EEE8099 – Information Theory & Coding
EEE8104 – Digital Communications

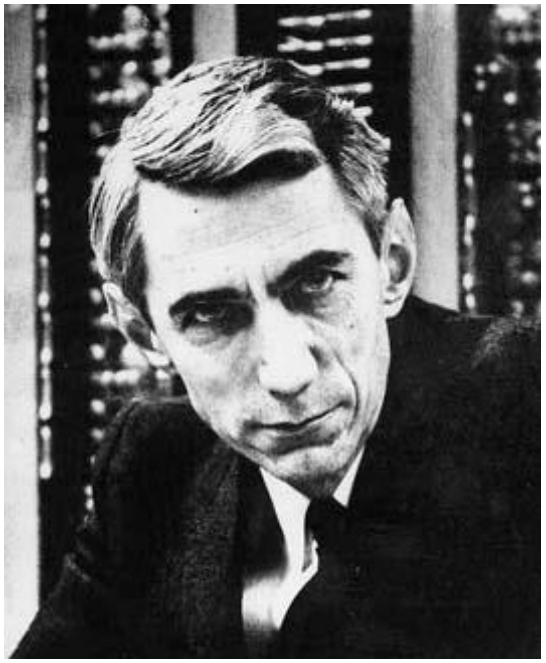
S. Le Goff
School of Engineering @ Newcastle University

Part 1
Introduction

"The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point"

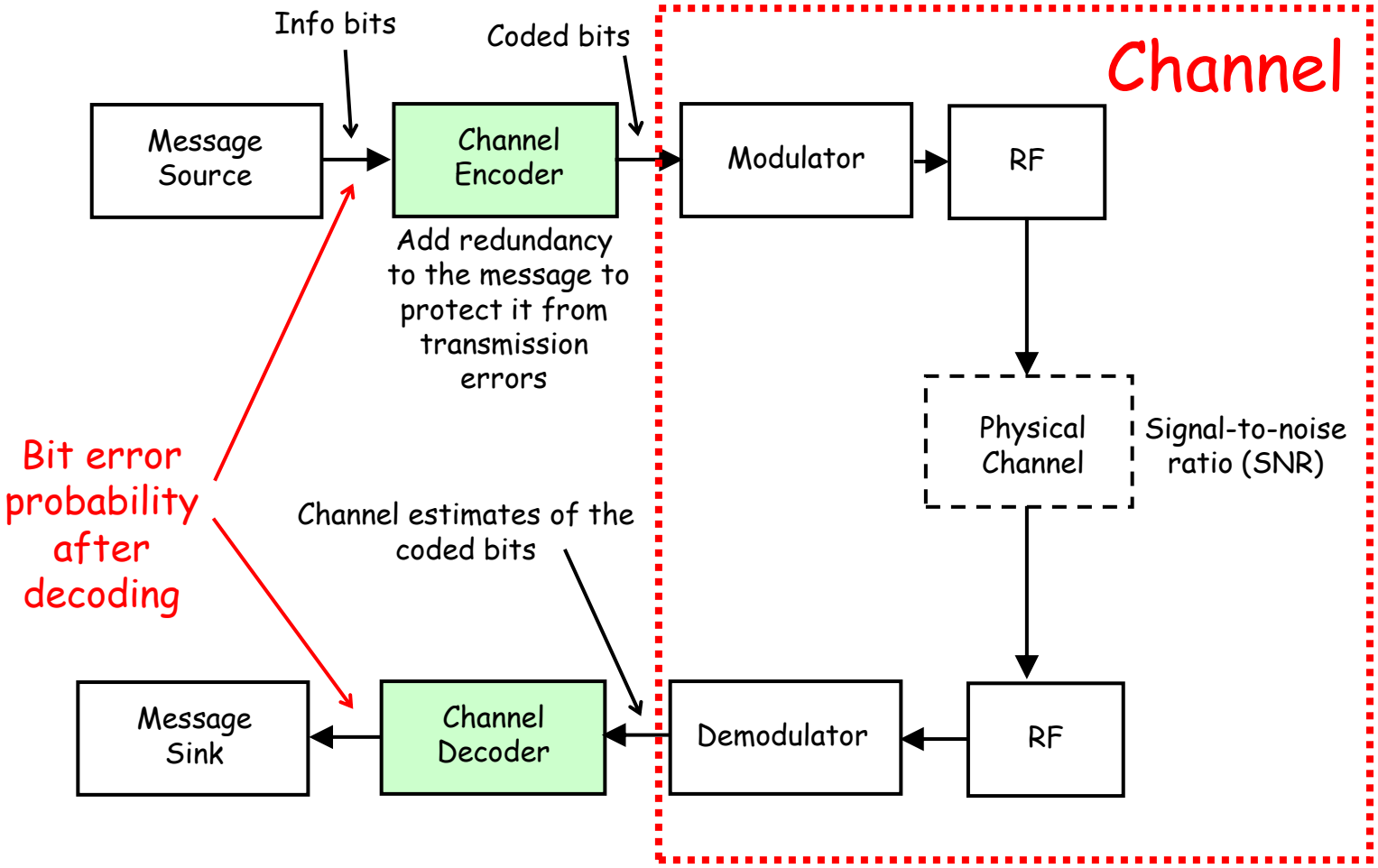
Claude E. Shannon, 1948.

To solve this problem, Shannon created a branch of applied mathematics: Information Theory.



Claude E. Shannon (1916-2001)

Digital communication systems



Part 2
Introduction to Channel Coding

Channel coding (error-correcting coding)

Channel coding adds redundancy to the message to be transmitted/stored.

It results in a loss of data rate for a given bandwidth OR an increase of bandwidth for a given data rate.

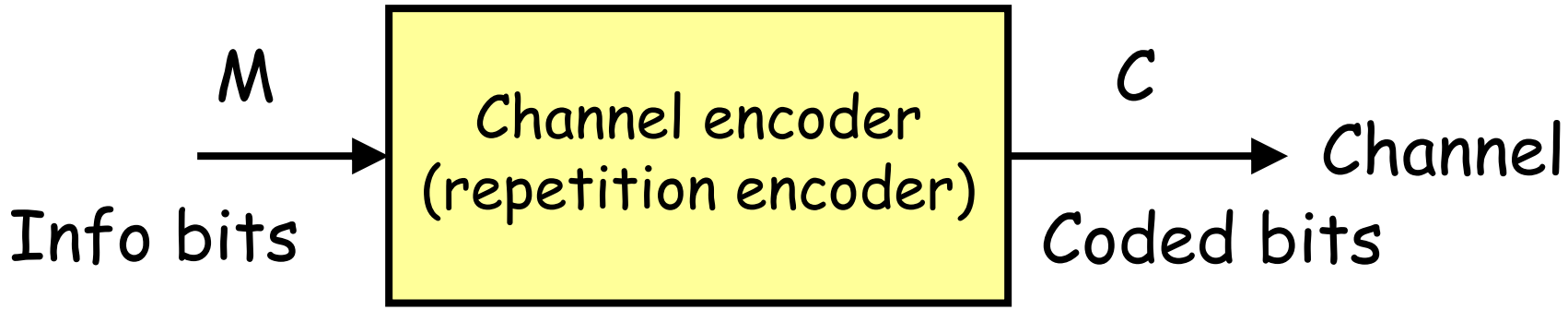
The "controlled" redundancy allows for detection or correction of transmission errors at the receiver side.

An example: Rate-1/3 repetition code

The info bit stream is encoded prior to transmission as follows:

$$M = 0 \rightarrow C = 000 \text{ and } M = 1 \rightarrow C = 111.$$

M is the "message" and C is the "codeword". The coding rate of this code is $R_c = \frac{1}{3}$.



An example: Rate-1/3 repetition code

Consider first an uncoded system.

We use the following notations:

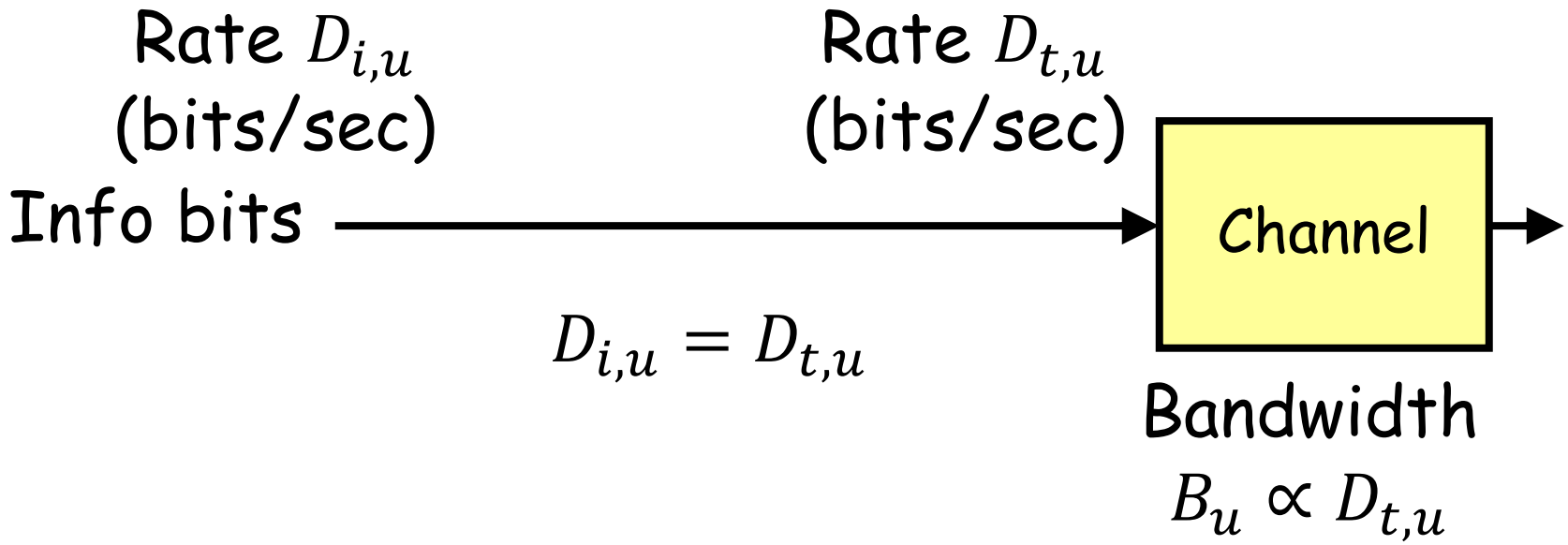
$D_{i,u}$ is the info bit rate, expressed in bits/sec;

$D_{t,u}$ is the rate at which bits are transmitted over the channel, also expressed in bits/sec;

B_u is the bandwidth required for the transmission of the modulated signal.

An example: Rate-1/3 repetition code

With an uncoded system, info bits are directly transmitted over the channel.



An example: Rate-1/3 repetition code

Consider now a coded system using the repetition code with $R_c = \frac{1}{3}$.

We use the following notations:

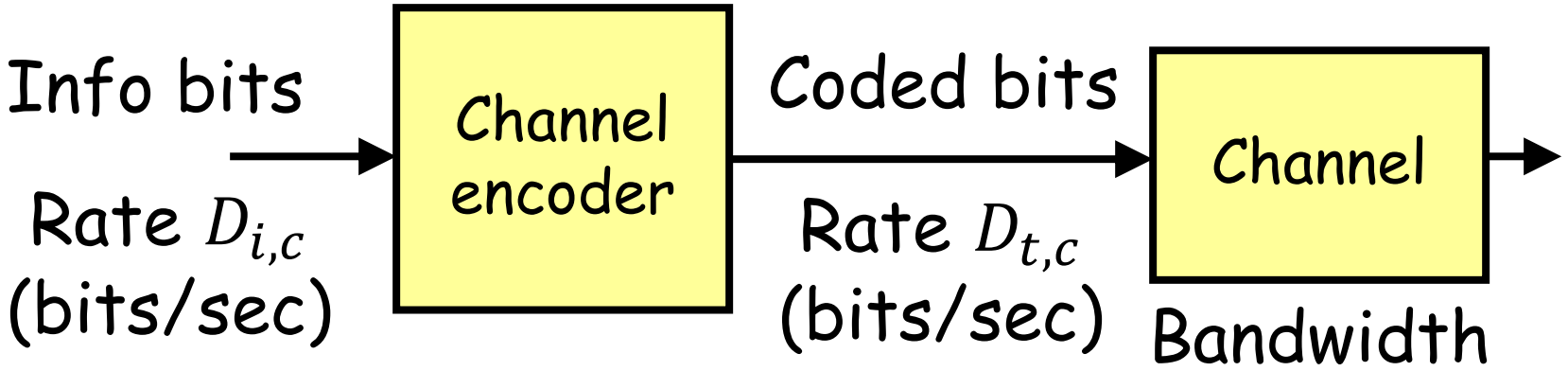
$D_{i,c}$ is the info bit rate, expressed in bits/sec;

$D_{t,c}$ is the rate at which bits are transmitted over the channel, also expressed in bits/sec;

B_c is the bandwidth required for the transmission of the modulated signal.

An example: Rate-1/3 repetition code

With a coded system, the bits transmitted over the channel are the coded bits, not the info bits.



An example: Rate-1/3 repetition code

Why can we write $D_{i,c} = R_c \cdot D_{t,c}$?

Consider an example: $R_c = \frac{1}{3}$, $D_{i,c} = 1$ Mbits/sec.

It takes $T_{i,c} = \frac{1}{D_{i,c}} = 10^{-6}$ sec to transmit one info bit.

Since there are $\frac{1}{R_c} = 3$ coded bits for each info bit, it also takes $T_{i,c} = 10^{-6}$ sec to transmit these $\frac{1}{R_c} = 3$ coded bits.

An example: Rate-1/3 repetition code

As a result, each coded bit must be transmitted in $T_{t,c} = \frac{T_{i,c}}{\frac{1}{R_c}} = R_c \cdot T_{i,c} = 10^{-6}/3$ sec.

The rate, $D_{t,c}$, at which the coded bits are transmitted over the channel is thus given by $D_{t,c} = \frac{1}{T_{t,c}} = \frac{1}{R_c \cdot T_{i,c}} = \frac{D_{i,c}}{R_c} = 3$ Mbits/sec.

We can finally write $D_{i,c} = R_c \cdot D_{t,c}$.

An example: Rate-1/3 repetition code

If the bandwidth required for the transmission of the RF signal must be the same for both uncoded and coded systems, i.e. $B_c = B_u$, then the info bit rate of the coded system must be 3 times lower than that of the uncoded system.

$$B_c = B_u \longrightarrow D_{t,c} = D_{t,u}$$

$$\downarrow$$

$$D_{i,u} = 3D_{i,c} \longleftarrow \frac{D_{i,c}}{R_c} = D_{i,u}$$

An example: Rate-1/3 repetition code

If the info bit rate must be the same for both uncoded and coded systems, i.e. $D_{i,u} = D_{i,c}$, then the bandwidth required by the coded system must be 3 times larger than that needed for the uncoded system.

$$D_{i,u} = D_{i,c} \longrightarrow D_{t,u} = R_c \cdot D_{t,c}$$

$$\downarrow$$

$$B_c = 3B_u \longleftarrow B_c = \frac{B_u}{R_c} \longleftarrow D_{t,c} = \frac{D_{t,u}}{R_c}$$

An example: Rate-1/3 repetition code

The bandwidth (i.e., the range of frequencies required for RF signal transmission) is a very valuable (and expensive) resource that must be shared among all users of the frequency spectrum.

Each user should use as little bandwidth as possible!

An example: Rate-1/3 repetition code

Whether we are talking about performance in terms of bandwidth or info bit rate, we reach the same conclusion: the use of channel coding is detrimental to this performance.

The performance deterioration depends entirely on the value of the coding rate R_c.

Try to keep R_c as close to the unit as possible!

An example: Rate-1/3 repetition code

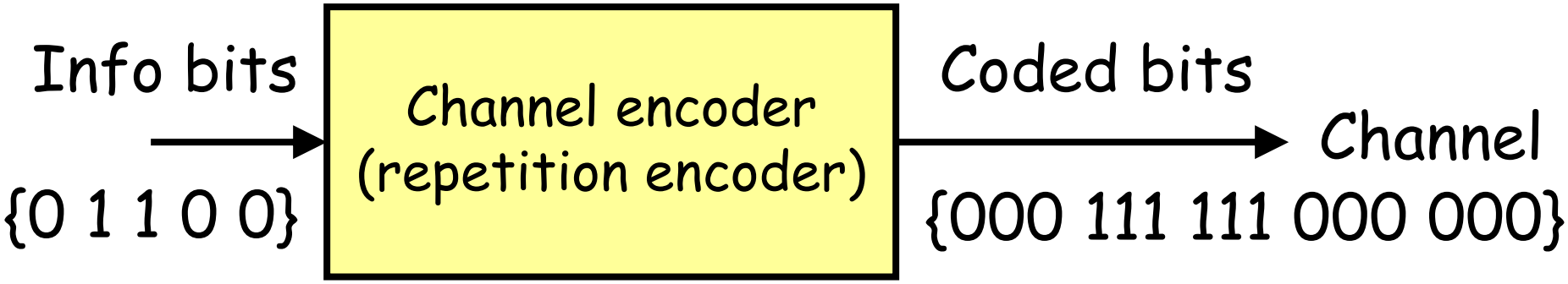
There is a natural trade-off between performance in terms of error correction and performance in terms of bandwidth/info bit rate.

Lower values of R_c tend to improve the error performance after decoding (more efficient error correction as more coded bits are transmitted for a given number of info bits) but degrade the performance in terms of bandwidth/info bit rate.

An example: Rate-1/3 repetition code

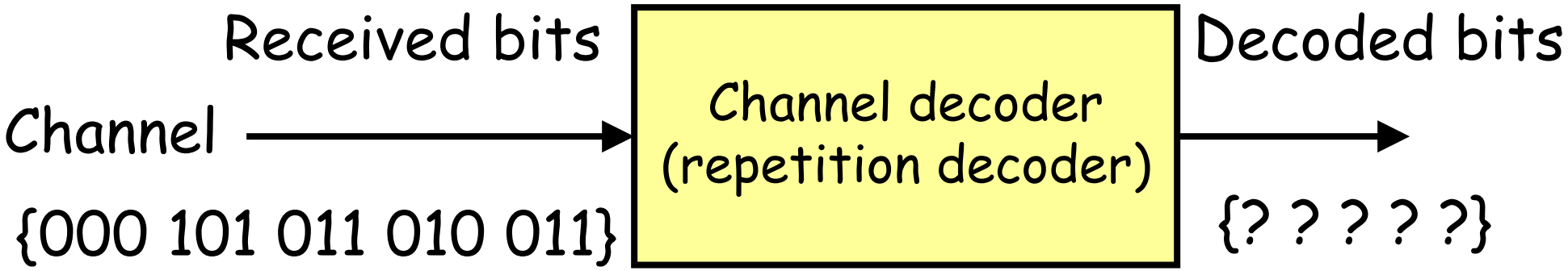
Ex: Sequence of info bits = {0 1 1 0 0}

→ Coded sequence = {000 111 111 000 000}.



An example: Rate-1/3 repetition code

Assume the corresponding received sequence is {000 101 011 010 011}. Find the decoded sequence.

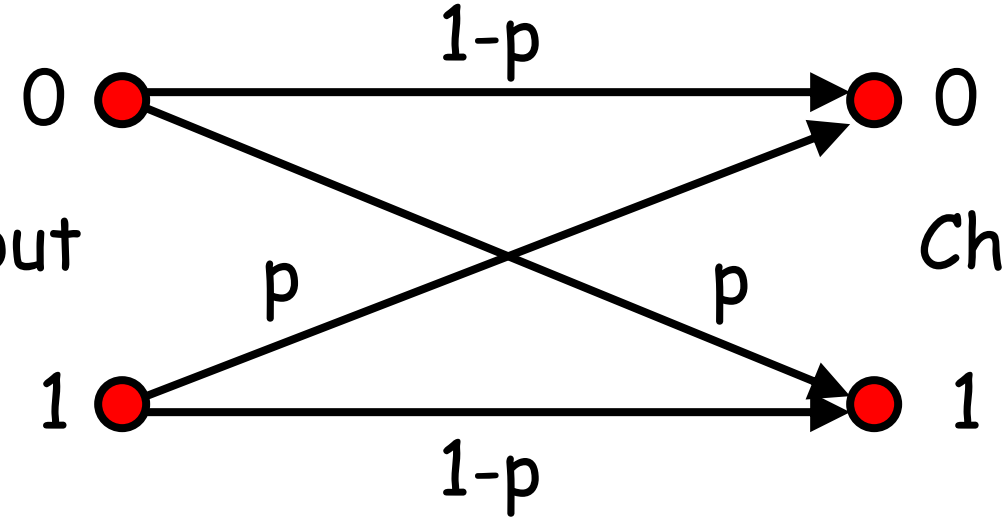


Answer:

An example: Rate-1/3 repetition code

We want to express the bit error probability after decoding, P_{eb}, as a function of the bit error probability before decoding, p.

Assume transmission over a binary symmetric channel (BSC):



p is the error probability over the channel.

An example: Rate-1/3 repetition code

Probability of wrong detection of the transmitted codeword:

$$p_{wd} = \Pr\{2 \text{ or } 3 \text{ errors in a received word of 3 bits}\}.$$

The events "2 errors in a received word of 3 bits" and "3 errors in a received word of 3 bits" are mutually exclusive.

We have $p_{wd} = \Pr\{2 \text{ errors in a received word of 3 bits}\} + \Pr\{3 \text{ errors in a received word of 3 bits}\} = \binom{3}{2}p^2(1-p) + \binom{3}{3}p^3$.

An example: Rate-1/3 repetition code

The binomial coefficient $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ is used to compute the number of possible permutations of k elements in a word of n elements.

Therefore, we can write $p_{wd} = 3p^2(1-p) + p^3$.

Assuming $p \ll 1$, we have $p_{wd} \sim 3p^2$.

However, we need to compute the bit error probability P_{eb} after decoding, not p_{wd}.

An example: Rate-1/3 repetition code

For the particular case of a repetition code, we are fortunate since $P_{eb} = p_{wd} \sim 3p^2$.

$$p = 10^{-2} \rightarrow P_{eb} \sim 3 \times 10^{-4}$$
$$p = 10^{-3} \rightarrow P_{eb} \sim 3 \times 10^{-6}$$
$$p = 10^{-4} \rightarrow P_{eb} \sim 3 \times 10^{-8}$$
$$p = 10^{-5} \rightarrow P_{eb} \sim 3 \times 10^{-10}$$

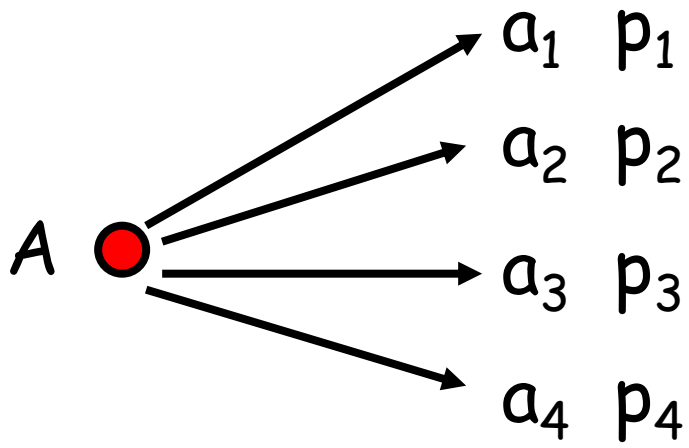
The use of a repetition code does seem to improve the transmission reliability, at a rather high cost in terms of info bit rate/bandwidth.

Part 3
Shannon Theory

Sources of information

An information source A consists of an alphabet of M symbols a_i.

Each symbol a_i has a probability $p_i = \Pr(a_i)$ to be generated by A.



We must always have $\sum_{i=1}^{M} p_i = 1$.

Information generated by a source

Shannon: Information is a measurable quantity with a precise definition.

When a symbol a_i with probability p_i is generated by A, the amount of info produced by A is

$I_i = \log_2\left(\frac{1}{p_i}\right) = -\log_2(p_i).$ Unit: Shannon (or bit)

We use the Shannon rather than the bit as unit of info in order to avoid confusion between unit of info and binary digit.

Information generated by a source

Remember that $\log_2(x) = \frac{\log_{10}(x)}{\log_{10}(2)} = \frac{\ln(x)}{\ln(2)}$ when using your calculators, i.e. $\log_2(x) \sim 3.322 \times \log_{10}(x)$ and $\log_2(x) \sim 1.443 \times \ln(x)$.

Generation of a highly probable symbol contains little info. For example, if p_i = 0.999, then the info conveyed is $I_i \approx 0.00144$ Shannon.

Generation of a highly improbable symbol provides a lot of info. For example, if p_i = 0.001, then the info conveyed is $I_i \approx 9.96578$ Shannon.

Entropy of a source of information

The entropy of source A is the average amount of info generated by A:

$$H(A) = E\{I_i\} = E\{-\log_2(p_i)\},$$

where $E\{\cdot\}$ denotes the expected value of "$\cdot$".

It is computed using

$$H(A) = \sum_{i=1}^{M} p_i I_i = -\sum_{i=1}^{M} p_i \log_2(p_i)\,.$$

Ex: A source that can generate three symbols a_1, a_2, a_3 with associated probabilities $p_1 = 0.6$, $p_2 = p_3 = 0.2$ has an entropy of 1.371 Shannon.

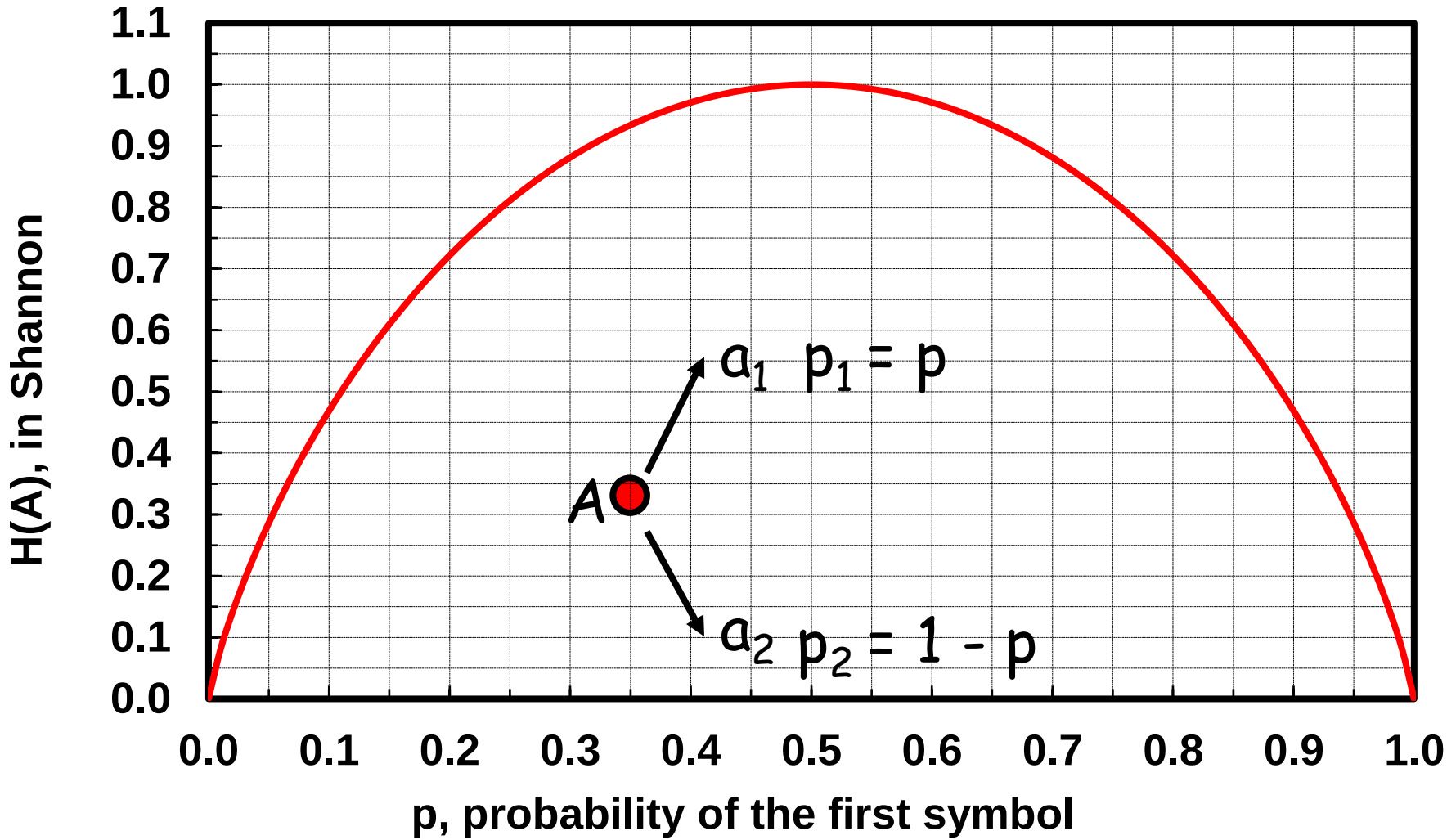
Entropy of a source of information

We could show that the entropy of a source is maximal when the M symbols a_i have equal probabilities, i.e. $p_1 = p_2 = p_3 = \cdots = p_M = \frac{1}{M}$.

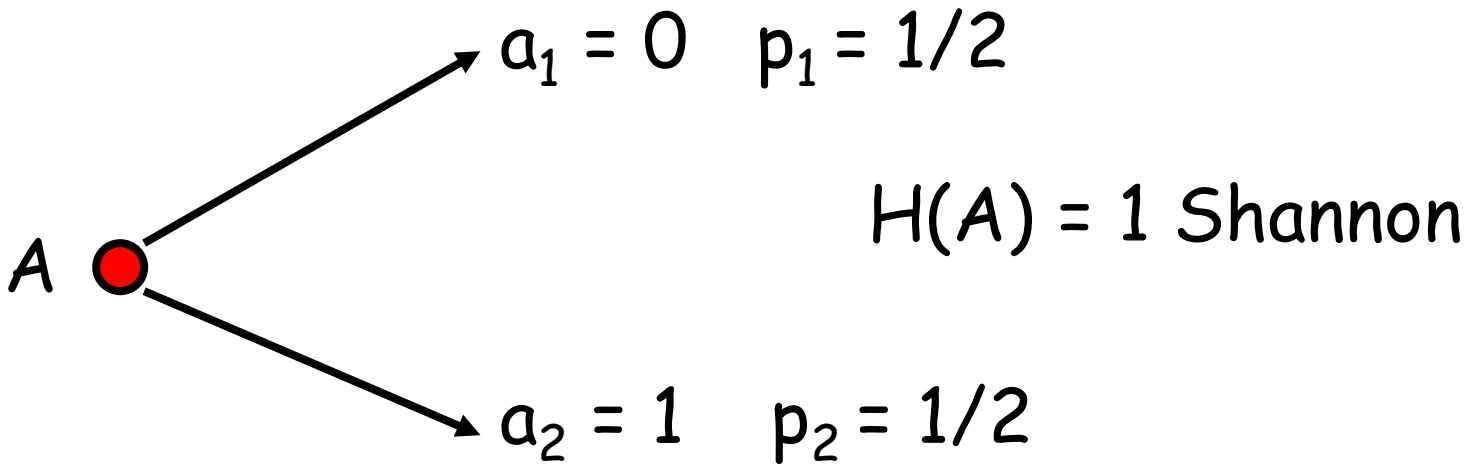
The maximal value of H(A) is then given by $H_{max}(A) = -\sum_{i=1}^{M} \frac{1}{M} \log_2\left(\frac{1}{M}\right) = -\frac{1}{M} \log_2\left(\frac{1}{M}\right) \sum_{i=1}^{M} 1 = -\frac{1}{M} \log_2\left(\frac{1}{M}\right) M = -\log_2\left(\frac{1}{M}\right) = \log_2(M)$.

Entropy of a binary source

Entropy of a 2-symbol source with probabilities (p, 1-p)



Entropy of a binary source



The amount of info contained in a sequence of k <u>independent</u> bits, with $\Pr(0) = \Pr(1) = \frac{1}{2}$, is equal to k Shannons.

If those bits were not independent, this amount would be less than k Shannons.

Joint entropy of two sources

Consider two sources A (a_i, p_i, M_A) and B (b_j, p_j, M_B).

The joint entropy H(A, B) is the average amount of info jointly generated by A and B, i.e. the total amount of info contained in both sources A and B.

It is thus defined as $H(A,B) = E\{-\log_2(p_{i,j})\}$,

It is computed using

$$H(A,B) = -\sum_{i=1}^{M_A}\sum_{j=1}^{M_B} p_{i,j}\log_2(p_{i,j}),$$

where $p_{i,j} = \Pr(a_i \cap b_j)$ denotes the probability that a_i is generated by A and b_j is generated by B.

Joint entropy of two sources

We can derive an interesting expression of the joint entropy as follows:
$H(A,B) = E\{-\log_2(p_{i,j})\} = E\{-\log_2(p_{j|i}p_i)\}$, by using Bayes' rule.

Thus, we have $H(A,B) = E\{-\log_2(p_{j|i}) - \log_2(p_i)\} = E\{-\log_2(p_{j|i})\} + E\{-\log_2(p_i)\} = H(B|A) + H(A)$, where $p_{j|i} = \Pr(b_j|a_i)$ denotes the probability that b_j is generated by B given the knowledge that a_i has been generated by A.

Joint entropy of two sources

$$H(A,B) = H(B|A) + H(A)$$

In this expression, $H(B|A)$ is defined as the conditional entropy of B given A, and represents the average amount of info provided by B when the outcome of A is known.

In the same way, we could also show that
$H(A,B) = H(A|B) + H(B)$,
which implies that $H(B|A) + H(A) = H(A|B) + H(B)$.

Joint entropy of two sources

If A and B are independent sources, we can write $H(A,B) = H(B) + H(A)$ as $H(B|A) = H(B)$ in this case. The outcome of A does not give us any clue about the outcome of B, and vice versa.

If source A completely defines source B, we have $H(A,B) = H(A)$ as $H(B|A) = 0$ in this case. The outcome of A tells us everything about the outcome of B.

Example of two independent sources

Source A: In what city will Jenny study next year?

a_1: London $p_1 = 0.25$

a_2: Newcastle upon Tyne $p_2 = 0.25$

a_3: Glasgow $p_3 = 0.25$

a_4: Edinburgh $p_4 = 0.25$

H(A) = 2 Shannons

Example of two independent sources

Source B: Who will be the next US President?

b_1: Donald Trump $p_1 = 0.5$

b_2: Bernie Sanders $p_2 = 0.5$

H(B) = 1 Shannon

Joint entropy H(A, B) = H(A) + H(B) = 3 Shannons

Example: A completely defines B

Source A: In what city will Jenny study next year?

a_1: London $p_1 = 0.25$

a_2: Newcastle upon Tyne $p_2 = 0.25$

a_3: Glasgow $p_3 = 0.25$

a_4: Edinburgh $p_4 = 0.25$

H(A) = 2 Shannons

Example: A completely defines B

Source B: In what country will Jenny study next year?

b_1: England $p_1 = 0.5$

b_2: Scotland $p_2 = 0.5$

H(B) = 1 Shannon

Joint entropy H(A, B) = H(A) = 2 Shannons

Mutual information of two sources

How much does a source A tell us about another source B ?

Mutual information $I(B;A) = H(B) - H(B|A).$

Amount of info in source B

Amount of info provided by B when the outcome of A is known

Crucial parameter to assess the quality of a transmission channel.

Mutual information of two sources

The value of $I(B;A)$ indicates how much the channel output A can tell us about the channel input B.

In other words, the quality of a channel is measured by the mutual info $I(B;A) = H(B) - H(B|A)$.

Ex: $I(B;A) = 0$ when $H(B|A) = H(B)$ → Useless channel.

Ex: $I(B;A) = H(B)$ when $H(B|A) = 0$ → Noiseless (i.e., perfect) channel.

Mutual information of two sources

How to express $I(B;A)$ as a function of probabilities?

$$I(B;A) = H(B) - H(B|A) = E\{-\log_2(p_j)\} - E\{-\log_2(p_{j|i})\} = E\{-\log_2(p_j) + \log_2(p_{j|i})\} = E\left\{\log_2\left(\frac{p_{j|i}}{p_j}\right)\right\}.$$

Therefore, $I(B;A)$ can be computed using the following expression:

$$I(B;A) = \sum_{i=1}^{M_A}\sum_{j=1}^{M_B} p_{i,j}\log_2\left(\frac{p_{j|i}}{p_j}\right).$$

Mutual information of two sources

We notice that

$$I(B;A) = E\left\{\log_2\left(\frac{p_{j|i}}{p_j}\right)\right\} = E\left\{\log_2\left(\frac{p_{j|i}p_i}{p_j p_i}\right)\right\} =$$
$$E\left\{\log_2\left(\frac{p_{i,j}}{p_i p_j}\right)\right\} = E\left\{\log_2\left(\frac{p_{i|j}p_j}{p_i p_j}\right)\right\} = E\left\{\log_2\left(\frac{p_{i|j}}{p_i}\right)\right\}.$$

This result shows that $I(B;A) = I(A;B)$.

Part 4
Channel Capacity
and
Shannon's Capacity Theorem

Communication channels

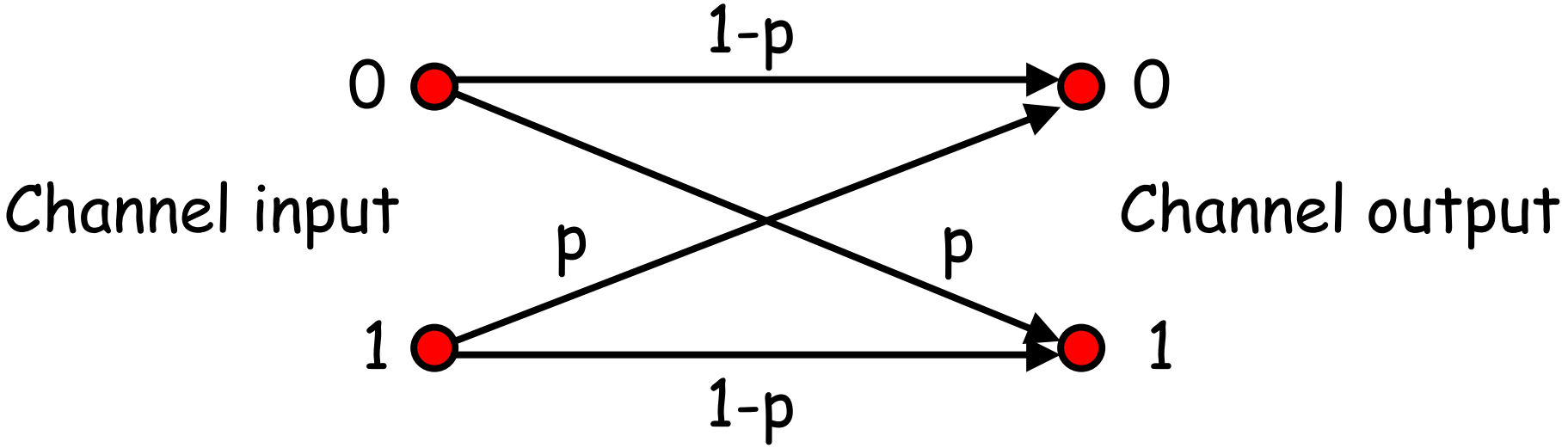
We consider, throughout this module, two types of channel:

1. The binary symmetric channel (BSC);

2. The BPSK (binary phase shift keying), AWGN (additive white Gaussian noise) channel.
= the standard channel in communication theory

Binary symmetric channel (BSC)

BSC model:



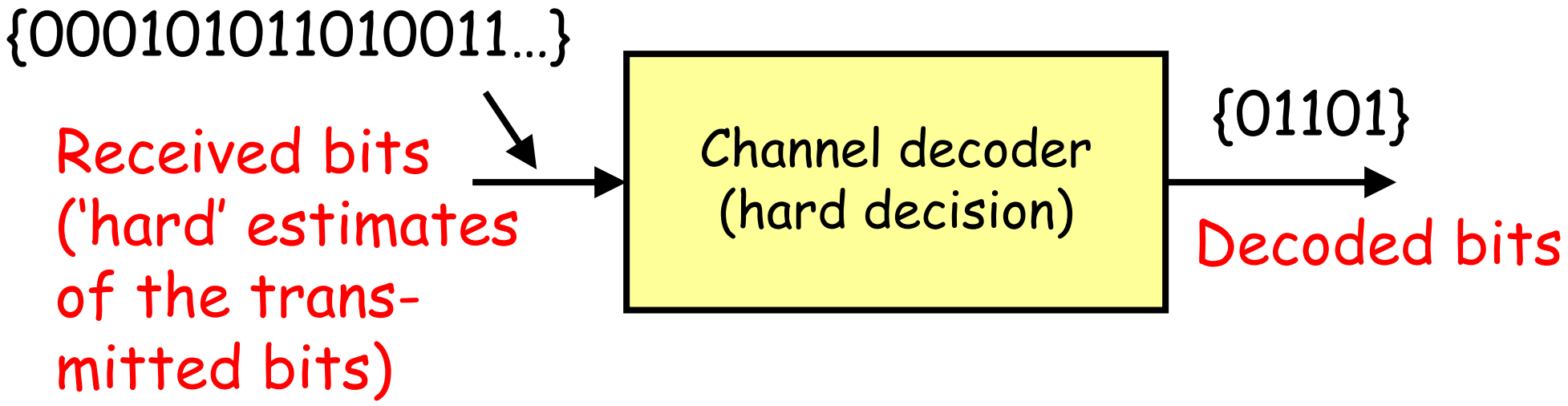
p is the error probability over the channel, with $p < 0.5$.

At the channel input, we have $\Pr(0) = \Pr(1) = \frac{1}{2}$.

Binary symmetric channel (BSC)

The channel output is a sequence of bits that are estimates of the transmitted bits.

In the case of a coded system, the channel decoder processes these 'hard' estimates.



BPSK, AWGN channel

BPSK, AWGN channel model:

Channel output: $r = s + n$, where $s = +1$ or -1 is the transmitted bit with $\Pr\{s = +1\} = \Pr\{s = -1\} = \frac{1}{2}$.

For an uncoded system, s is an info bit.

For a coded system, s is a coded bit (generated by the channel encoder).

For this channel, we use the (+1, -1) notation rather than the traditional (0, 1) notation.

BPSK, AWGN channel

Remember the way BPSK modulation schemes work:

- To transmit a '0', we actually transmit the RF signal $s_0(t) = \cos(2\pi f_0 t) = (+1)\cdot \cos(2\pi f_0 t)$;

- To transmit a '1', we transmit the RF signal $s_1(t) = \cos(2\pi f_0 t + \pi) = -\cos(2\pi f_0 t) = (-1)\cdot\cos(2\pi f_0 t)$.

If we ignore the carrier wave (which is irrelevant in info theory), we can say that we simply transmit a bit which is either equal to +1 or -1

BPSK, AWGN channel

In the equation r = s + n, the quantity n is a Gaussian noise sample.

In other words, n is a random real number that follows a Gaussian distribution (i.e., has a Gaussian probability density function, PDF).

A study of the physical layer of a BPSK modulation scheme shows that

- the mean m of the noise sample is given by $m = E\{n\} = 0$,
- its variance $\sigma^2 = E\{n^2\} - m^2$ is given by $\sigma^2 = \frac{1}{2\frac{E_S}{N_0}}$.

BPSK, AWGN channel

The quantity $\frac{E_S}{N_0}$ represents the signal-to-noise ratio per transmitted bit.

In this ratio, E_s denotes the energy allocated to each bit transmitted over the channel, whereas N_0 is the power spectral density of the white Gaussian noise process present over the physical channel.

Another SNR definition that is also frequently used is the ratio $\frac{E_b}{N_0}$.

BPSK, AWGN channel

The ratio $\frac{E_b}{N_0}$ represents the signal-to-noise ratio per info bit. In this ratio, E_b denotes the energy allocated to each info bit.

In an uncoded system, the info bits are directly transmitted over the channel. So, we have $E_b = E_s$, and thus $\frac{E_b}{N_0} = \frac{E_s}{N_0}$.

In a coded system, the bits that are transmitted over the channel are the coded bits, not the info bits.

BPSK, AWGN channel

In this case, $\frac{E_S}{N_0}$ is the SNR per coded bit, and is thus different from the ratio $\frac{E_b}{N_0}$.

The equation that links the SNRs $\frac{E_S}{N_0}$ and $\frac{E_b}{N_0}$ in a coded system will be derived later on.

The noise sample n has a Gaussian probability density function.

BPSK, AWGN channel

A random variable X is said to be Gaussian if it has a probability density function (PDF) given by

$$P_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(x-m)^2}{2\sigma^2}\right),$$

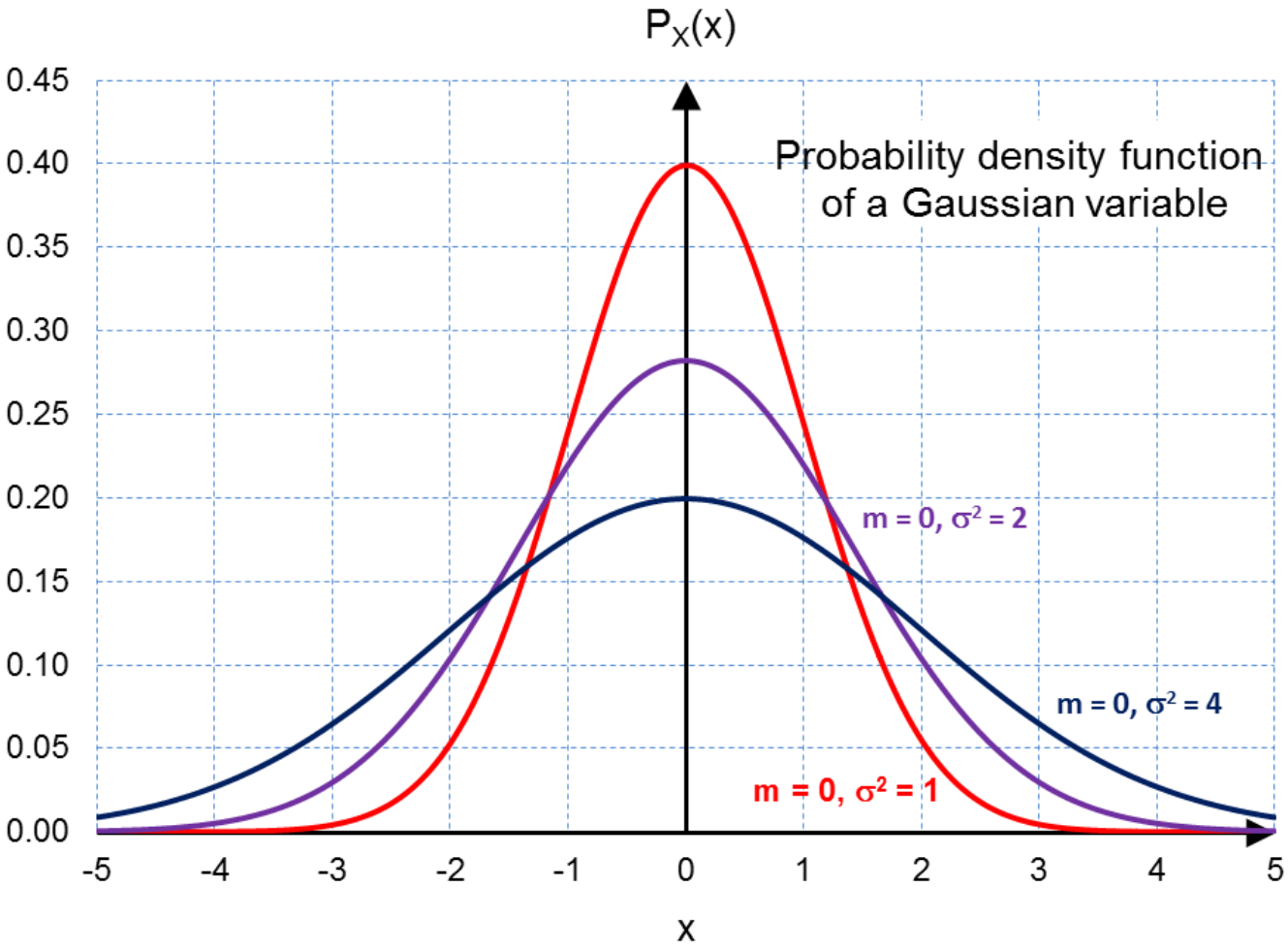
where m is the mean of X and σ^2 is its variance.

In the case of our noise sample n, its PDF can thus be written as

$$P(n) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{n^2}{2\sigma^2}\right),$$

since its mean is equal to 0.

BPSK, AWGN channel



BPSK, AWGN channel

What is the PDF of the channel sample r ?

The answer depends on whether or not we make an assumption on the value of the transmitted bit s:

- Assumption on the value of s: conditional PDF $P(r|s)$;
- No assumption on the value of s : PDF $P(r)$.

Assume first that a particular bit $s = +1$ or -1 has been transmitted. Since $r = s + n$, the channel sample is also Gaussian with a mean $m = E\{r\} = E\{s + n\} = E\{s\} + E\{n\} = s$, and a variance equal to that of n.

BPSK, AWGN channel

As a result, the probability density function of the channel sample r is

$$P(r|s) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(r-s)^2}{2\sigma^2}\right),$$

where s is the transmitted bit.

Let us turn our attention to the case where no assumption is made on the value of the transmitted bit s.

The probability density function $P(r)$ can be derived as follows:

BPSK, AWGN channel

$$P(r) = P\big((r \cap s = +1) \cup (r \cap s = -1)\big).$$

As the events "$r \cap s = +1$" and "$r \cap s = -1$" are mutually exclusive, we can write

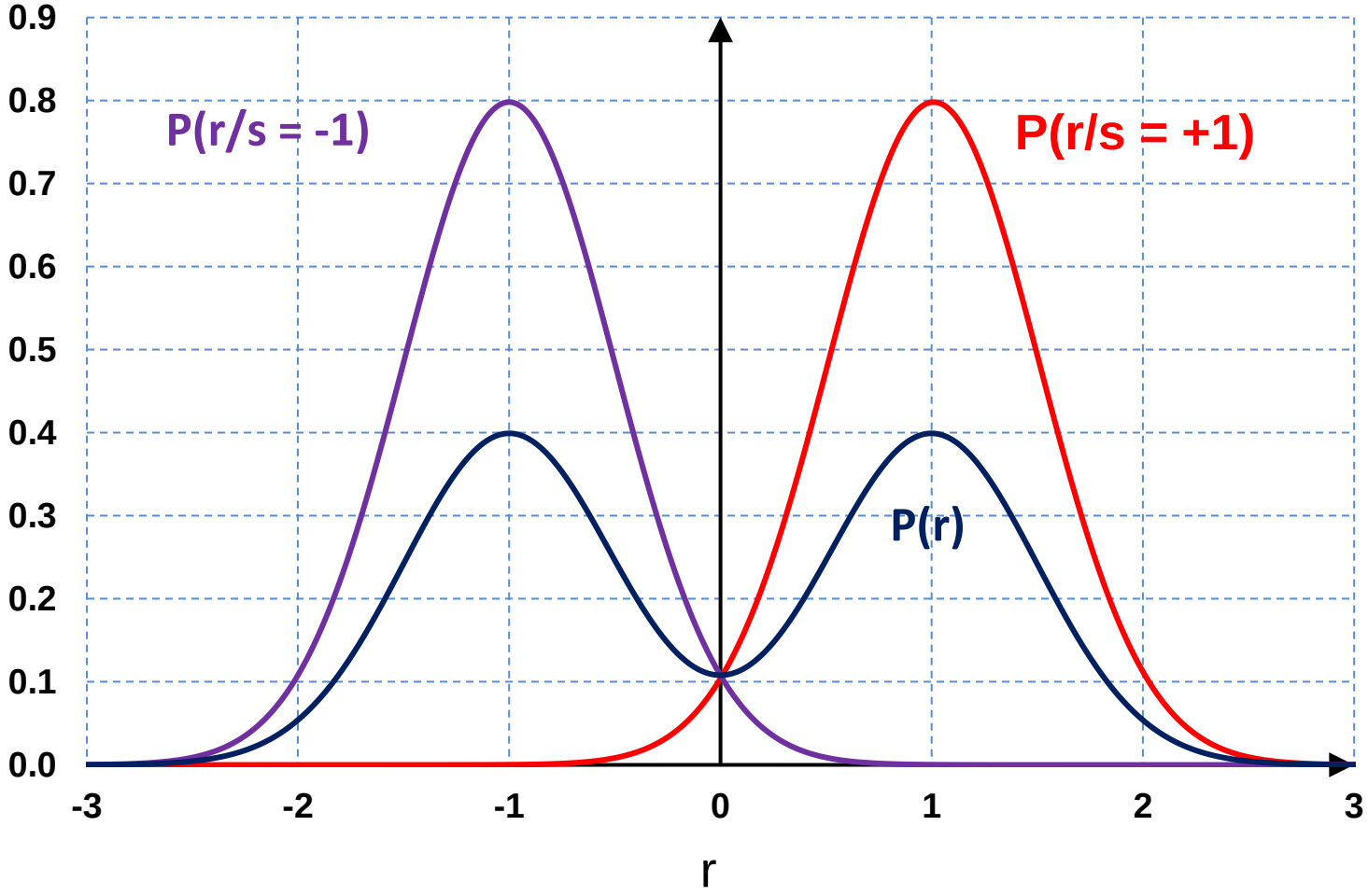
$$P(r) = P(r \cap s = +1) + P(r \cap s = +1).$$

Using Bayes' rule, we then obtain $P(r) = P(r|s = +1)\cdot\Pr(s = +1) + P(r|s = -1)\cdot\Pr(s = -1)$

$$= \frac{1}{2\sqrt{2\pi\sigma^2}} exp\left(-\frac{(r-1)^2}{2\sigma^2}\right) + \frac{1}{2\sqrt{2\pi\sigma^2}} exp\left(-\frac{(r+1)^2}{2\sigma^2}\right).$$

BPSK, AWGN channel

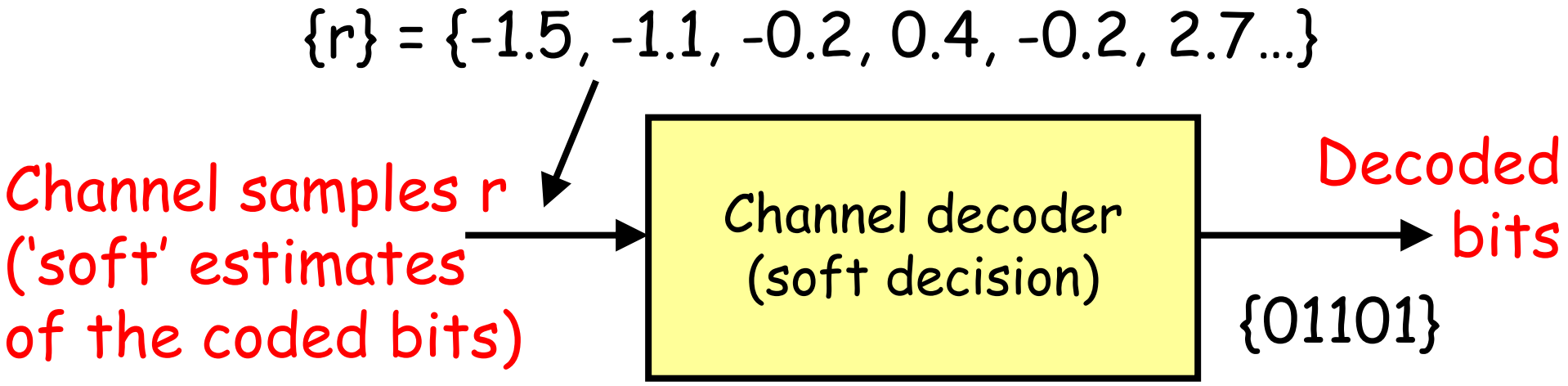
Probability density function of a channel
sample r = s + n (σ^2 = 1/4)



BPSK, AWGN channel

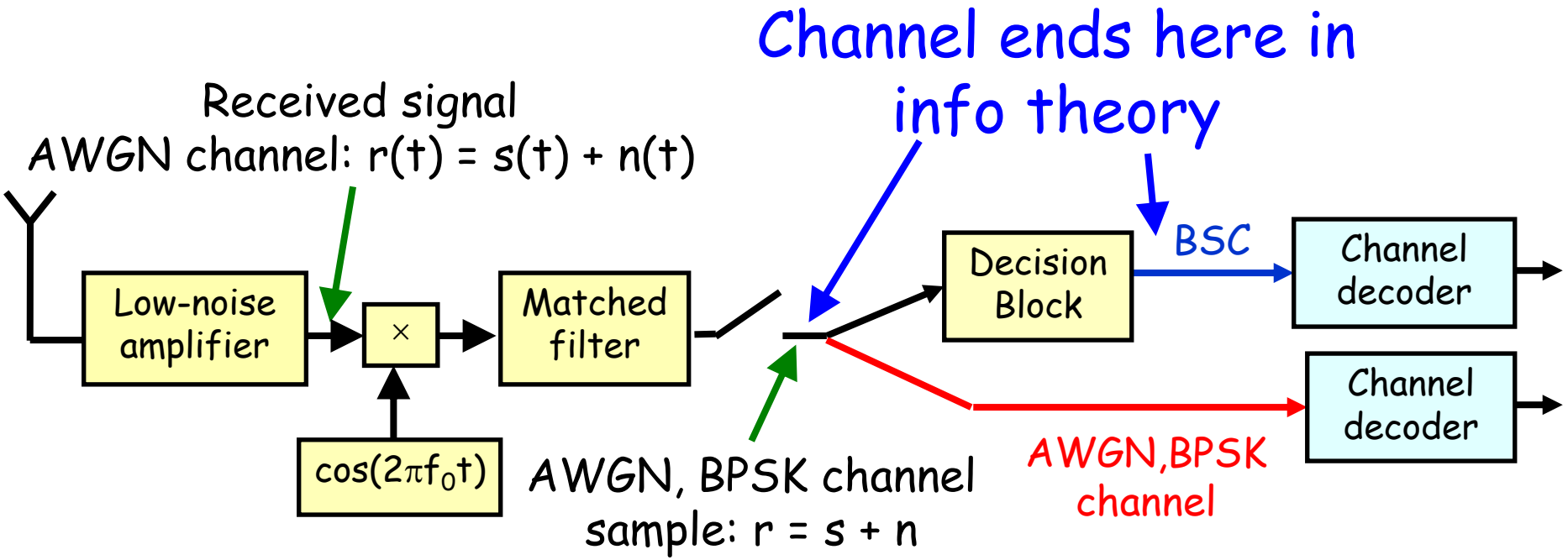
The channel output is a sequence of samples r = s + n that are analogue (soft) estimates of the transmitted bits s = +1 or -1.

In a coded system, this sequence is processed by the channel decoder.



What type of channel: you decide!

The use of a decision block inside the demodulator is often the only difference between a BSC and a BPSK, AWGN channel.



What type of channel: you decide!

The decision block converts a BPSK, AWGN channel into a BSC.

To do so, the decision block proceeds as follows:
If $r > 0$, it converts the channel sample r to +1, which corresponds, say, to a '1';
If $r < 0$, then it convert r to -1, which corresponds, say, to a '0'.

After conversion, the decoder is able to operate using hard decisions rather than soft decisions, which can somewhat simplifies the decoding algorithm.

What type of channel: you decide!

But, the use of a decision block will result in a significant error performance degradation after decoding (because much info has been discarded rather than fed to the decoder).

The decoder should always be fed with soft decisions rather than hard decisions, unless there are good reasons not to do so (complexity issues, soft decisions not available in some applications…)

It is not possible to convert a BSC into a BPSK, AWGN channel: it is easy to destroy info, impossible to get it back.

Channel capacity

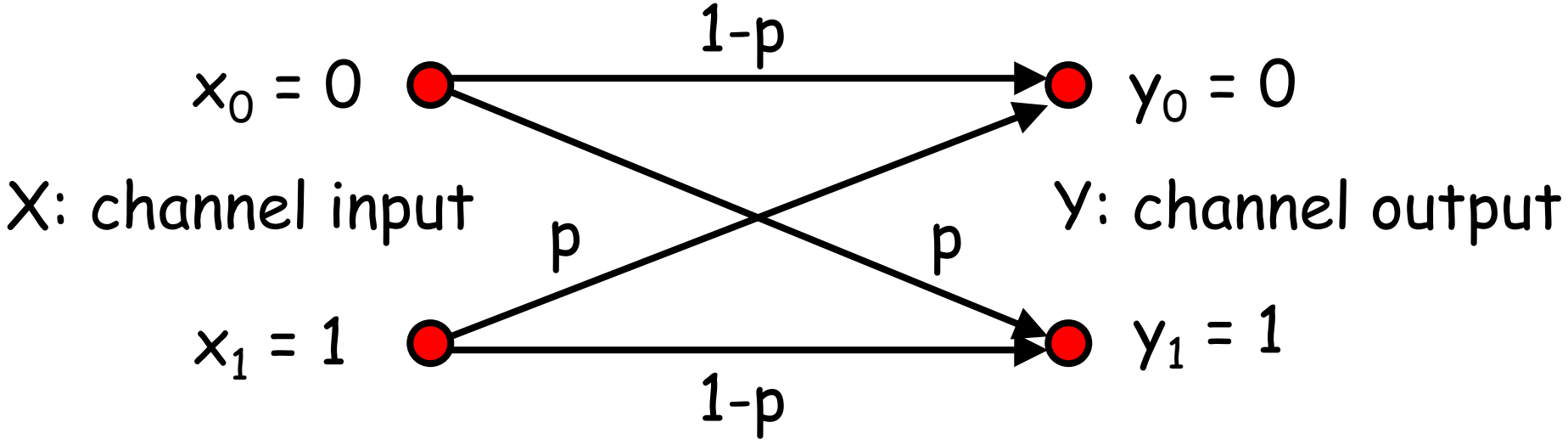
The capacity C of a channel (to transmit information) is measured by the mutual information I(X;Y).

I(X;Y) indicates how much the channel output Y can tell us about the channel input X.

→ The quality of a channel is measured by I(X;Y).

BSC capacity

Consider the case of a binary symmetric channel (BSC).



Using the generic expression of the mutual info, we can write $C = I(X;Y) = E\left\{\log_2\left(\frac{p_{j|i}}{p_j}\right)\right\}$.

BSC capacity

$$C = I(X;Y) = E\left\{\log_2\left(\frac{p_{j|i}}{p_j}\right)\right\}.$$

Notations:

- $p_{i,j} = \Pr(x_i \cap y_j)$ → Pr{symbol $x_i \in \{0, 1\}$ at channel input and symbol $y_j \in \{0, 1\}$ at channel output}

- $p_i = \Pr(x_i)$ → Pr{symbol x_i at input}

- $p_j = \Pr(y_j)$ → Pr{symbol y_j at output}

- $p_{j|i} = \Pr(y_j|x_i)$ → Pr{symbol y_j at output given symbol x_i at input}

BSC capacity

$$C = E\left\{\log_2\left(\frac{p_{j|i}}{p_j}\right)\right\}.$$

Let us start by deriving an expression for the term $p_j = \Pr(y_j)$.

We can write $p_j = \Pr(y_j) = \Pr\Big(\big(y_j \cap x_0\big) \cup \big(y_j \cap x_1\big)\Big)$.

As the events "$y_j \cap x_0$" and "$y_j \cap x_1$" are mutually exclusive, we can write

$$p_j = \Pr\big(y_j \cap x_0\big) + \Pr\big(y_j \cap x_1\big).$$

BSC capacity

Using Bayes' rule, we then obtain $p_j = \Pr(y_j|x_0)\cdot\Pr(x_0) + \Pr(y_j|x_1)\cdot\Pr(x_1)$, which leads to $p_j = \frac{1}{2}\left(p_{j|i=0} + p_{j|i=1}\right)$.

We notice that $p_{j|i=0} + p_{j|i=1} = 1, \forall j \in \{0,1\}$.

The capacity expression thus becomes $C = E\left\{\log_2\left(\frac{2p_{j|i}}{p_{j|i=0}+p_{j|i=1}}\right)\right\} = E\{\log_2(2p_{j|i})\} + E\{\log_2(p_{j|i})\}$.

BSC capacity

We can now use the usual definition of the "expected value" to proceed further:

$$C = 1 + E\{\log_2(p_{j|i})\} = 1 + \sum_{i=0}^{1}\sum_{j=0}^{1} p_{i,j}\log_2(p_{j|i}).$$

As $p_{i,j} = p_{j|i}p_i = \frac{p_{j|i}}{2}$ according to Bayes' rule, we have

$$C = 1 + \frac{1}{2}\sum_{i=0}^{1}\sum_{j=0}^{1} p_{j|i}\log_2(p_{j|i}).$$

To obtain the final expression of the channel capacity for a BSC, we can develop the double sum as follows:

BSC capacity

$$
\begin{aligned}
&C \\
&= 1 \\
&+ \frac{1}{2}\left(p_{j=0|i=0}\log_2\left(p_{j=0|i=0}\right) + p_{j=1|i=0}\log_2\left(p_{j=1|i=0}\right)\right. \\
&\left. + p_{j=0|i=1}\log_2\left(p_{j=0|i=1}\right) + p_{j=1|i=1}\log_2\left(p_{j=1|i=1}\right)\right)
\end{aligned}
$$

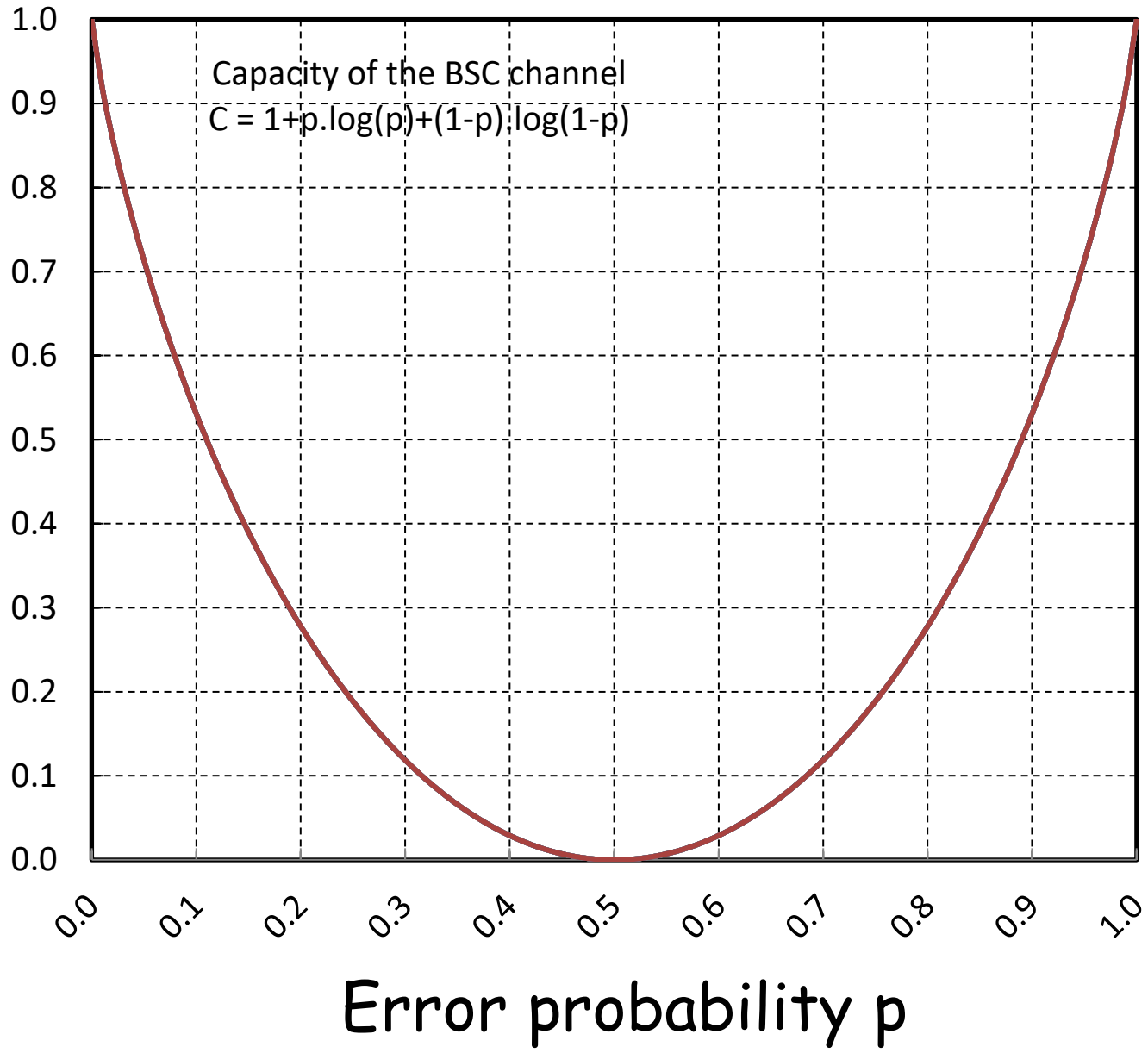
which leads to

$$
\begin{aligned}
&C \\
&= 1 \\
&+ \frac{1}{2}\left((1-p)\log_2(1-p) + p\log_2(p) + p\log_2(p)\right. \\
&\left. + (1-p)\log_2(1-p)\right)
\end{aligned}
$$

and finally $C = 1 + (1-p)\log_2(1-p) + p\log_2(p)$.

BSC capacity

Capacity C, in Shannon per channel use



BSC capacity

C is expressed in Shannon/channel use.

Useless channel: $C = 0$ Shannon/channel use, if $p = \frac{1}{2}$.

Perfect channel: $C = 1$ Shannon/channel use, if $p = 0$ or $p = 1$.

The capacity C depends on the error probability at the channel output (expected).

For $0 < p < 0.5$, we have $0 < C < 1$ Shannon/channel use.

BPSK, AWGN channel capacity

The channel output is now a continuous, rather than a discrete, source of info.

We use the same generic capacity expression

$$C = E\left\{\log_2\left(\frac{p_{j|i}}{p_j}\right)\right\}$$

but discrete probabilities must now be replaced by probability density functions (PDFs).

BPSK, AWGN channel capacity

The term $p_{j|i}$ can now be defined as Pr{sample r at channel output given a particular transmitted bit s $\in$ {-1, +1} at channel input}.

The term $p_{j|i}$ is thus given by the conditional PDF:

$$P(r|s) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(r-s)^2}{2\sigma^2}\right),$$

where s is the transmitted bit and the variance σ^2 is given by $\sigma^2 = \frac{1}{2\frac{E_s}{N_0}}$.

BPSK, AWGN channel capacity

The term p_j is now defined as Pr{sample r at channel output}.

The term p_j is thus given by the PDF $P(r) =$

$$= \frac{1}{2\sqrt{2\pi\sigma^2}} exp\left(-\frac{(r-1)^2}{2\sigma^2}\right) + \frac{1}{2\sqrt{2\pi\sigma^2}} exp\left(-\frac{(r+1)^2}{2\sigma^2}\right),$$

where the variance σ^2 is given by $\sigma^2 = \frac{1}{2\frac{E_S}{N_0}}$.

Therefore, we obtain the expression of the capacity for the BPSK, AWN channel as follows:

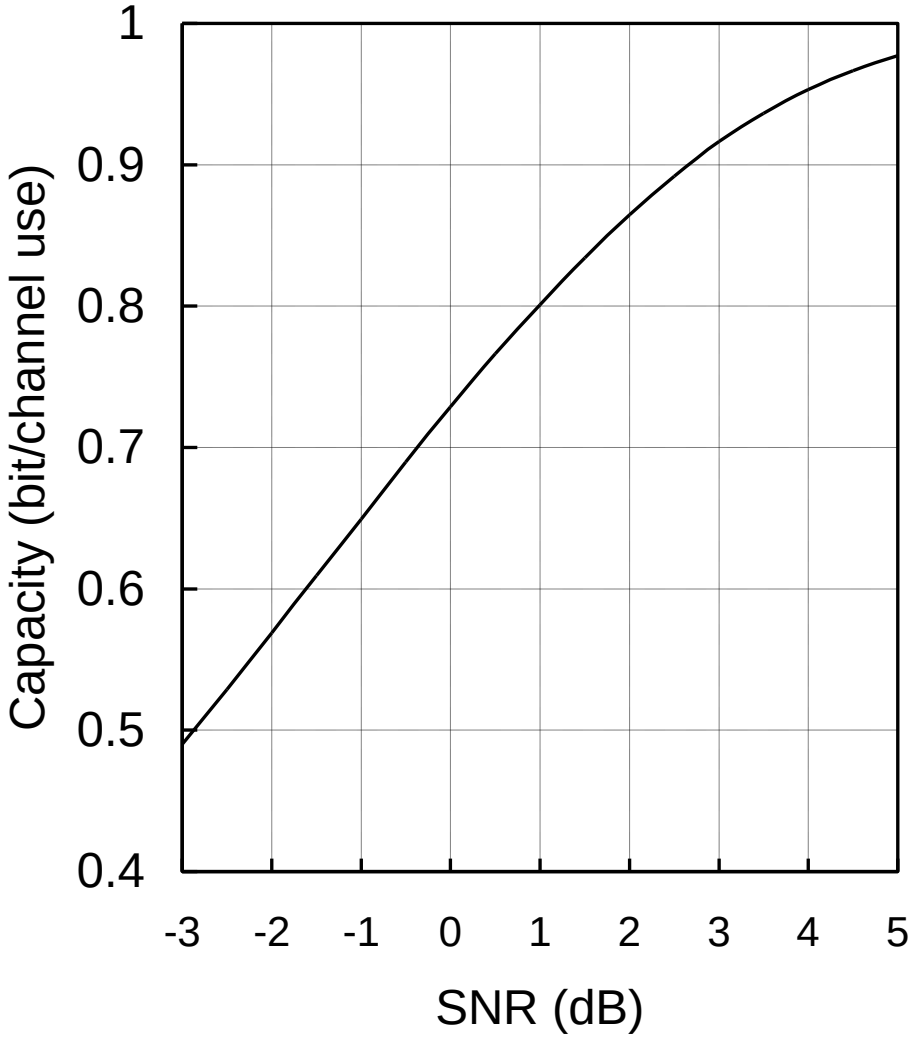
BPSK, AWGN channel capacity

$C = E\left\{\log_2\left(\frac{P(r|s)}{P(r)}\right)\right\}$, which yields $C = 1 + E\left\{\log_2\left(\frac{exp\left(-\frac{E_S}{N_0}(r-s)^2\right)}{exp\left(-\frac{E_S}{N_0}(r-1)^2\right)+exp\left(-\frac{E_S}{N_0}(r+1)^2\right)}\right)\right\}$.

There is no closed-form expression for this equation.

To evaluate C, we must perform a numerical integration using Monte Carlo simulations: generate millions of bits s, add a noise sample to each of them in order to generate the channel samples r, and find the average value of $\log_2(\ldots)$ over all (s, r) pairs.

BPSK, AWGN channel capacity



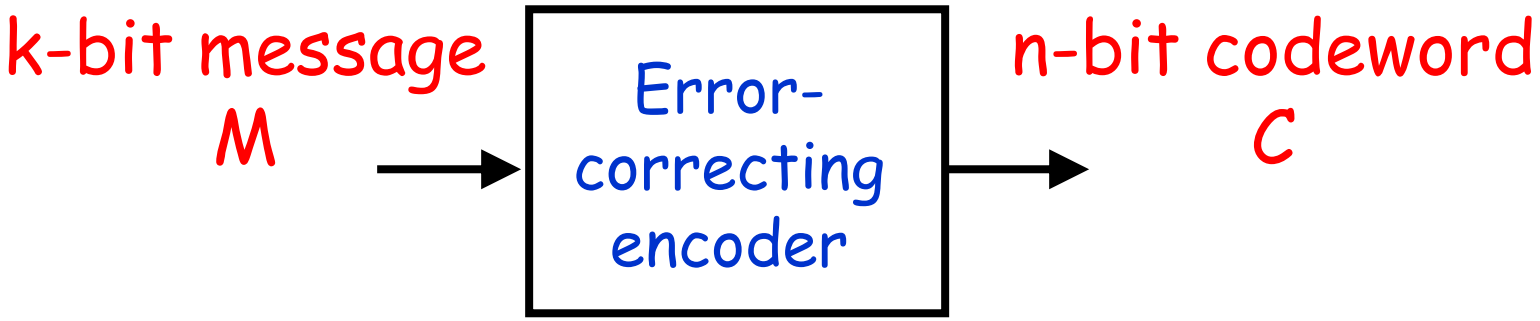
C depends on the SNR over the channel.

The higher the SNR, the higher the capacity.

On the x axis, SNR is the signal-to-noise ratio per transmitted bit s, $\frac{E_S}{N_0}$.

Shannon's capacity theorem

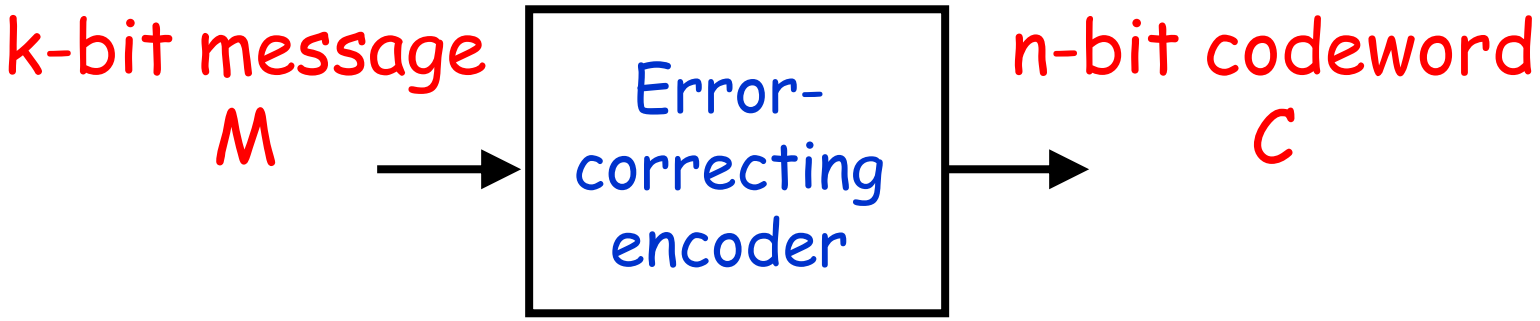
Consider the use of an error-correcting code at the transmitter side.



For every k-bit message, there is a corresponding n-bit codeword.

The ratio $\frac{k}{n}$ is the coding rate R_c of the code.

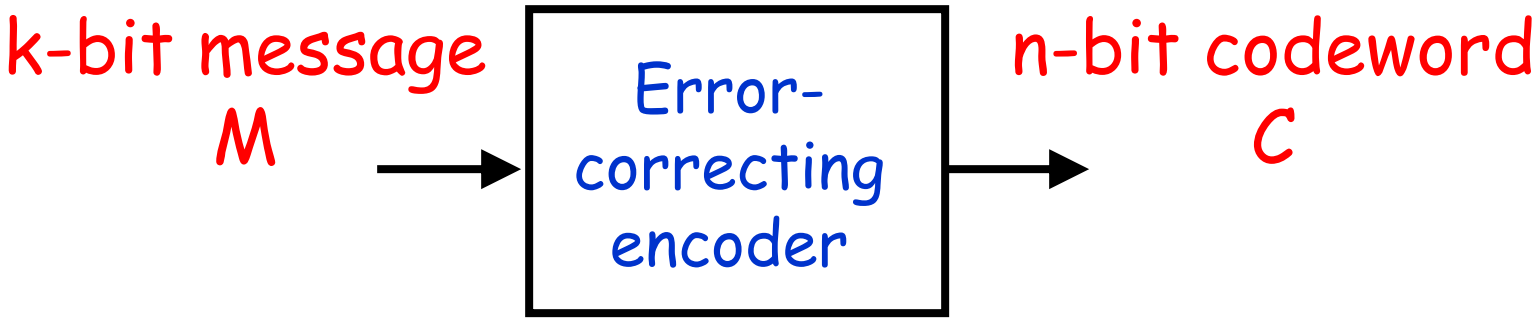
Shannon's capacity theorem



A message of k bits (referred to as 'info bits') contains k Shannons of info.

A codeword of n bits (referred to as 'coded bits') also contains k Shannons of info since the encoding process does not add any info to the message.

Shannon's capacity theorem



Each coded bit carries $\frac{k}{n}$ (< 1) Shannon of info.

Each time we use the channel to transmit a coded bit, we attempt to transmit $\frac{k}{n}$ Shannon of info.

→ The info rate R over the channel is equal to $\frac{k}{n}$, which implies that $R = R_c$.

Shannon's capacity theorem

Shannon: "The error probability at the receiver output can be reduced to zero, using an error-correcting code, if and only if the info rate R is less than or equal to the channel capacity C."

→ Capacity limit (also called Shannon limit): $R = C$

This theorem allows us to determine the upper limit on info rate while maintaining reliable communications ($P_{eb} = 0$).

It does not explicitly tell us how to reach this limit…

Application to the BSC

As long as the coding rate satisfies the condition $R_c \leq C$, with $C = 1 + (1-p)\log_2(1-p) + p\log_2(p)$, there is an error-correcting code allowing for error-free communications.

Ex:

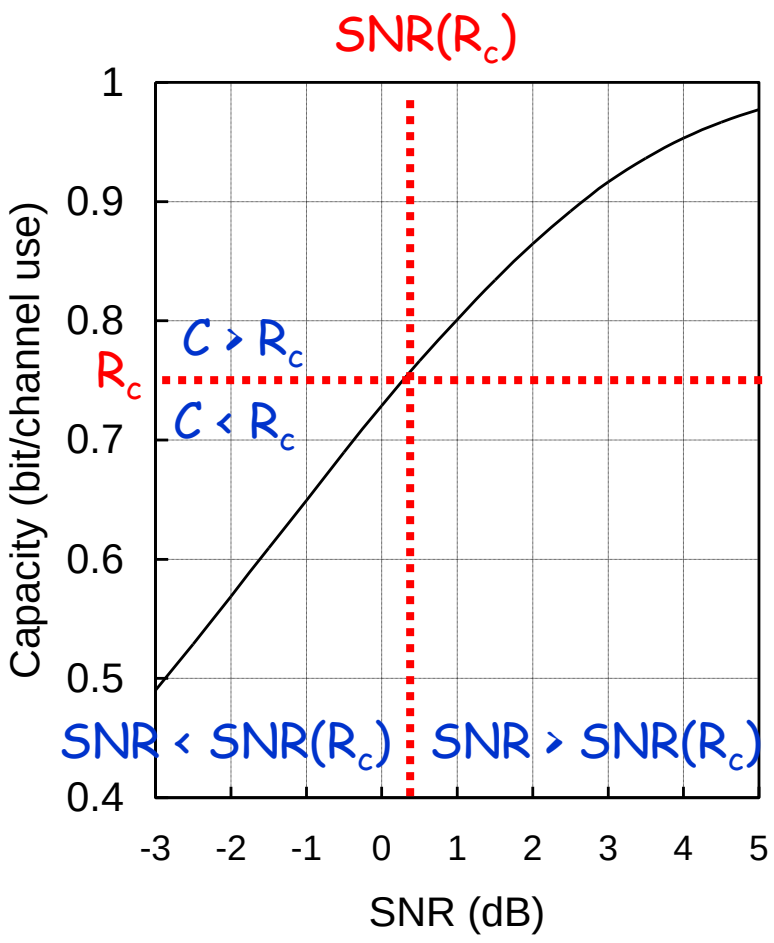
p = 10^{-1} → $R_c \leq 0.531$;

p = 10^{-2} → $R_c \leq 0.91921$;

p = 10^{-3} → $R_c \leq 0.98859$;

p = 10^{-4} → $R_c \leq 0.99853$.

Application to the BPSK, AWGN channel

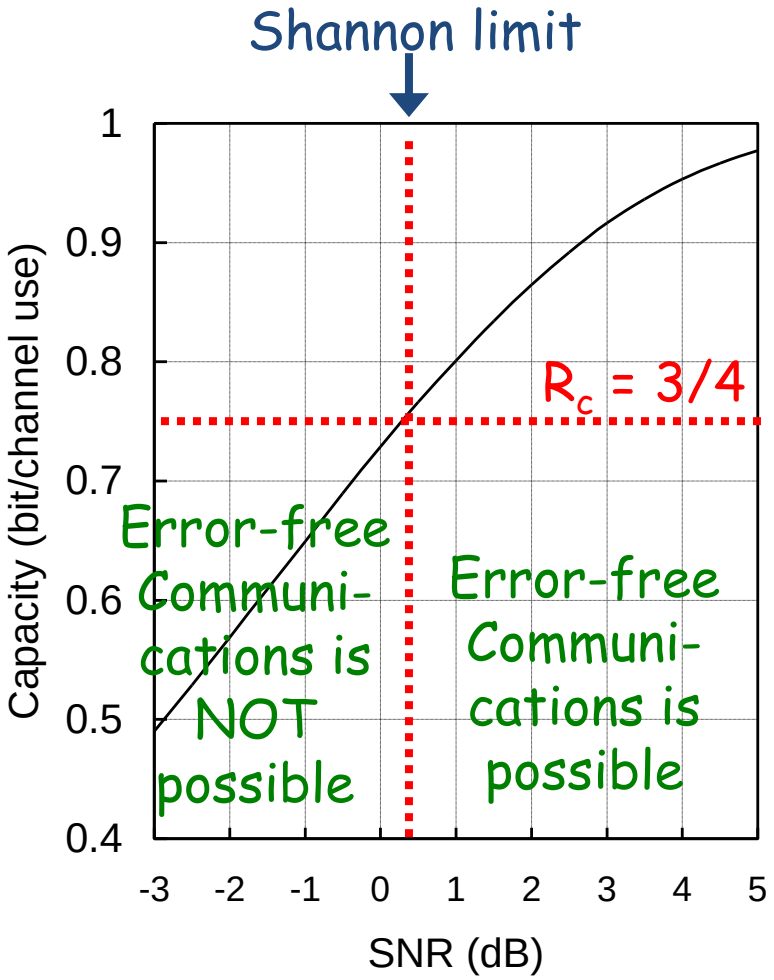


There is a rate-R_c code allowing for error-free communications, provided that $R_c \leq C$, i.e. $C \geq R_c$.

Assume the use of a code with $R_c = \frac{3}{4} = 0.75$. Error-free communications is then possible if $C \geq 0.75$ Shannon/channel use.

Since C depends on the SNR, this means that error-free communications is possible as long as SNR $\geq$ SNR($C = R_c$).

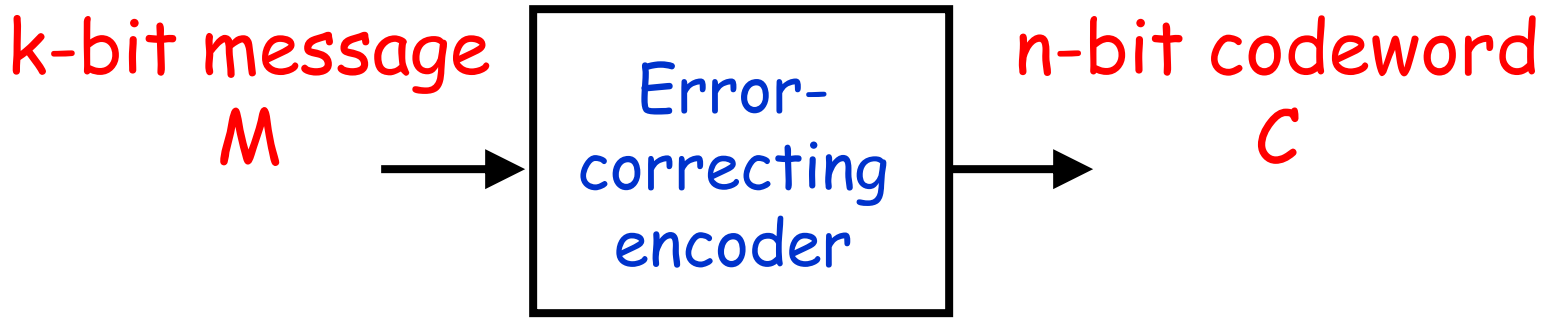
Application to the BPSK, AWGN channel



Ex: If we use a rate-$\frac{3}{4}$ code, error-free communications is possible if $\frac{E_S}{N_0} \geq 0.3845$ dB.

We must consider the SNR per info bit, $\frac{E_b}{N_0}$, instead of the SNR per coded bit, $\frac{E_S}{N_0}$.

Application to the BPSK, AWGN channel



The total energy in a message M is equal to kE_b.

The total energy in a codeword C is equal to nE_s.

We must have $kE_b = nE_s$ as the encoding process does not add energy. We thus have $E_s = \frac{k}{n}E_b$, which yields $\frac{E_s}{N_0} = \frac{k}{n}\frac{E_b}{N_0} = R_c\frac{E_b}{N_0}$.

Application to the BPSK, AWGN channel

In practice, we often express SNRs in decibels (dB).

We can write $10\log_{10}\left(\frac{E_s}{N_0}\right) = 10\log_{10}\left(\frac{E_b}{N_0}\right) + 10\log_{10}(R_c)$.

The link between both SNRs when they are expressed in dB is as follows: $\frac{E_s}{N_0} = \frac{E_b}{N_0} + 10\log_{10}(R_c)$

or $\frac{E_b}{N_0} = \frac{E_s}{N_0} - 10\log_{10}(R_c) = \frac{E_s}{N_0} + 10\log_{10}\left(\frac{1}{R_c}\right)$.

Note that $\frac{E_b}{N_0} > \frac{E_s}{N_0}$, as $R_c < 1$.

Application to the BPSK, AWGN channel

Ex: If $R_c = \frac{3}{4}$, then $\frac{E_b}{N_0} \approx 0.3845$ dB – $10\log_{10}(\frac{3}{4}) \approx$ 1.634 dB.

→ There is a rate-3/4 code that allows for error-free communications as long as the SNR per info bit is greater than or equal to ~1.634 dB.

→ Shannon limit ≈ 1.634 dB.

How to interpret the Shannon limit in practice?

Application to the BPSK, AWGN channel

The Shannon/capacity limit simply represents the smallest possible SNR at which a communication system can operate.

For instance, if the capacity limit is $\frac{E_b}{N_0} \approx 1.634$ dB, the system can operate reliably as long as the SNR $\frac{E_b}{N_0}$ is greater than or equal to 1.634 dB.

It also means that the same system cannot operate reliably at a SNR $\frac{E_b}{N_0}$ lower than 1.634 dB.

Application to the BPSK, AWGN channel

Assume a particular rate-3/4 code has been shown to achieve error-free communications at $\frac{E_b}{N_0}$ = 3 dB.

Note: In practice, $P_{eb} = 10^{-5}$ is often the reference value for "error-free" communications.

→ This code is said to perform within (3 - 1.634) ≈ 1.37 dB of the capacity limit.

→ We can easily assess the error-correction capability of any code.

Application to the BPSK, AWGN channel

There is yet another way to exploit the knowledge of the Shannon limit.

If we use a BPSK, AWGN channel with $\frac{E_b}{N_0}$ = 1.634 dB, we know that error-free transmission is only achievable by using a code with $R_c \leq \frac{3}{4}$.

This puts a upper limit on the bandwidth/bit rate of the system. If we go over this limit, then error-free communications is definitely not possible, no matter what.

Application to the BPSK, AWGN channel

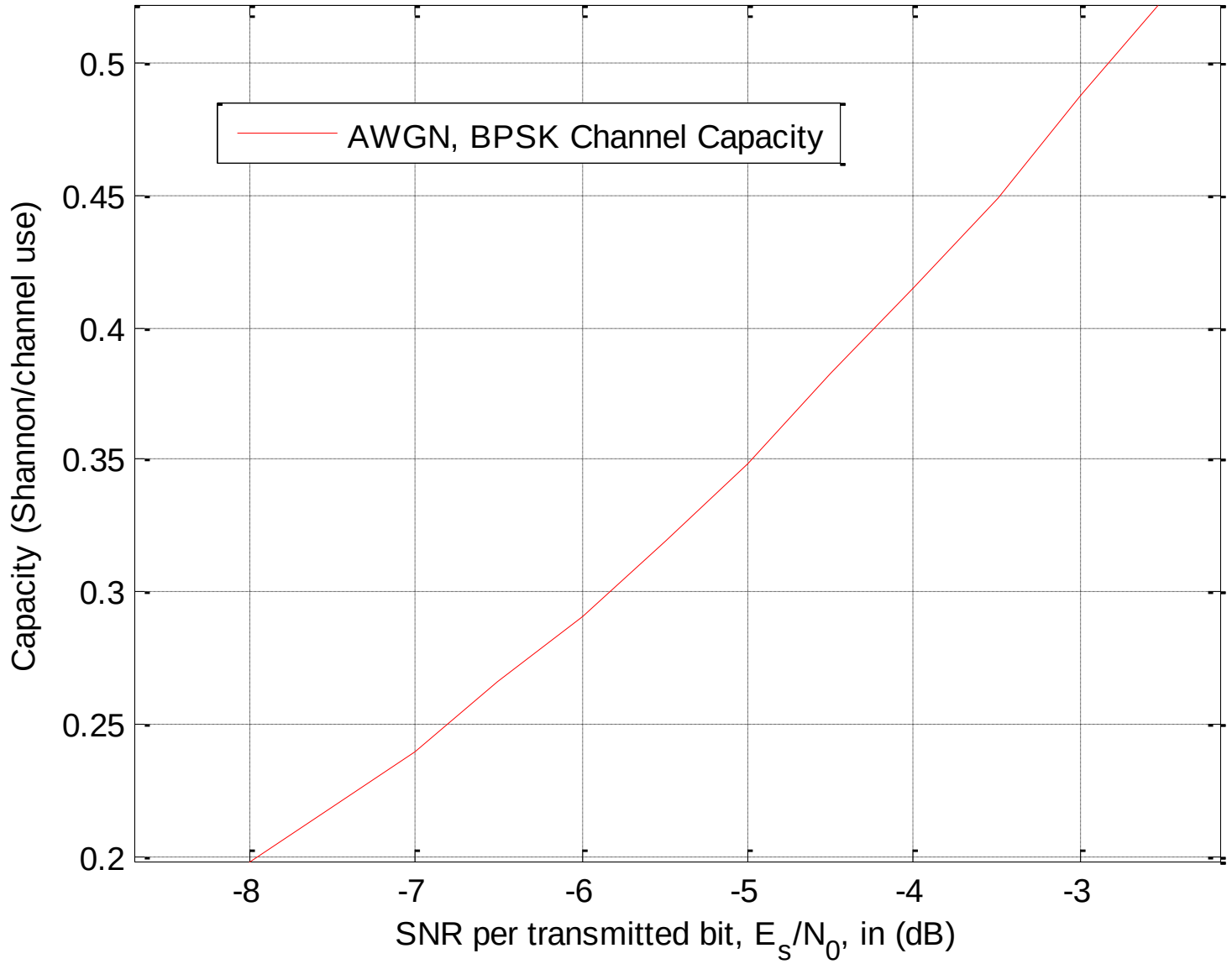
We give below a few more capacity limit values so that you can practice the technique to determine the capacity limit value for a given coding rate R_c.

For $R_c = \frac{1}{4}$, we find $\frac{E_b}{N_0} \approx$ -6.80 dB – $10\log_{10}(\frac{1}{4}) \approx$ -0.78 dB.

For $R_c = \frac{1}{3}$, we find $\frac{E_b}{N_0} \approx$ -5.25 dB – $10\log_{10}(\frac{1}{3}) \approx$ -0.48 dB.

For $R_c = \frac{1}{2}$, we find $\frac{E_b}{N_0} \approx$ -2.835 dB – $10\log_{10}(\frac{1}{2}) \approx$ 0.175 dB.

Application to the BPSK, AWGN channel



Application to the BPSK, AWGN channel

What code should we use to reach the Shannon limit?

(1) Message and codeword of infinite length ($\mathrm{k}, n \to +\infty$). Not very realistic…

(2) For any message, the corresponding codeword is chosen randomly and independently of the other codewords → Concept of random coding.

These two rules were assumptions used in the mathematical demonstration of the Shannon's capacity theorem in 1948.

The challenge after 1948

The clues offered by Shannon are actually not very useful since the implementation of long random codes seems far too complex.

Think of the encoding and decoding complexity of such codes at a time where there were no integrated circuits or computers around.

At the encoder side, how to implement a random encoder?

Use a look-up table with a capacity of $n \times 2^k$ bits?

The challenge after 1948

At the decoder side, how to search for the most likely codeword among the 2^k possible codewords?

As a result, researchers have traditionally focused on the search for codes that possess a lot of "structure", i.e. for which the encoding and decoding complexities remain reasonable.

For a given coding rate R_c, the goal has been to maximize a parameter called the minimum Hamming distance, d_{min}, between codewords.

The challenge after 1948

The problem with such approach is that it leads to codes that do not necessarily perform close to the Shannon limit.

For 45 years, finding a "practical" code that could perform close to the Shannon limit was considered an utopia.

Then came the IEEE International Conference on Communications, Geneva, May 1993…