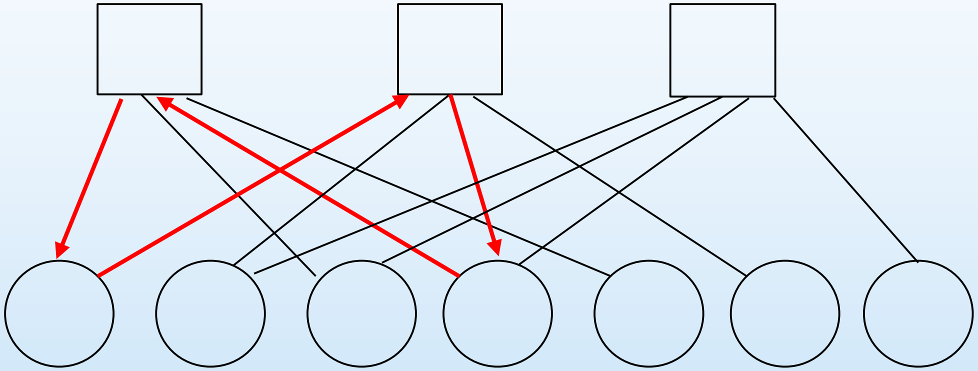
Hamming Code



Tanner graph

Cycles and Girth of Tanner Graphs

-



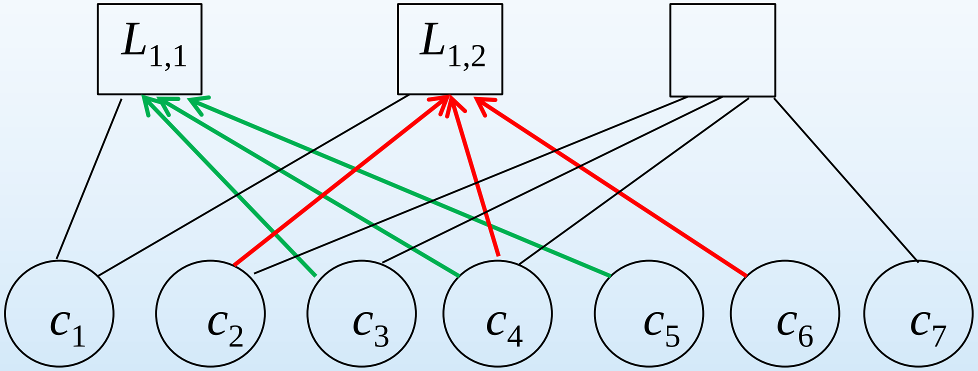
$$\mathbf{H} = \left[\begin{array}{cccc|ccc} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array}\right]$$

A cycle of length L in a Tanner graph is a path of L edges which closes back on itself

The girth of a Tanner graph is the minimum cycle length of the graph.

Message passing decoding

$$\mathbf{H}=\left[\begin{array}{cccc|ccc}1&0&1&1&1&0&0\\1&1&0&1&0&1&0\\0&1&1&1&0&0&1\end{array}\right]$$



$c_1 \oplus c_3 \oplus c_4 \oplus c_5 = 0$

$c_1 \oplus c_2 \oplus c_4 \oplus c_6 = 0$

$c_2 \oplus c_3 \oplus c_4 \oplus c_7 = 0$

$c_1 = c_3 \oplus c_4 \oplus c_5$

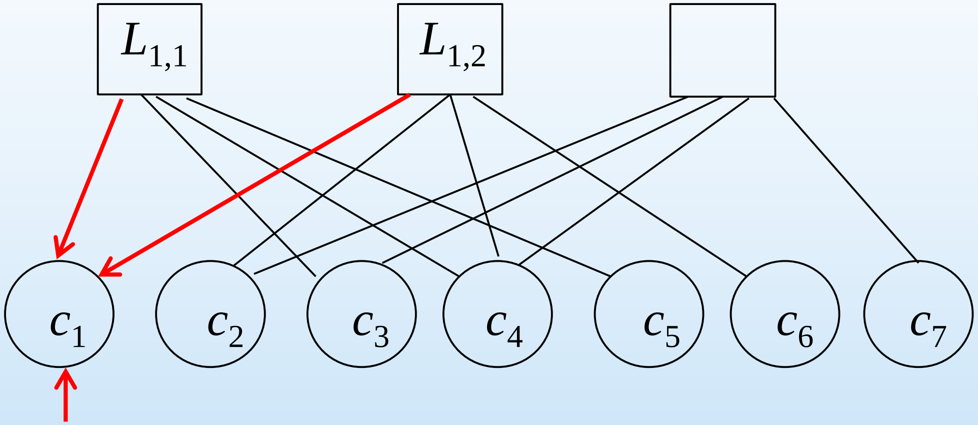
$$L_{1,1} = 2\ \mathrm{arctanh}\left(\tanh\left(\frac{L(c_3)}{2}\right)\tanh\left(\frac{L(c_4)}{2}\right)\tanh\left(\frac{L(c_5)}{2}\right)\right)$$

$c_1 = c_2 \oplus c_4 \oplus c_6$

$$L_{1,2} = 2\ \mathrm{arctanh}\left(\tanh\left(\frac{L(c_2)}{2}\right)\tanh\left(\frac{L(c_4)}{2}\right)\tanh\left(\frac{L(c_6)}{2}\right)\right)$$

Message passing decoding

$$\mathbf{H} = \left[\begin{array}{cccc|ccc} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array}\right]$$



$L_{1,0}$ from the channel

$$L(c_1) = L_{1,0} + L_{1,1} + L_{1,2}$$

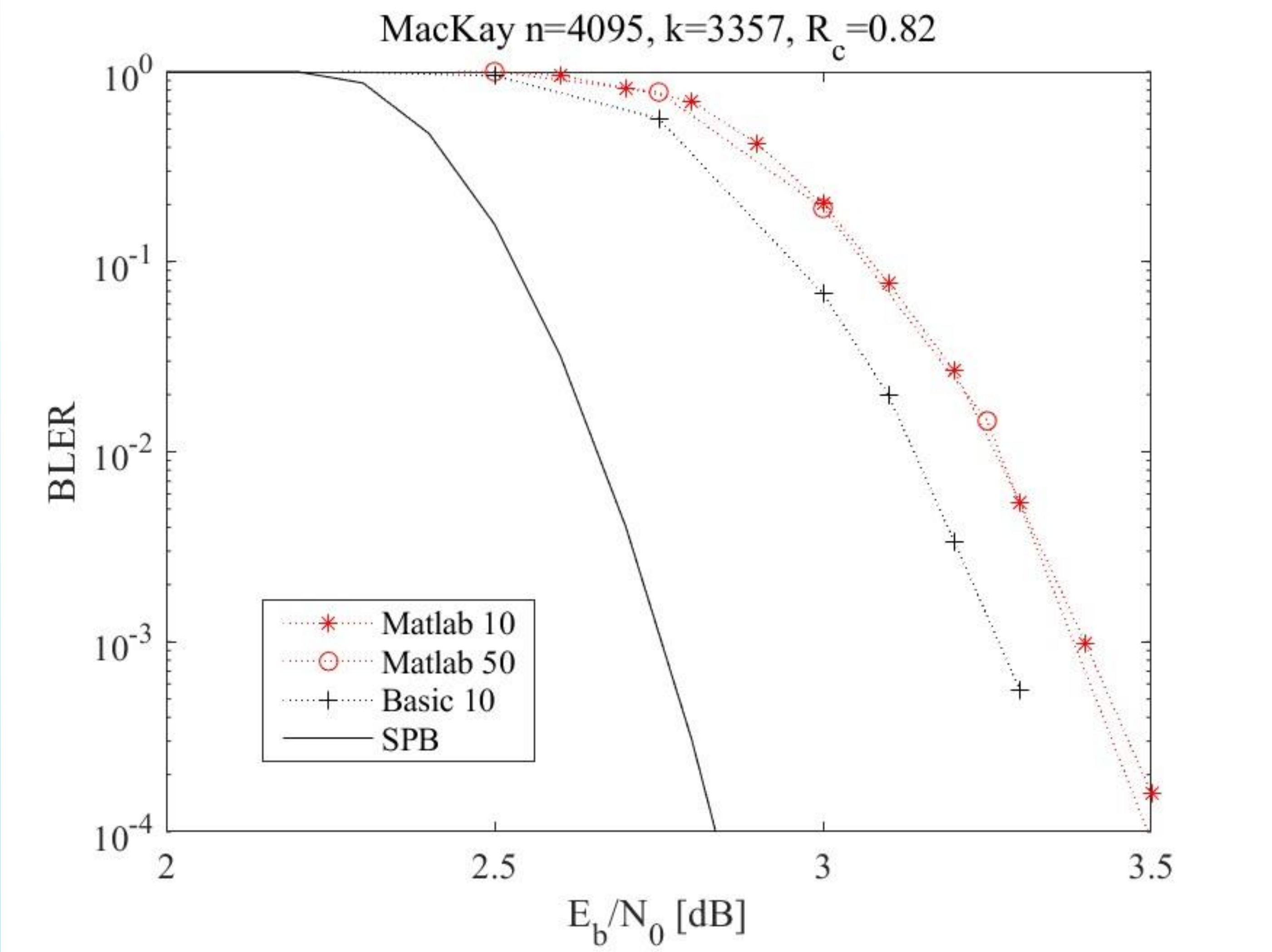
Decoding through message passing

- The algorithm continues iteratively until all the equations are satisfied, otherwise it stops after a given number of steps (error detected).

- Message passing is optimal if there are no cycles in the Tanner graph, otherwise it is suboptimal.

- The performance approach the optimal one if the girth is more than 4 (if there are no 4-cycles in the Tanner graph.

Performance example



No cycles of length 4

Single Parity Check Codes (I)

- Multi-dimensional product codes (M-SPC)
- Data are arranged in a hypercube of dimension *D* [12].
- The length of each dimension is k_i, corresponding to a encoded length $n_i = k_i + 1$. Before puncturing $R_c = \prod_{i=1}^{D} k_i / (k_i + 1)$
- Limited flexibility.
- Interleaving may be not used.
- Puncturing is effective.
- Good performance at average/low rates.

[12] M. Ranking and T. A. Gulliver, “Single parity check product codes”, IEEE-COM, Vol. 49, n. 8, Aug 2001, pp. 1354-1362.

Single Parity Check Codes (II)

Multiple parity-check codes (P-SPC, E-SPC)

- SPC: Data are arranged in a LC matrix.
  - P-SPC: P parity columns; $R_c = C/(C+P)$ [13].
  - E-SPC: added 1 parity row; $R_c \approx C/(C+P)$, for $L >> 1$.
- More flexibility (N depends on L; R_c does not depend on L).
- Interleaving is required (no interleaver gain).
- Puncturing is less effective.
- Good performance at high rates (best for $R_c=3/4$).

[13] J.S.K. Tee, D.P. Taylor, P. A. Martin, “Multiple serial and parallel concatenated single parity-check codes”, IEEE-COM, Vol. 51, n. 10, Oct 2003, pp. 1666-1675.

Design (short blocks)

PCCCs

Modulation	Rc	R	B	Thr. [dB]
BPSK	1/3	1/3	1286	-4.9
QPSK	1/3	2/3	642	-1.9
QPSK	1/2	1	428	0.6
8-PSK	1/2	3/2	285	3.5
16-QAM	1/2	2	214	5.6
8-PSK	3/4	9/4	191	7.7
16-QAM	3/4	3	143	9.8
64-QAM	2/3	4	107	13.2

SPCs

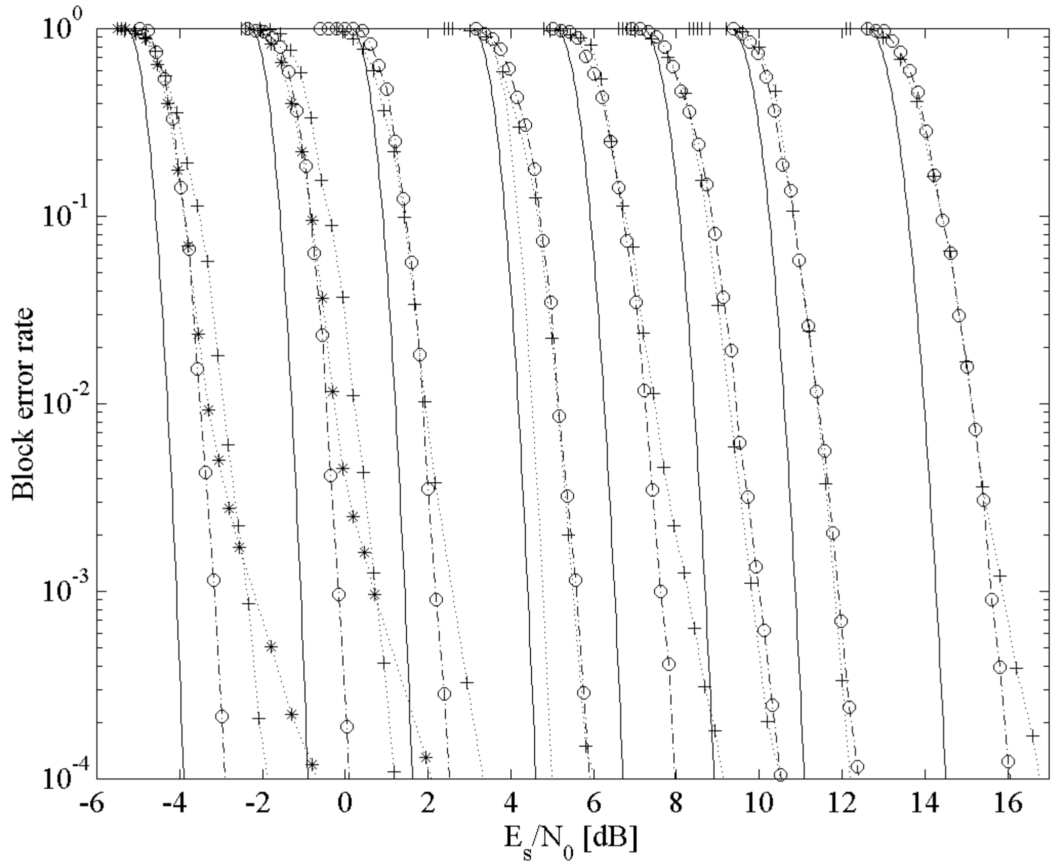
Modulation	Rc	R	Type	K	B	Thr. [dB]
BPSK	1/3	1/3	M	4,4,3,3,3	1296	-4.9
QPSK	1/3	2/3	M	4,4,3,3,3	648	-1.9
QPSK	1/2	1	E	65,7	455	0.6
QPSK	3/4	3/2	E	29,15	290	3.8
16-QAM	1/2	2	E	65,7	227	5.5
8-PSK	3/4	9/4	E	29,15	194	7.6
16-QAM	3/4	3	E	27,17	153	9.8
64-QAM	2/3	4	E	45,10	112	13.2

Before puncturing R_c=0.27

M-SPCs at low rates, E-SPCs at intermediate and high rates

AWGN: code performance

N=429/*R*. Bound (solid lines: PCCCs, dotted lines: SPCs); actual performance (dash-dotted lines with ‘o’PCCCs; dotted lines with ‘*’: M-SPCs; with ‘+’: E-SPCs)



Water-fall performance affects efficiency (goodput)
Error floor performance affects residual error rate

Design (long blocks)

PCCCs

Modulation	Rc	R	B	Thr. [dB]
BPSK	1/3	1/3	165	-5.2
QPSK	1/3	2/3	8183	-2.1
QPSK	1/2	1	5458	0.3
8-PSK	1/2	3/2	3639	3.1
16-QAM	1/2	2	2729	5.2
8-PSK	3/4	9/4	2426	7.3
16-QAM	3/4	3	1820	9.4
64-QAM	2/3	4	1365	12.8

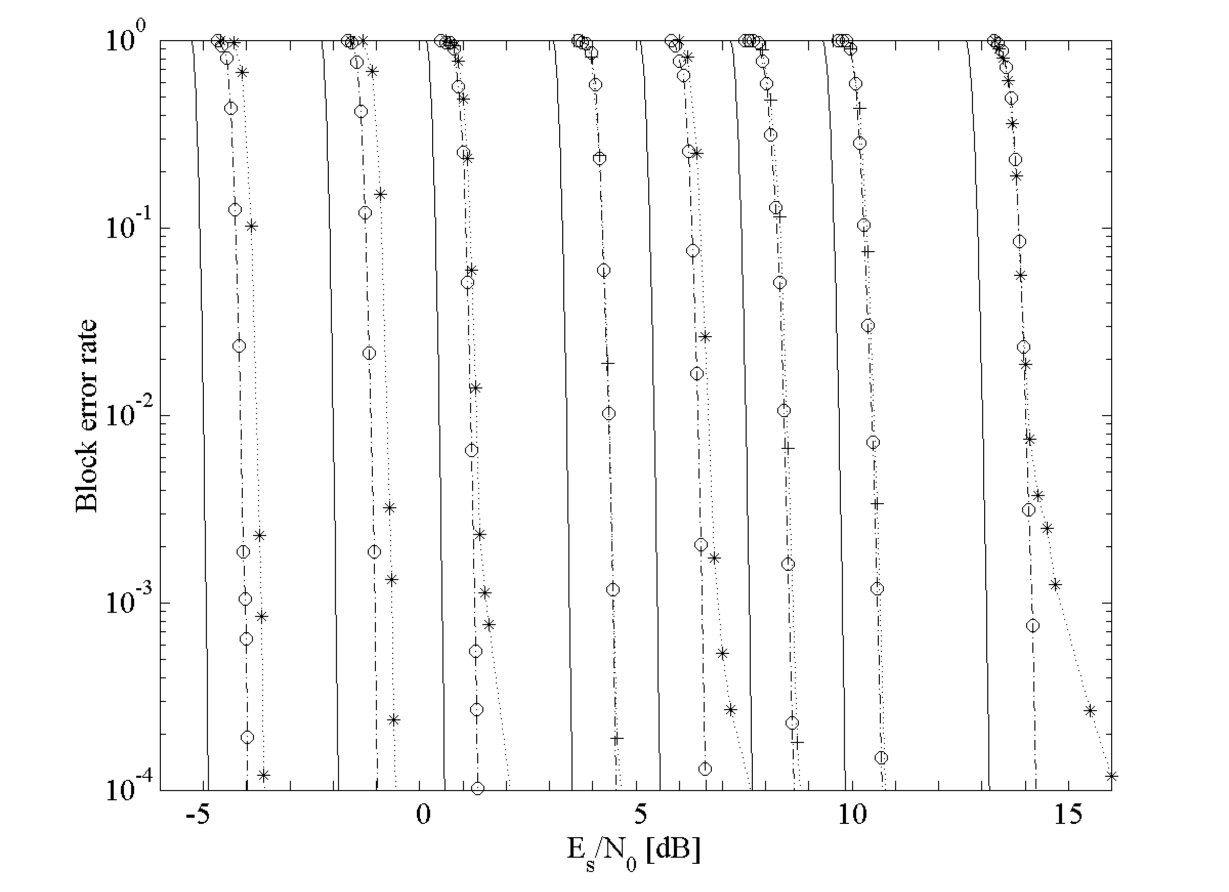
SPCs

Modulation	Rc	R	Type	K	B	Thr. [dB]
BPSK	1/3	1/3	M	5,4,4,4,4,4	15360	-5.2
QPSK	1/3	2/3	M	5,4,4,4,4,4	7680	-2.1
QPSK	1/2	1	M	6,6,5,5,5	4500	0.3
QPSK	3/4	3/2	E	303,18	3648	3.5
16-QAM	1/2	2	M	6,6,5,5,5	2250	5.3
8-PSK	3/4	9/4	E	303,18	2432	7.3
16-QAM	3/4	3	E	303,18	1824	9.4
64-QAM	2/3	4	M	11,9,8,7	1326	12.8

Boundary moves at higher rates for longer blocks

AWGN: code performance

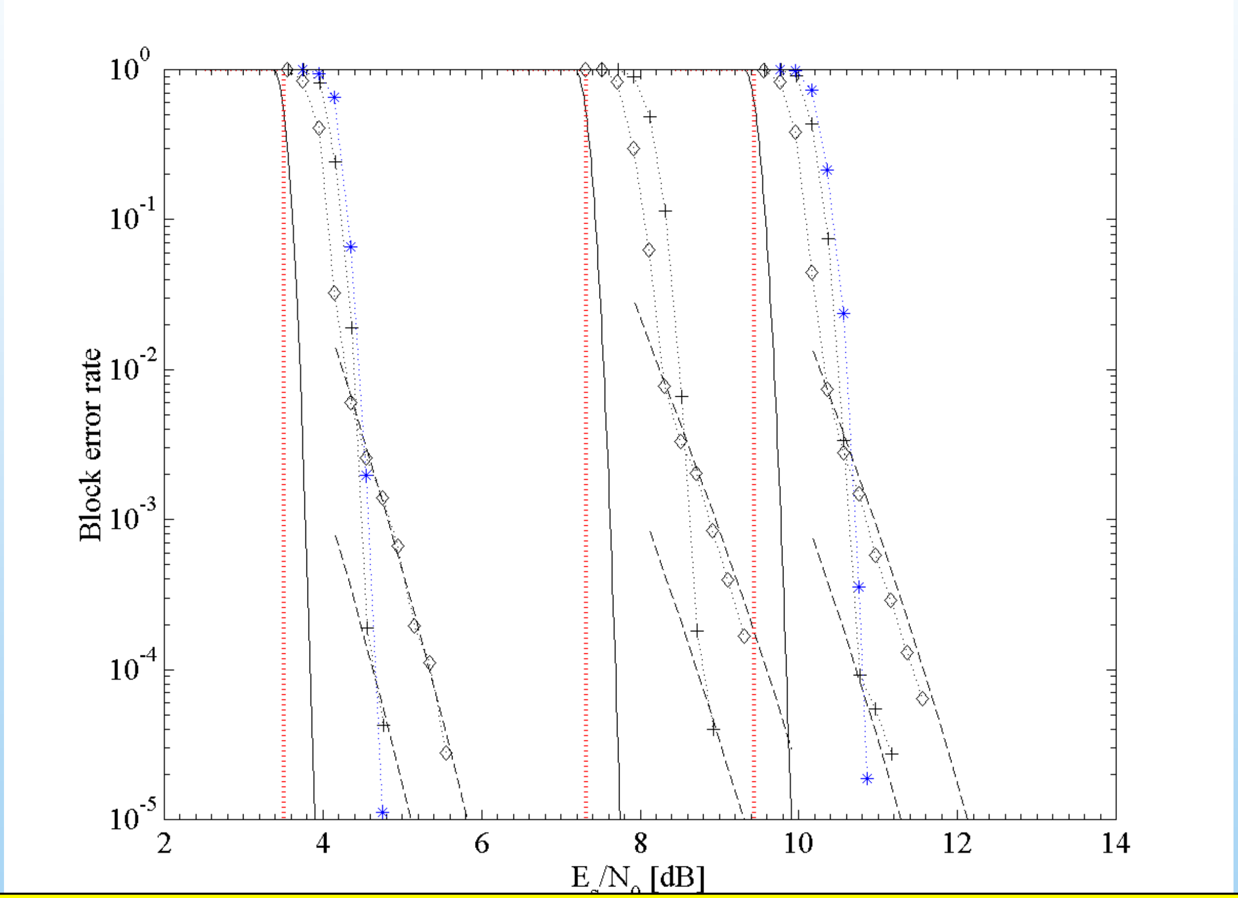
N=5000/*R*. Bound (solid lines: PCCCs, dotted lines: SPCs); actual performance (dash-dotted lines with ‘o’PCCCs; dotted lines with ‘*’: M-SPCs; with ‘+’: E-SPCs)



Slightly different water-fall performance (PCCCs and M-SPCs)
Improved error floor performance for E-SPCs

AWGN: SCCC code performance

R_c=3/4, QPSK, 8PSK 16QAM.
Dotted lines with '◊': best waterfall performance; with '+': chosen candidates.



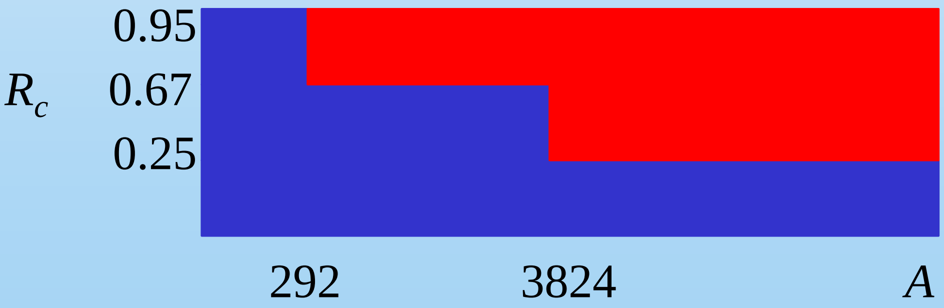
Residual error rate may be reduced
adopting serial versions (in blue)

Error Control for 5G: LDPC

- Name *A* the payload size.
- The information block, *B*, is obtained by adding to the payload *L* bits of CRC, being *L*=24 if *A*>3824, and *L*=16 otherwise,.
- The information block is then segmented in blocks of size *K*, for channel encoding.
- The rules for selecting among base graph 1 and base graph 2 are

– $A \leq 292$	base graph 2 (blue)
– $A \leq 3824$, $R_c \leq 0.67$	base graph 2
– $R_c \leq 0.25$	base graph 2
– Else	base graph 1 (red)

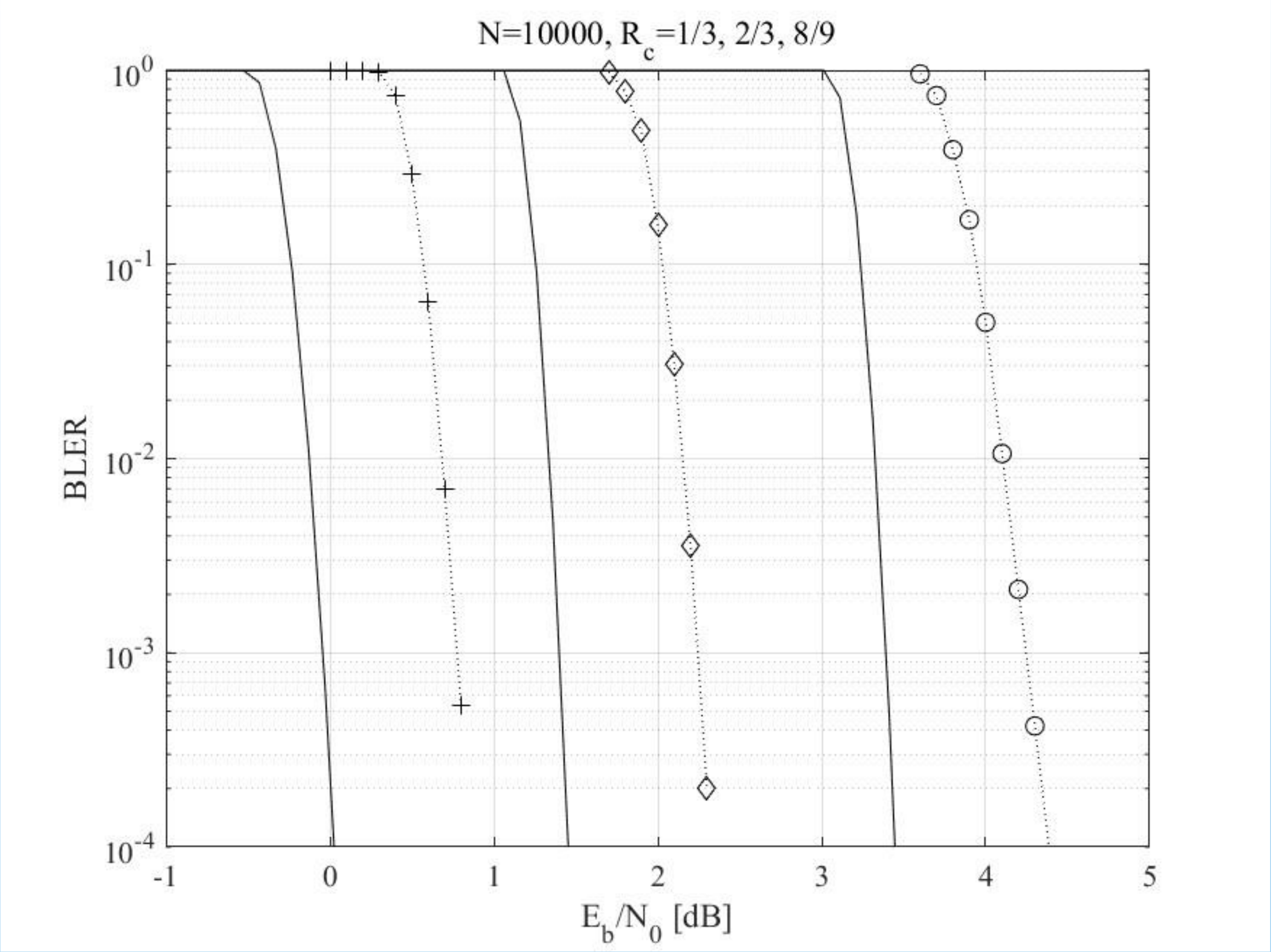
-



Error Control for 5G: LDPC

- Name N the number of channel bits per block.
- For base graph 1, K=22 Z_c, while N=66 Z_c. Thus, the basic rate is R_c=1/3.
- For base graph 2, K=10 Z_c, while N=50 Z_c. Thus, the basic rate is R_c=1/5.
- $N \leq 8448 = 22 \times 384$ for base graph 1 and $N \leq 3840 = 10 \times 384$ for base graph 2.
- Different rates are obtained through a Rate Matching algorithm.
- 5G supports hybrid-ARQ, by adopting incremental redundancy schemes.

Limiting performance



A comment

- LDPC limiting performance is close to the Shannon bound.
- Thus, they may be used instead of Turbo Codes which have a similar performance but are more complex.
- Are we satisfied?
- Not completely because we need reliable codes for short packets, and LDPC have a poor performance for short packets.
- But there is a solution: the Polar codes.

Error Control for 5G: Polar Codes (1)

- Polar codes [15] are error-correcting codes, which are able to achieve the capacity of binary-input memoryless symmetric (BMS) channels. This means that one can transmit at the highest possible rate over that class of channels. In addition, the encoding and decoding operations can be performed with low complexity, thanks to recursive techniques.

- Polar codes exploit channel polarization. More precisely, the BMS channel is characterized by the transition probability $W(y|x)$ (the probability that, having transmitted x, y was received). The polarization technique of channels consists in recursively building, starting from W, $N = 2^n$ binary input channels $\mathrm{W}_N^{(1)}$, …, $\mathrm{W}_N^{(\mathrm{N})}$. These channels are said to be polarized. They behave asymptotically as perfect channels or useless channels, allowing you to create a method of coding that sends information only through asymptotically perfect channels.

- [15] E. Arıkan, "Channel polarization: a method for constructing capacity-achieving codes for symmetric binary-input memoryless channels", IEEE Trans. Inf. Theory, vol. 55, no. 7, pp. 3051-3073, July 2009.

Error Control for 5G: Polar Codes (2)

- Formally, a specific polar code is fully defined by a 4-tuple (N, R_c, A, u_{Ac}) where:
  - N is the block length, i.e. the total number of bits transmitted over the channel.
  - R_c is the code rate, $R_c \in [0,1]$
  - A is the information set, $A \subset \{1,\ldots,N\}$ i.e. the set of positions which contains the information bits.
  - u_{Ac} are the frozen bits, $u_{Ac} \in \{0,1\}^{N(1-Rc)}$, i.e. bits which have fixed values, shared between the encoder and the decoder.
- The coding matrix has a recursive structure: $G_N = B_N G_2^{\otimes \log_2 N} = B_N G_2^{\otimes n}$

in which B_2 is a bit reordering matrix, $G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$, and $\otimes$ is the Kronecker product: $G_2 \otimes G_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$. After the reordering $G_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$

Error Control for 5G: Polar Codes (3)

- $x^N{}_1$ is transmitted over the channel W_N (which corresponds to N uses of the channel W), and $y^N{}_1$ is received.

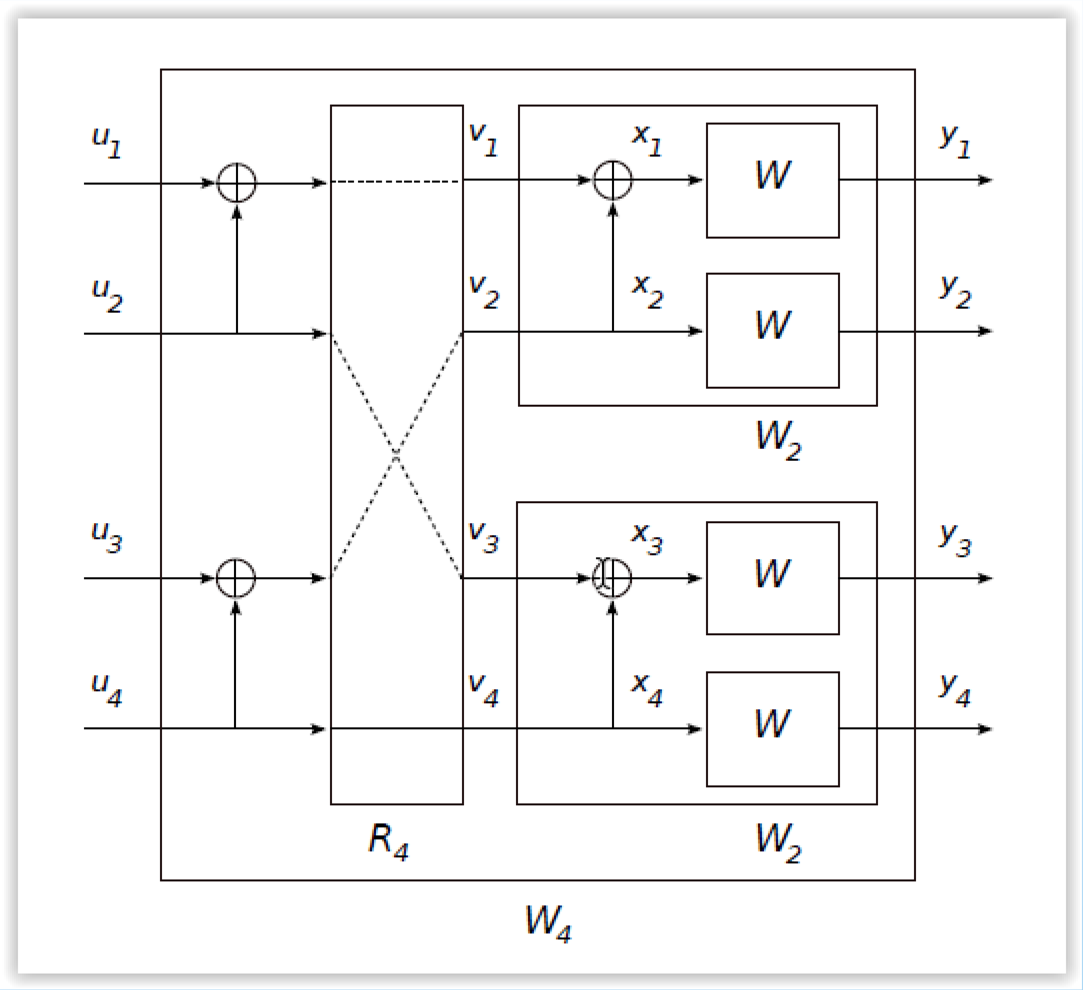
- From $y^N{}_1$ the successive cancellation (SC) decoder produces an estimate of $u^N{}_1$ (making also use of the frozen bits values).

- The complexity of this operation is O(N log N).

- Finally, only the components of u^N corresponding to information bits are kept, yielding u^A

Error Control for 5G: Polar Codes (4)

- Encoder and transmission process for *N*=4;



Capacity of individual channels

- Consider, for simplicity, the erasure channel with erasure probability *e*.

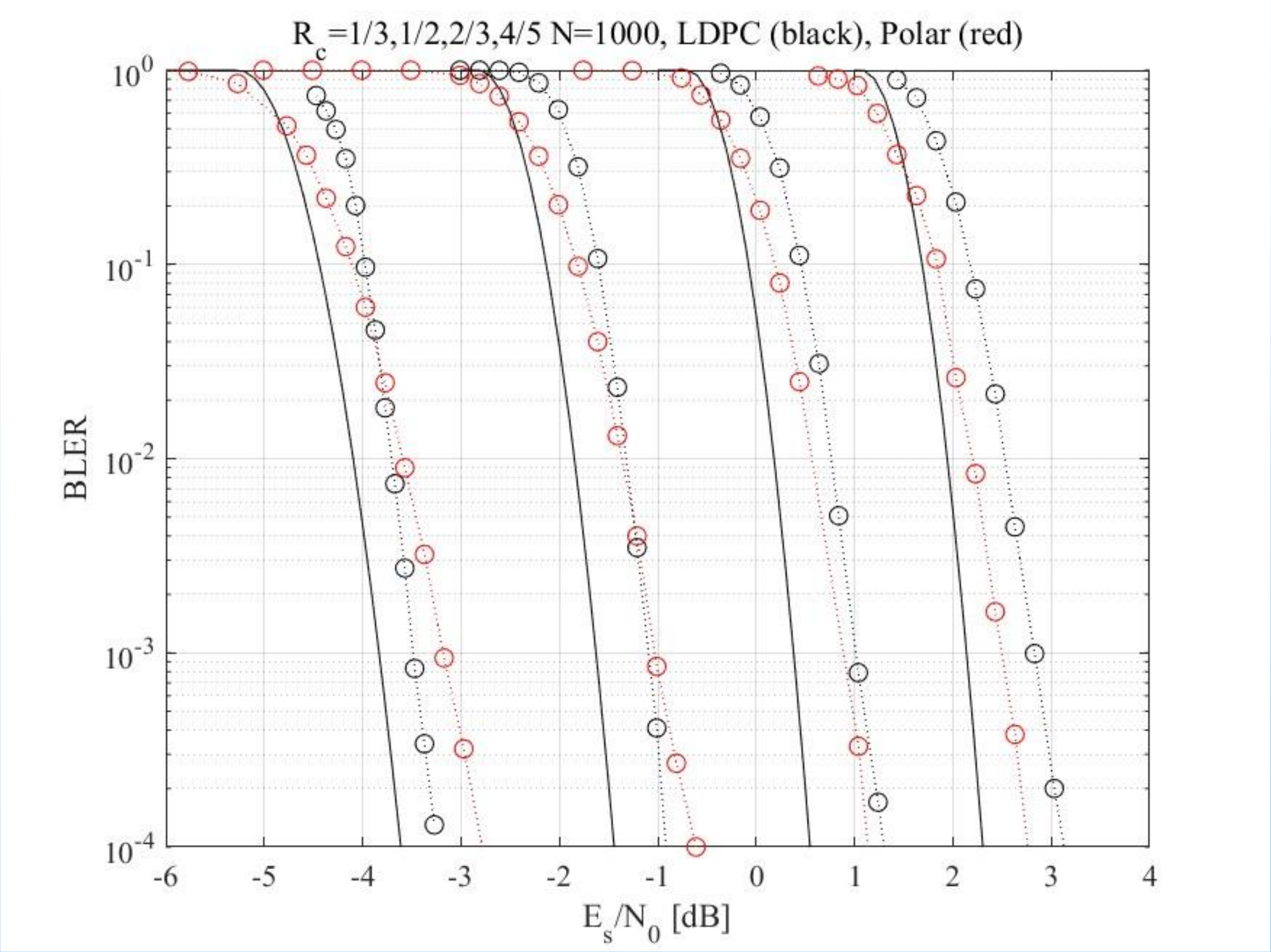
- At every iteration step, each channel of capacity C_n originates two channels of capacity respectively $2C_n e - C_n{}^2$ (the good one) and $C_n{}^2$ (the bad one), and the total capacity becomes $2C_n$.

- Example with e=0.5; C_1=1-e=0.5.

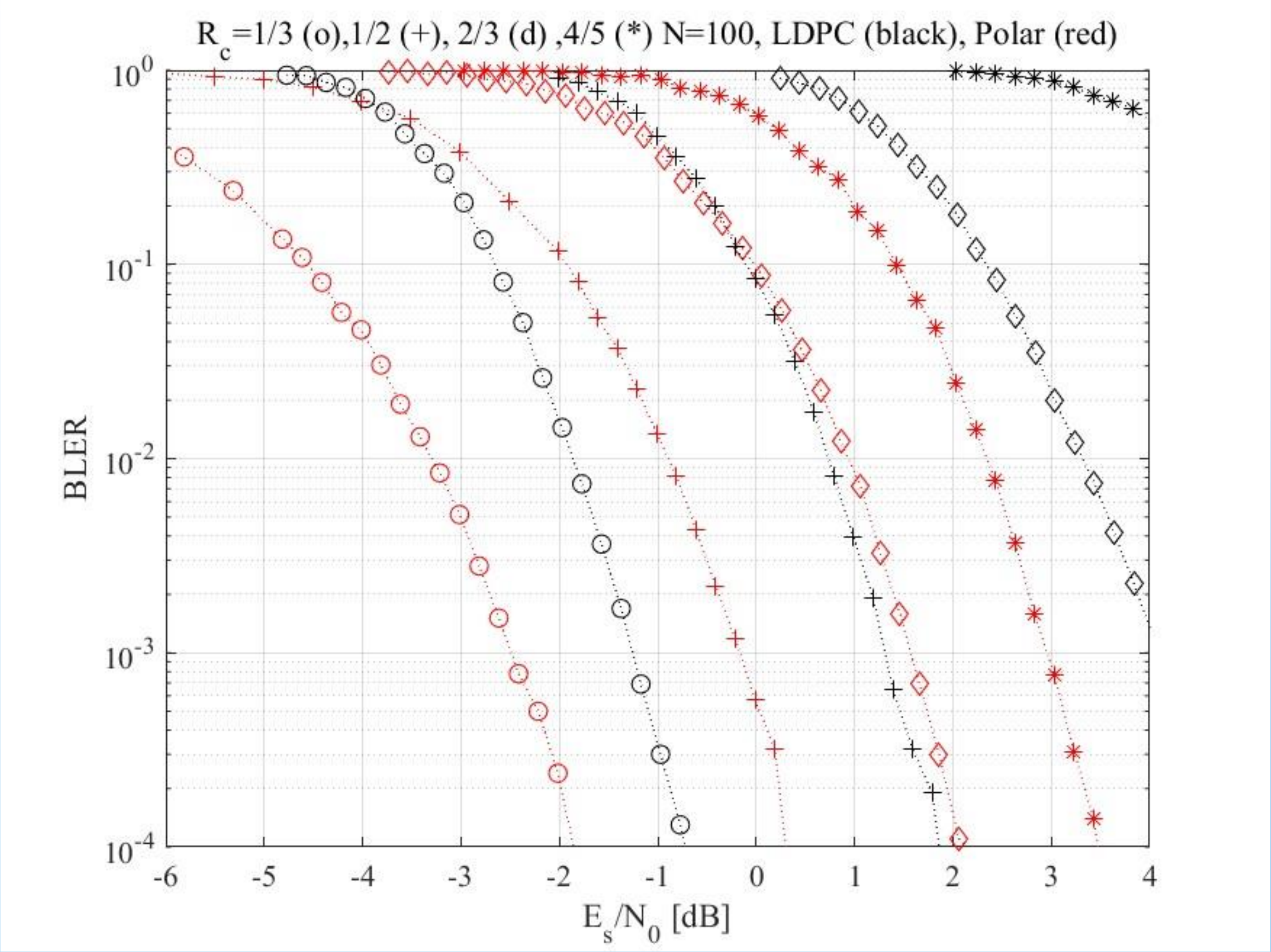
Capacity of individual channels

C_1	C_2	C_3	C_4
			0.99609375
		0.9375	
			0.87890625
	0.75		
			0.80859375
		0.5625	
			0.31640625
0.5			
			0.68359375
		0.4375	
			0.19140625
	0.25		
			0.12109375
		0.0625	
			0.00390625

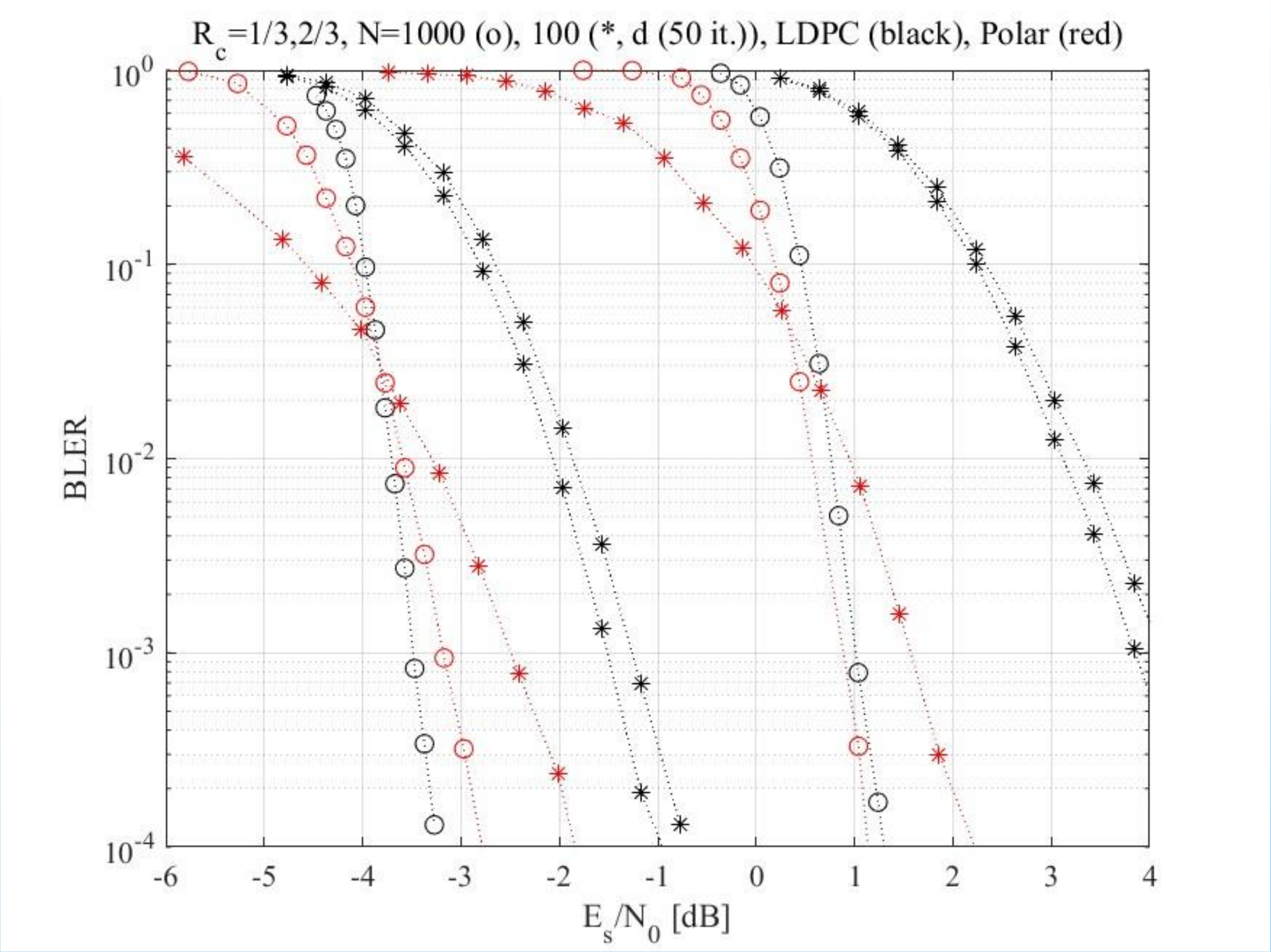
A performance comparison (1)



A performance comparison (2)



A performance comparison



A performance comparison (4)

- For medium length packets LDPC may have a slightly better performance, specially for the high SNR values.

- For short packets, polar codes are significantly better, specially for moderate block error rates.

- Increasing the number of iterations does not allow one to improve the performance of LDPC codes significantly,

- It may be concluded that polar codes are more suitable for IT applications.