CT216 Introduction to Communication Systems
Lecture 3: Channel Coding

Yash M. Vasavada

Professor, DA-IICT, Gandhinagar

09/02/2024

Overview of Today's Talk

Overview of Today's Talk

Overview of Today's Talk

1. Block Diagrams
2. Channel Coding
3. Practical Channel Coding
   - Hamming Distance
   - Several FEC Schemes

Overview of Today's Talk

Overview of Today's Talk

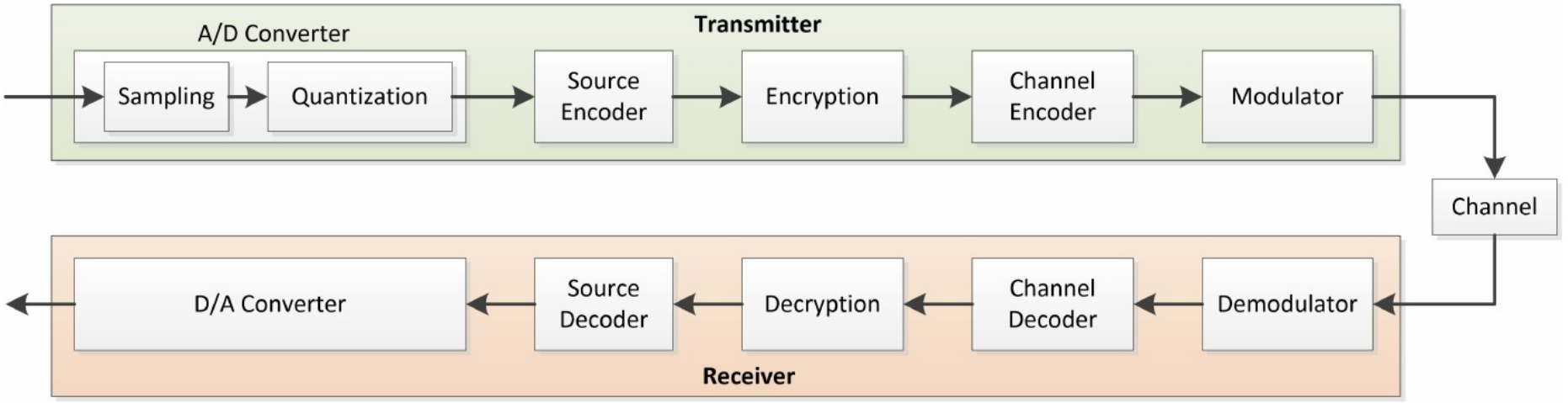
Overview of Today's Talk

Overview of Today's Talk

Digital Communication Transceiver

Block Diagram

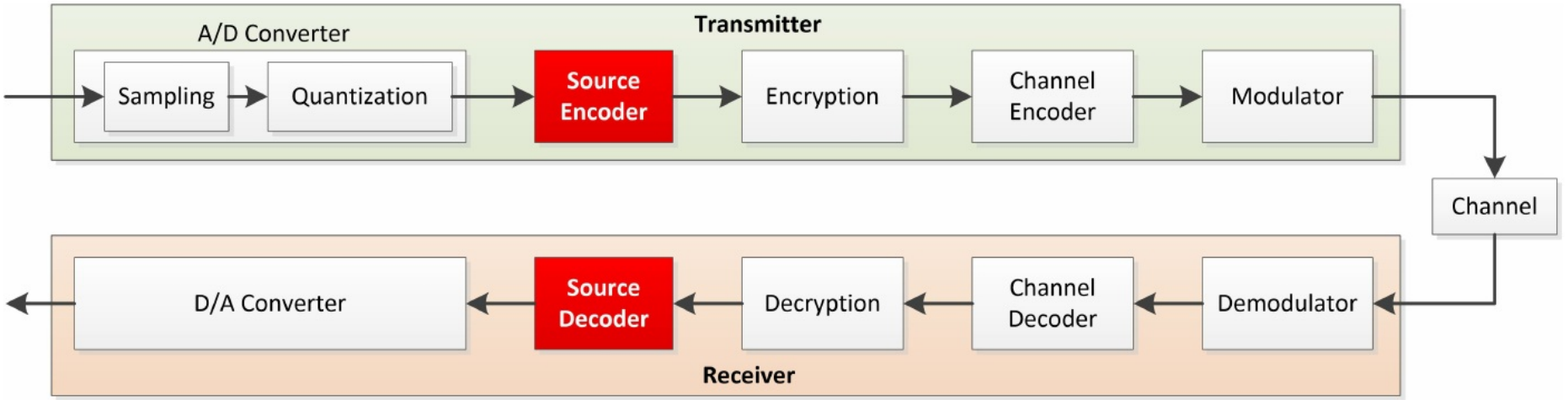
- A conceptual block diagram model of a digital communication transceiver



Digital Communication Transceiver

Block Diagram

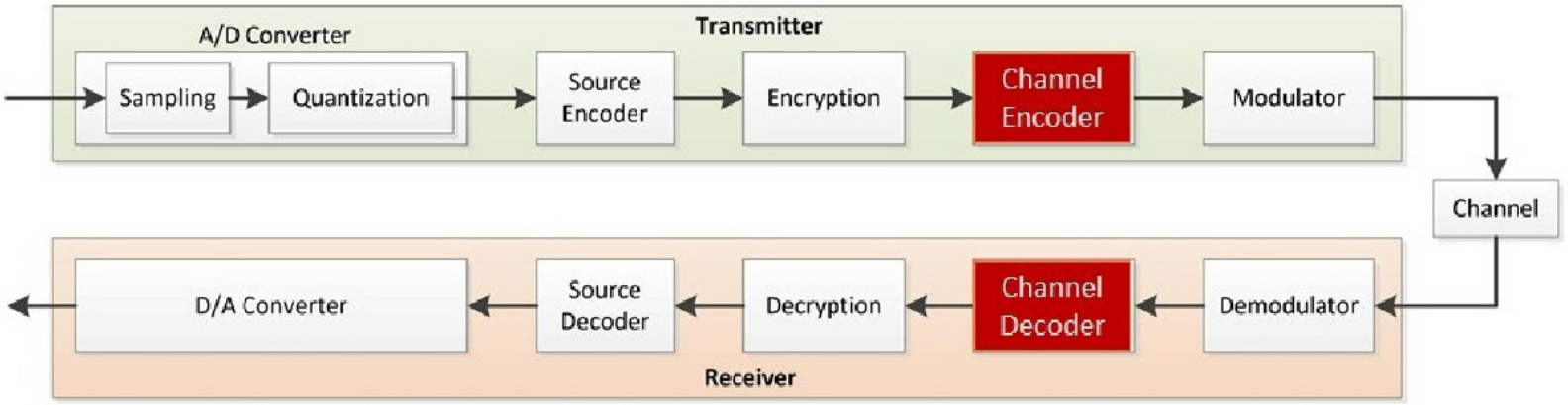
- We will be studying the mathematics and algorithms of source encoding



Digital Communication Transceiver

Channel Coding

- For now, we focus on channel encoding/decoding, also known as FEC (forward error correction) coding



Channel Coding

also known as Forward Error Correction (FEC) Coding

- Purpose: to turn a noisy channel, which can introduce errors in the transmitted symbols, to (ideally) a noiseless channel
  - The motivation for this is clear — if the channel coding works in this ideal manner, you will not hear disturbances when you are talking on the phone

Channel Coding

also known as Forward Error Correction (FEC) Coding

- Purpose: to turn a noisy channel, which can introduce errors in the transmitted symbols, to (ideally) a noiseless channel
  - The motivation for this is clear — if the channel coding works in this ideal manner, you will not hear disturbances when you are talking on the phone
  - One of the 5G service targets is remote surgery; when the surgeon is operating an instrument remotely, using a communication link, that link better be ultra-reliable, i.e., it should be as noiseless as possible.

Channel Coding

also known as Forward Error Correction (FEC) Coding

- Purpose: to turn a noisy channel, which can introduce errors in the transmitted symbols, to (ideally) a noiseless channel
  - The motivation for this is clear — if the channel coding works in this ideal manner, you will not hear disturbances when you are talking on the phone
  - One of the 5G service targets is remote surgery; when the surgeon is operating an instrument remotely, using a communication link, that link better be ultra-reliable, i.e., it should be as noiseless as possible.
- The real-world channels are always noisy, and the task of the channel coding is to turn the real-world channel into a channel as close to ideal as possible

Channel Coding

also known as Forward Error Correction (FEC) Coding

- Purpose: to turn a noisy channel, which can introduce errors in the transmitted symbols, to (ideally) a noiseless channel
  - The motivation for this is clear — if the channel coding works in this ideal manner, you will not hear disturbances when you are talking on the phone
  - One of the 5G service targets is remote surgery; when the surgeon is operating an instrument remotely, using a communication link, that link better be ultra-reliable, i.e., it should be as noiseless as possible.
- The real-world channels are always noisy, and the task of the channel coding is to turn the real-world channel into a channel as close to ideal as possible
  - The channel's "noisiness" is measured always relatively, i.e., as the received signal power P_s to the received noise power ratio, i.e., the SNR P_S/P_n

Channel Coding

also known as Forward Error Correction (FEC) Coding

- Purpose: to turn a noisy channel, which can introduce errors in the transmitted symbols, to (ideally) a noiseless channel
  - The motivation for this is clear — if the channel coding works in this ideal manner, you will not hear disturbances when you are talking on the phone
  - One of the 5G service targets is remote surgery; when the surgeon is operating an instrument remotely, using a communication link, that link better be ultra-reliable, i.e., it should be as noiseless as possible.
- The real-world channels are always noisy, and the task of the channel coding is to turn the real-world channel into a channel as close to ideal as possible
  - The channel's "noisiness" is measured always relatively, i.e., as the received signal power P_s to the received noise power ratio, i.e., the SNR P_S/P_n
  - The channel coding turns the actual physical communication channel with a bad SNR into an "encoded channel" whose performance is equivalent to an actual channel with a good SNR

Channel Coding

also known as Forward Error Correction (FEC) Coding

- At first, it appears that the task is not possible — the channel already has introduced the noise, how can its effect be undone?

Channel Coding

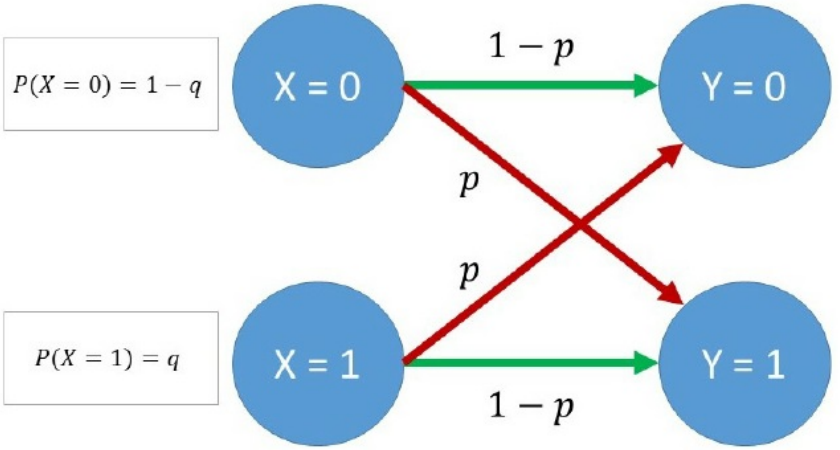
also known as Forward Error Correction (FEC) Coding

- At first, it appears that the task is not possible — the channel already has introduced the noise, how can its effect be undone?
- To answer this, we go back to the cause $\rightarrow$ effect model of the Bayesian belief model
- The fundamental reason why the receiver makes error in the presence of a noisy channel is that when it observes a particular channel output, $Y = 0$ or $Y = 1$, it cannot decide with certainty whether the cause is $X = 0$ or $X = 1$

Channel Coding

also known as Forward Error Correction (FEC) Coding

- At first, it appears that the task is not possible — the channel already has introduced the noise, how can its effect be undone?
- To answer this, we go back to the cause $\rightarrow$ effect model of the Bayesian belief model
- The fundamental reason why the receiver makes error in the presence of a noisy channel is that when it observes a particular channel output, $Y = 0$ or $Y = 1$, it cannot decide with certainty whether the cause is $X = 0$ or $X = 1$
  - the odds in favor of $X = 1$ are typically neither ∞ nor 0



Channel Coding

The Main Idea

- The art of channel coding is to turn the Bayesian diagram into one that looks like this:

Channel Coding

The Main Idea

- The art of channel coding is to turn the Bayesian diagram into one that looks like this:
- The cardinality of the transmitted message set is maintained the same (two, in this case), but the cardinality of the received message set is, somehow, expanded (to four, in this case)

Channel Coding

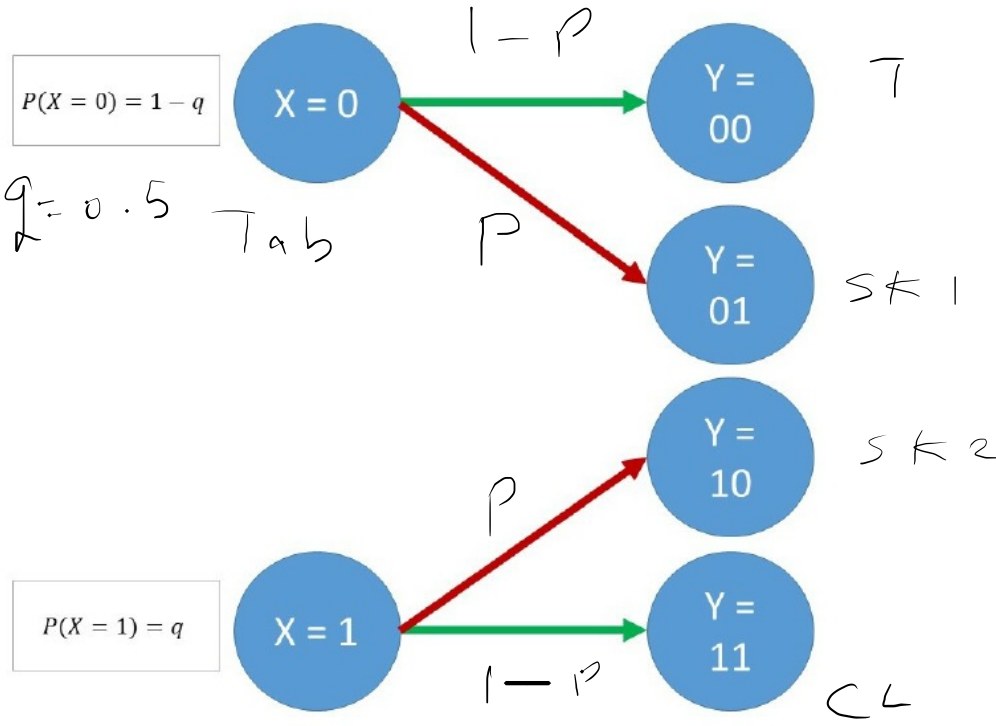
The Main Idea

- The art of channel coding is to turn the Bayesian diagram into one that looks like this:
- The cardinality of the transmitted message set is maintained the same (two, in this case), but the cardinality of the received message set is, somehow, expanded (to four, in this case)
- Furthermore, it is ensured that the expanded set is divided into two non-overlapping subsets, one each for each of the two transmitted symbols

Channel Coding

The Main Idea

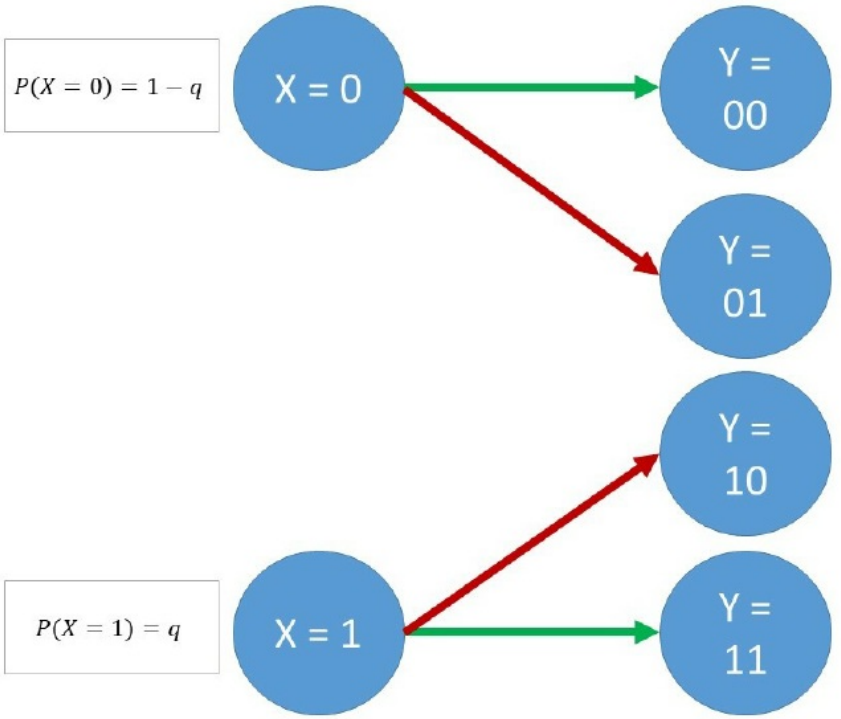
- The art of channel coding is to turn the Bayesian diagram into one that looks like this:
- The cardinality of the transmitted message set is maintained the same (two, in this case), but the cardinality of the received message set is, somehow, expanded (to four, in this case)
- Furthermore, it is ensured that the expanded set is divided into two non-overlapping subsets, one each for each of the two transmitted symbols



Channel Coding

The Main Idea

- The art of channel coding is to turn the Bayesian diagram into one that looks like this:
- The cardinality of the transmitted message set is maintained the same (two, in this case), but the cardinality of the received message set is, somehow, expanded (to four, in this case)
- Furthermore, it is ensured that the expanded set is divided into two non-overlapping subsets, one each for each of the two transmitted symbols

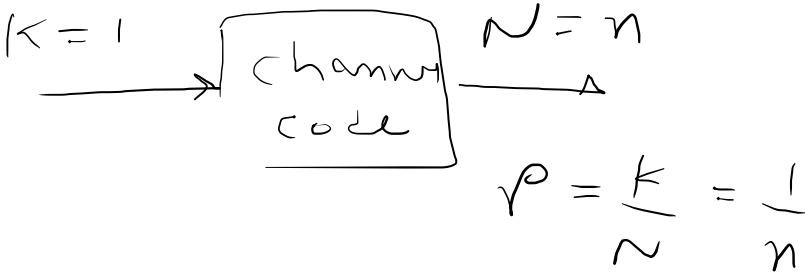


- Note that the receiver knows with certainty whether $X = 0$ or $X = 1$ for any one of the four possible received Y values

Channel Coding: A Basic Scheme

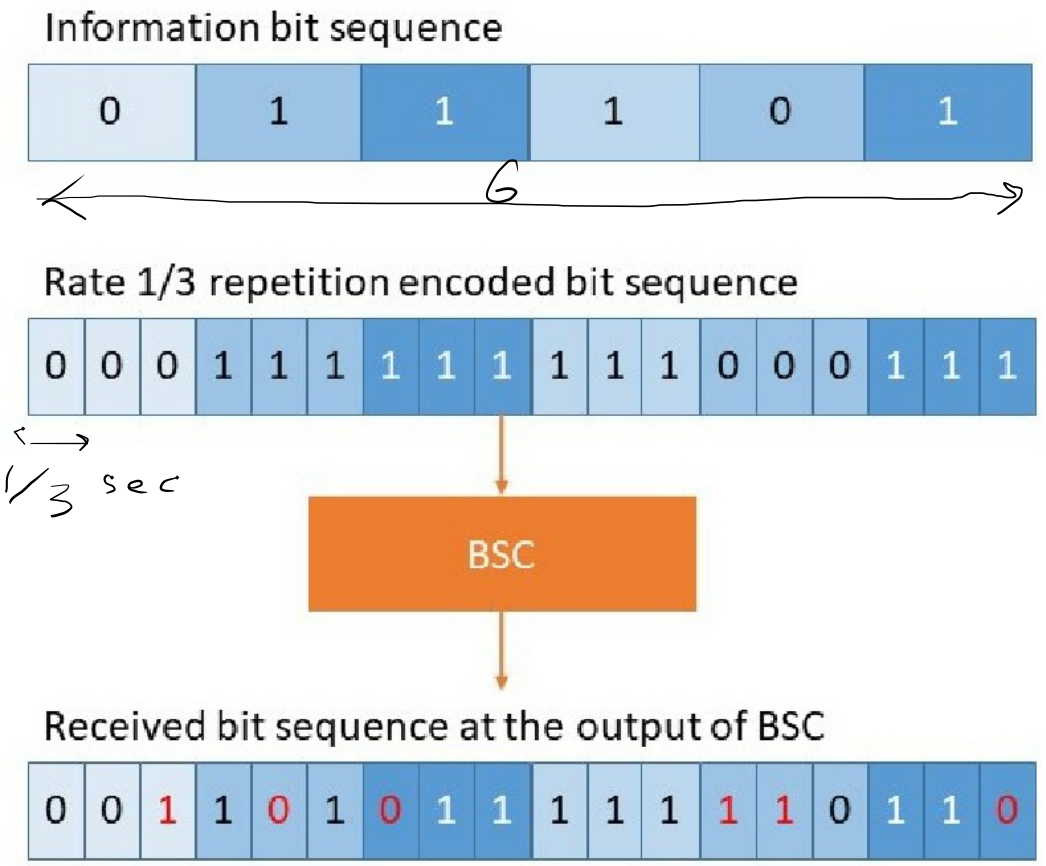
Repetition Code

- Encoding scheme:
  1. Take one bit at a time as the input.
  2. Repeat this bit $n-1$ times
  3. Transmit the resultant n bit long codeword
- Rate of this code is $r = \frac{1}{n}$.
  - $\rightarrow$ Rate r is the ratio of the number of information bits to total number of encoded bits



Channel Coding: A Basic Scheme

Repetition Code



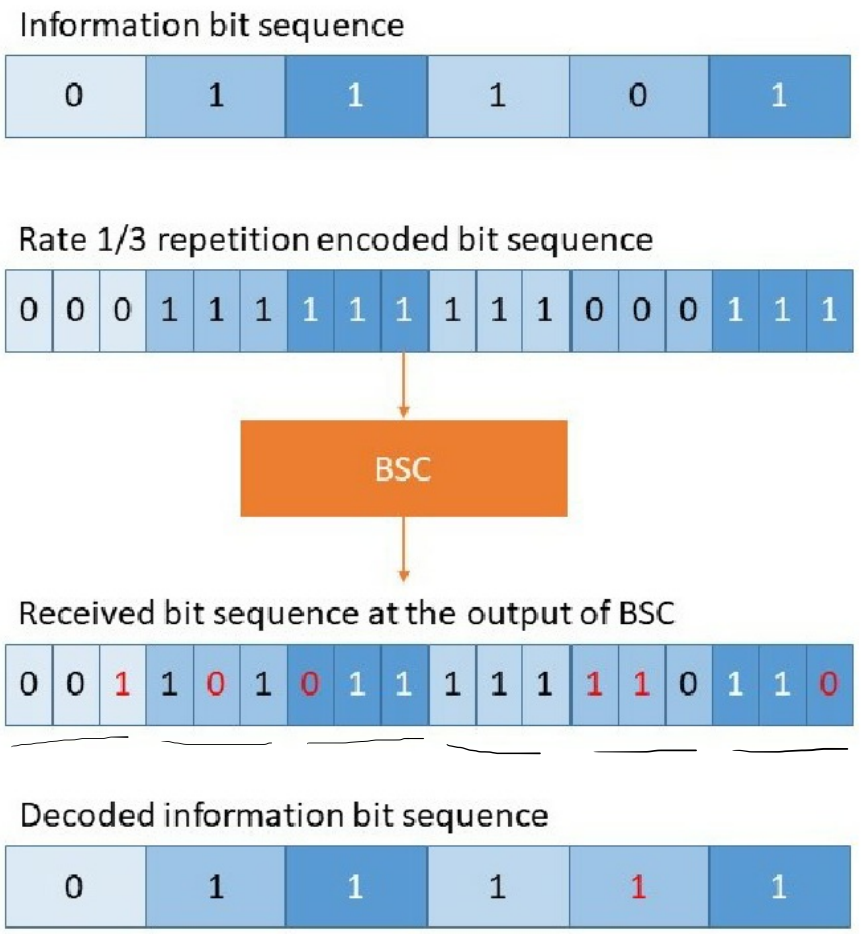
Channel Coding: A Basic Scheme

Repetition Code

What would be a good decoding strategy for this code?

Channel Coding: A Basic Scheme

Decoding of Repetition Code



Error is detected if atleast 2 bits are decoded incorrectly

$$\therefore \quad P(\text{error}) = {}^{3}C_2\, p^2(1-p) + {}^{3}C_3\, p^3(1-p)^0$$

$$= \sum_{i=\frac{N}{2}}^{N} {}^{N}C_i\, p^{(i)}(1-p)^{N-i}$$

Manan

2022013 10

Channel Coding: A Basic Scheme

Decoding of Repetition Code

- The decoding algorithm is called the majority-vote decoding
  - Take $n = 3$ bit block at a time, and decode it as that bit (either 0 or 1) that occurs the majority of times
  - To avoid the confusion in decoding, it maybe preferred to make n an odd number

Channel Coding

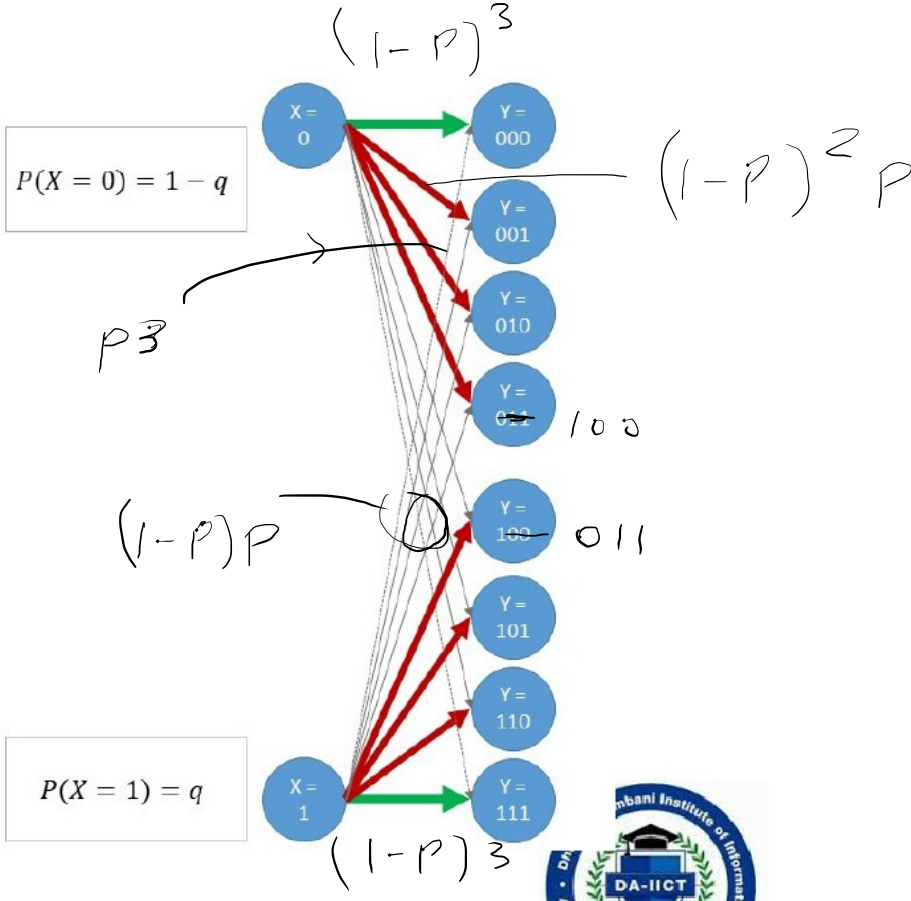
A Basic Scheme

- A view of repeat by n coding

Channel Coding

A Basic Scheme

- A view of repeat by n coding
  - Shown here for $n = 3$
  - Although the two sets of size 4 are created at the receiver, the channel does not become completely noiseless since there is a possibility that the receiver sees a message belonging to the set for $X = 1$ when $X = 0$ is transmitted



Channel Coding

Repeat-by-n Scheme

- There is a price that we pay with this repeat-by-$n = 3$ coding scheme

Channel Coding

Repeat-by-n Scheme

- There is a price that we pay with this repeat-by-$n = 3$ coding scheme
- We could have transmitted three independent bits of information; instead we are using the channel three times (to send three encoded bits) but sending actually only one bit of information

Channel Coding

Repeat-by-n Scheme

- There is a price that we pay with this repeat-by-$n = 3$ coding scheme
- We could have transmitted three independent bits of information; instead we are using the channel three times (to send three encoded bits) but sending actually only one bit of information
- Define as $r = K/N$ the rate of the channel coding scheme:

Channel Coding

Repeat-by-n Scheme

- There is a price that we pay with this repeat-by-$n = 3$ coding scheme
- We could have transmitted three independent bits of information; instead we are using the channel three times (to send three encoded bits) but sending actually only one bit of information
- Define as $r = K/N$ the rate of the channel coding scheme:
  - Here, here K is the number of information bits and $N \geq K$ is the number of encoded bits
  - $0 \leq r \leq 1$

Channel Coding

Repeat-by-n Scheme

- There is a price that we pay with this repeat-by-$n = 3$ coding scheme
- We could have transmitted three independent bits of information; instead we are using the channel three times (to send three encoded bits) but sending actually only one bit of information
- Define as $r = K/N$ the rate of the channel coding scheme:
  - Here, here K is the number of information bits and $N \geq K$ is the number of encoded bits
  - $0 \leq r \leq 1$
- For the repeat-by-n coding, $K = 1$ information bit is repeated $N = n$ times, i.e., the rate $r = 1/n$

Channel Coding

Repeat-by-n Scheme

- There is a price that we pay with this repeat-by-$n = 3$ coding scheme
- We could have transmitted three independent bits of information; instead we are using the channel three times (to send three encoded bits) but sending actually only one bit of information
- Define as $r = K/N$ the rate of the channel coding scheme:
  - Here, here K is the number of information bits and $N \geq K$ is the number of encoded bits
  - $0 \leq r \leq 1$
- For the repeat-by-n coding, $K = 1$ information bit is repeated $N = n$ times, i.e., the rate $r = 1/n$
- The two sets can be made increasingly nonoverlapping by increasing n. This reduces the probability of decoding error $p_{error} = p(e = 1)$ at the receiver but the price paid is that the rate $r \to 0$

Repetition Coding

Probability of Decoding Error

- Decoding error will occur in rate $r = 1/3$ repetition code if 2 or 3 bits are in error
- What is the probability of that occuring? The answer is given by the Binomial PMF

$$p_{error} = \binom{3}{2} p^2(1-p) + \binom{3}{1} p^3 = 3p^2 + p^3$$

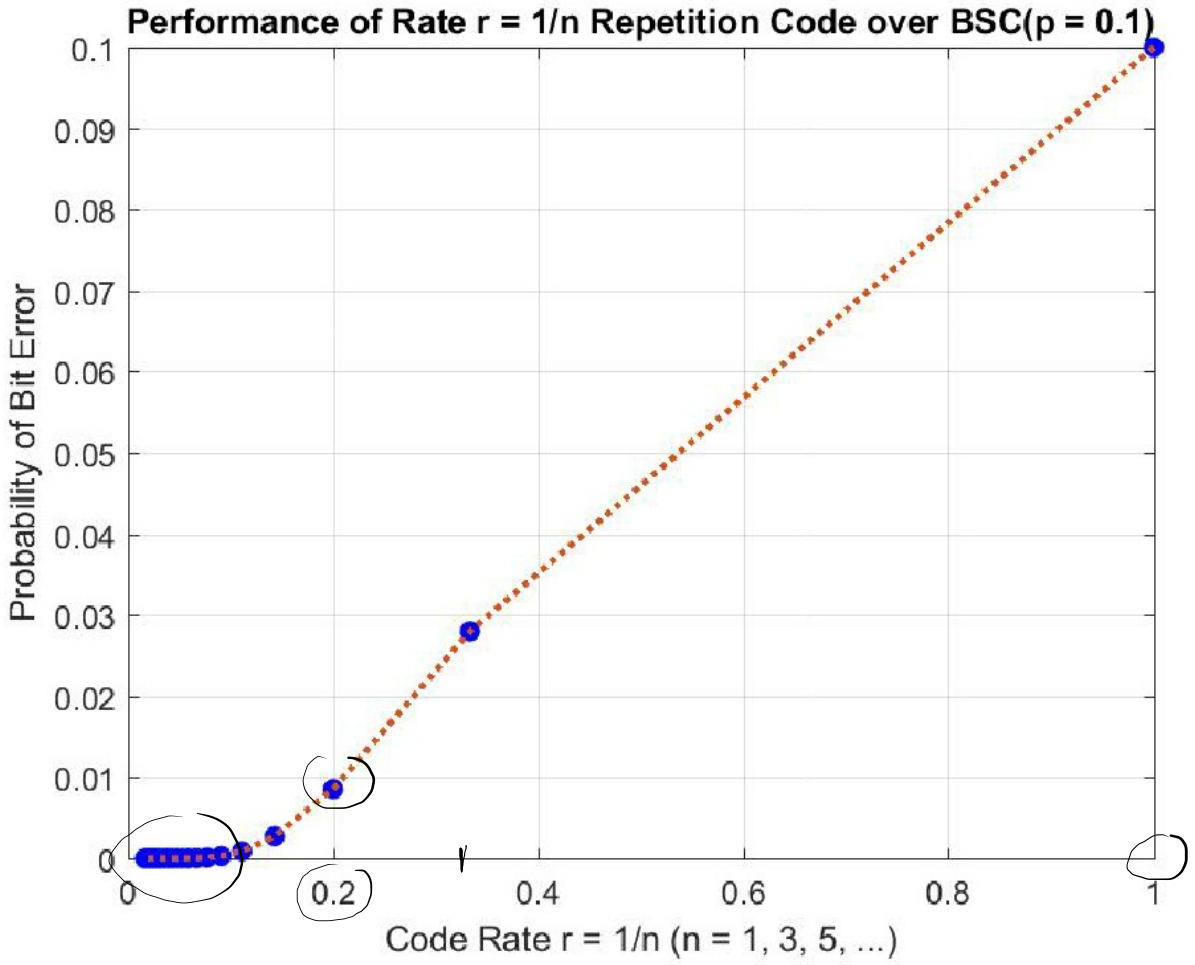
$\rightarrow$ If $p = 0.1$, $p_{error} = 3 \times 0.1^2 \times 0.9 + 0.1^3 \approx 0.03$

- Generalization to rate $1/n$ repetition code:

$$p_{error} = \sum_{k=(n+1)/2}^{n} \binom{n}{k} p^k (1-p)^{n-k}$$

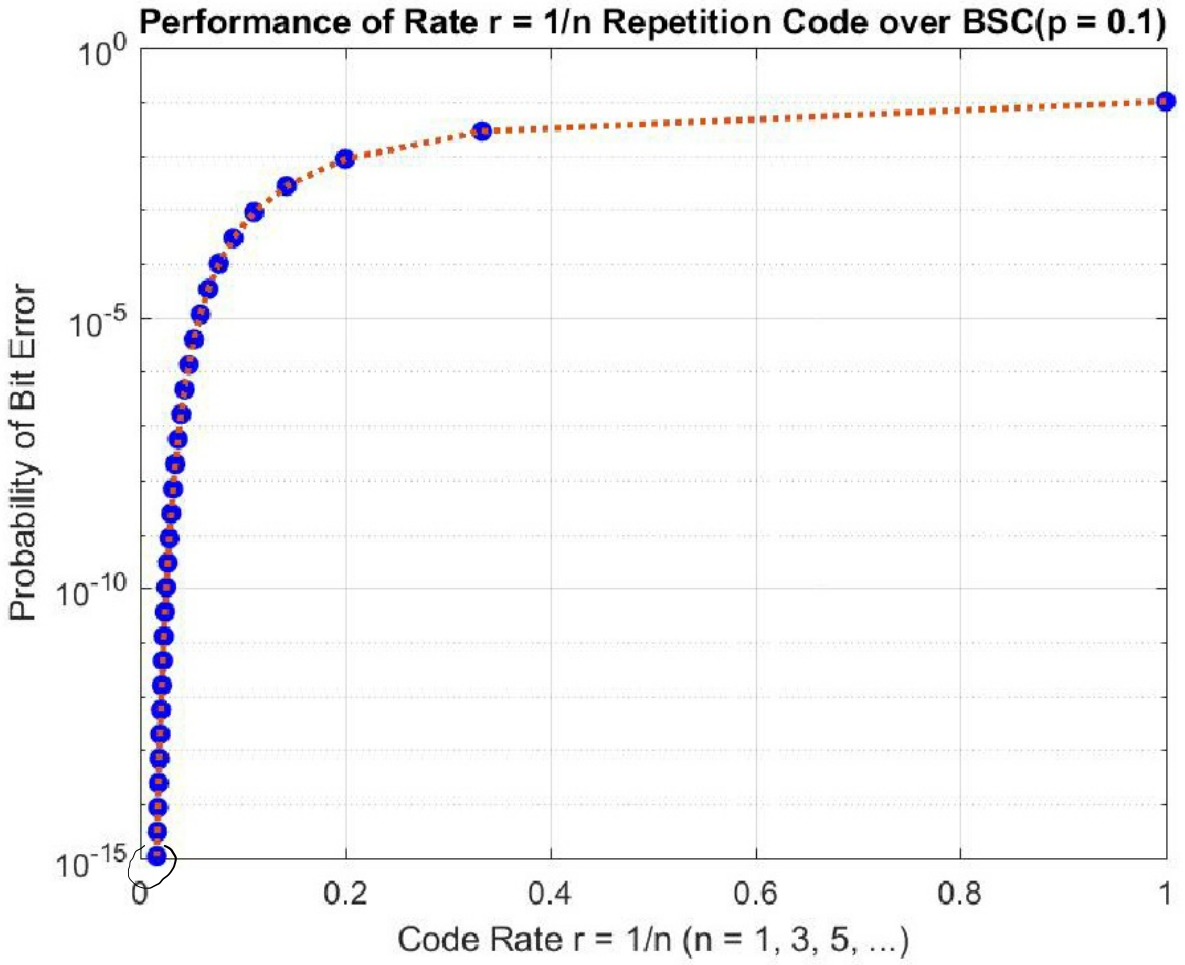
Performance of Repetition Code

over BSC(p)



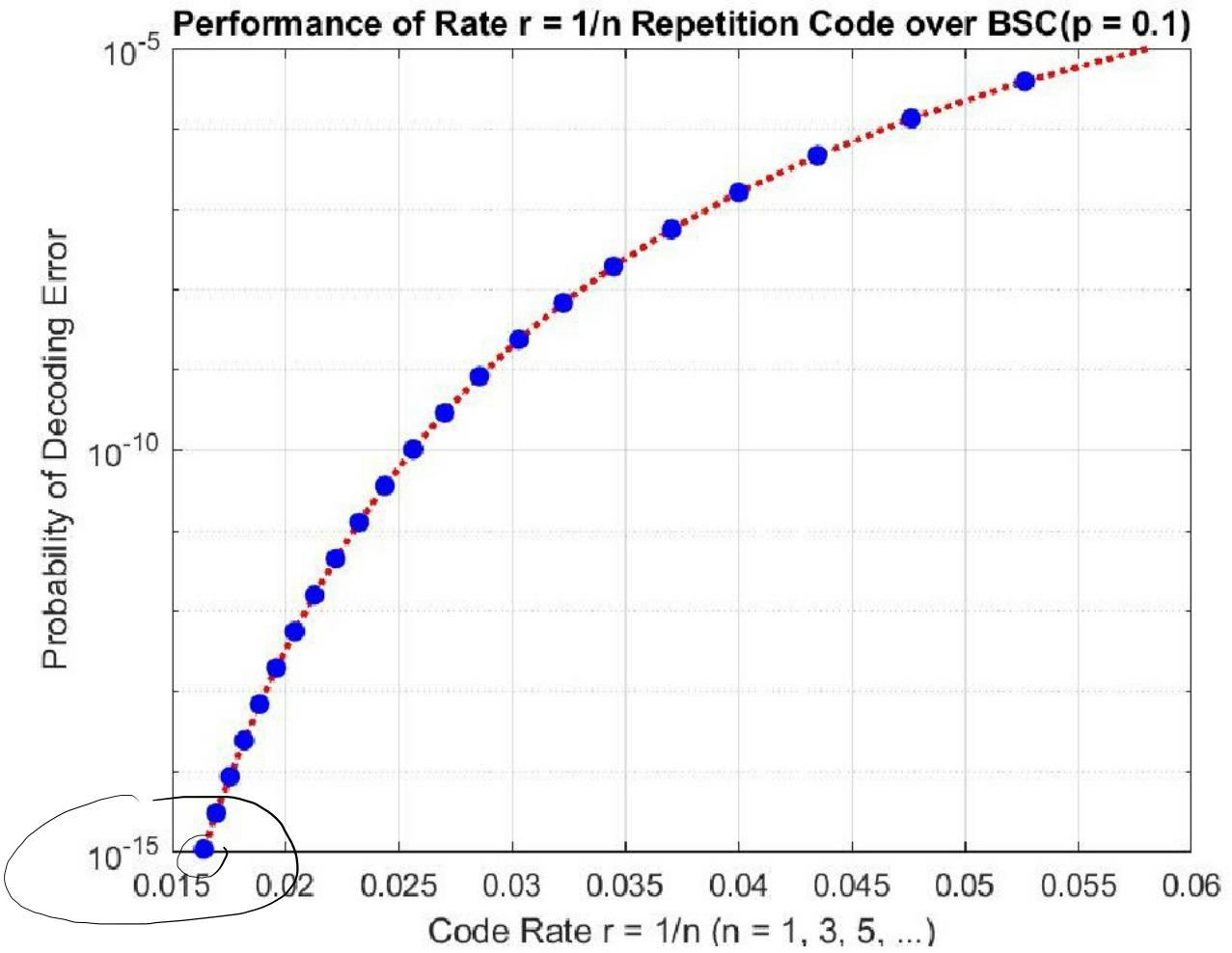
Performance of Repetition Code

over BSC(p)



Performance of Repetition Code

over BSC(p)



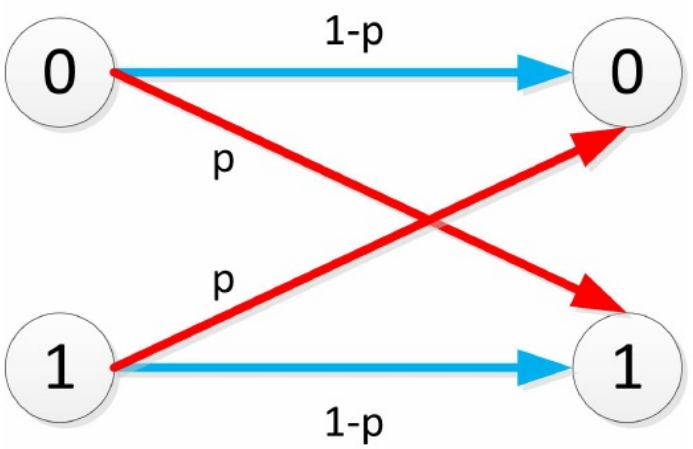
Performance of Repetition Code

over BSC(p)

- If $p = 0.1$, and we would like to design a repetition code that reduces this to $p_{error} = 10^{-15}$, we need $r = 0.016$, i.e, $n = 63$
  - $\rightarrow$ Repeat each bit sixty-three times!
- Error probability p_{error} can be arbitrarily reduced, but the price paid is huge. The rate r of information transfer is reduced significantly.

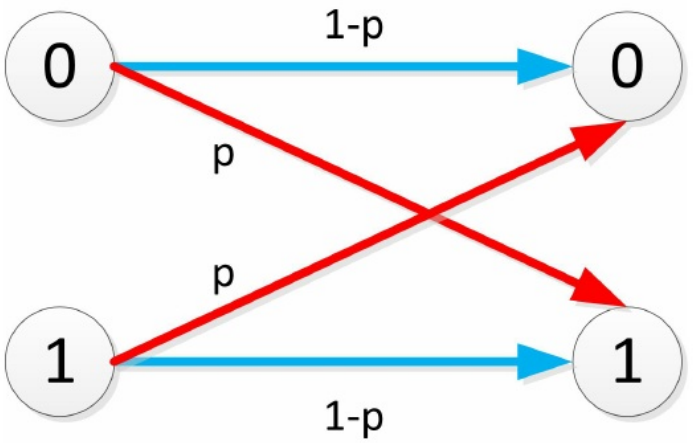
Several Questions about Information Transfer over BSC(p)

- The binary symmetric channel (BSC(p)) model under consideration
  - We will assume that the value of p is known both at the sender and the receiver



Several Questions about Information Transfer over BSC(p)

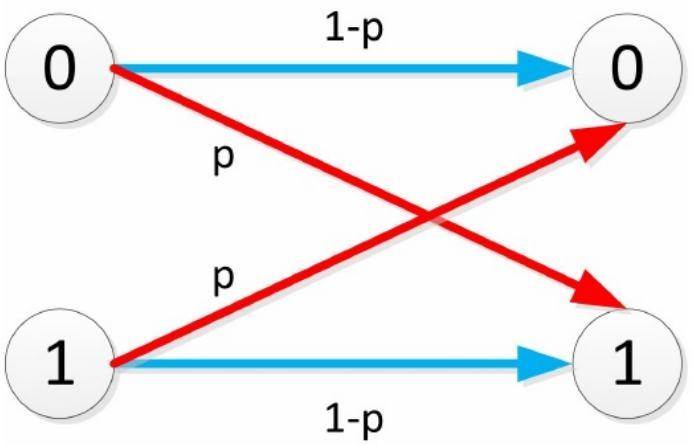
- The binary symmetric channel (BSC(p)) model under consideration
  - ▷ We will assume that the value of p is known both at the sender and the receiver



- Let us say that the transmitter is a Bernouilli($q = 0.5$) source which sends 1000 bits/sec, and the BSC p value is 0. What is the rate at which the information is received over this BSC($p = 0$)?

Several Questions about Information Transfer over BSC(p)

- The binary symmetric channel (BSC(p)) model under consideration
  - We will assume that the value of p is known both at the sender and the receiver



- Let us say that the transmitter is a Bernouilli($q = 0.5$) source which sends 1000 bits/sec, and the BSC p value is 0. What is the rate at which the information is received over this BSC($p = 0$)?
  - Obviously: 1000 bits/sec

Several Questions about Information Transfer over BSC(p)

- Question 1: Suppose $p = 0.1$. What is the rate at which the information is received over this BSC($p = 0.1$)?

Several Questions about Information Transfer over BSC(p)

- Question 1: Suppose $p = 0.1$. What is the rate at which the information is received over this BSC($p = 0.1$)?
- Question 2: For what value of p does the rate of information transfer over the BSC become zero?

Several Questions about Information Transfer over BSC(p)

- Question 1: Suppose $p = 0.1$. What is the rate at which the information is received over this BSC($p = 0.1$)?
- Question 2: For what value of p does the rate of information transfer over the BSC become zero?
  - ▷ One answer to Q2: the information transfer rate becomes zero when $p = 1$
    - → This answer is not right. When $p = 1$, the BSC behaves in a deterministic - nonrandom - manner since it flips each bit without fail. Therefore, the receiver just flips each received bit at the output of the channel to obtain the data at the full rate of 1000 bits/sec with zero probability of error

Several Questions about Information Transfer over BSC(p)

- Question 1: Suppose $p = 0.1$. What is the rate at which the information is received over this BSC($p = 0.1$)?
- Question 2: For what value of p does the rate of information transfer over the BSC become zero?
  - ▷ One answer to Q2: the information transfer rate becomes zero when $p = 1$
    - → This answer is not right. When $p = 1$, the BSC behaves in a deterministic - nonrandom - manner since it flips each bit without fail. Therefore, the receiver just flips each received bit at the output of the channel to obtain the data at the full rate of 1000 bits/sec with zero probability of error
  - ▷ One answer to Q1: 900 bits/sec (just subtract 10% of bits that are going to be received in error)

Several Questions about Information Transfer over BSC(p)

- Question 1: Suppose $p = 0.1$. What is the rate at which the information is received over this BSC($p = 0.1$)?
- Question 2: For what value of p does the rate of information transfer over the BSC become zero?
  - ▷ One answer to Q2: the information transfer rate becomes zero when $p = 1$
    - → This answer is not right. When $p = 1$, the BSC behaves in a deterministic - nonrandom - manner since it flips each bit without fail. Therefore, the receiver just flips each received bit at the output of the channel to obtain the data at the full rate of 1000 bits/sec with zero probability of error
  - ▷ One answer to Q1: 900 bits/sec (just subtract 10% of bits that are going to be received in error)
    - → This answer is not correct either. The strategy of determining the received data rate as $1000 \times (1 - p)$ does not work as we have seen earlier for the case of $p = 1$.

Several Questions about Information Transfer over BSC(p)

- Question 1: Suppose $p = 0.1$. What is the rate at which the information is received over this BSC($p = 0.1$)?
- Question 2: For what value of p does the rate of information transfer over the BSC become zero?
  - ▷ One answer to Q2: the information transfer rate becomes zero when $p = 1$
    - → This answer is not right. When $p = 1$, the BSC behaves in a deterministic - nonrandom - manner since it flips each bit without fail. Therefore, the receiver just flips each received bit at the output of the channel to obtain the data at the full rate of 1000 bits/sec with zero probability of error
  - ▷ One answer to Q1: 900 bits/sec (just subtract 10% of bits that are going to be received in error)
    - → This answer is not correct either. The strategy of determining the received data rate as $1000 \times (1 - p)$ does not work as we have seen earlier for the case of $p = 1$.
    - → Also, although the receiver knows that the BSC flips 10% of the bits in average, how can it get the remaining 900 bits/sec since it does not know which bits the BSC has flipped?

Several Questions about Information Transfer over BSC(p)

- These adhoc answers - based on guesswork - don't turn out right, they do not satisfy the logical expectation

Several Questions about Information Transfer over BSC(p)

- These adhoc answers - based on guesswork - don't turn out right, they do not satisfy the logical expectation
- Could it be that if we answer these questions using the mathematics of the information transfer that we have studied, the results are logically more satisfactory?

Several Questions about Information Transfer over BSC(p)

Answer of Q2

- To answer Question 2, note that when $p = 0.5$, $p(X \mid Y) = p(X)$.

Several Questions about Information Transfer over BSC(p)

Answer of Q2

- To answer Question 2, note that when $p = 0.5$, $p(X \mid Y) = p(X)$.
  - i.e., the BSC output Y becomes independent of the transmitted X

Several Questions about Information Transfer over BSC(p)

Answer of Q2

- To answer Question 2, note that when $p = 0.5$, $p(X \mid Y) = p(X)$.
  - ▷ i.e., the BSC output Y becomes independent of the transmitted X

Why?

Several Questions about Information Transfer over BSC(p)

Answer of Q2

- To answer Question 2, note that when $p = 0.5$, $p(X \mid Y) = p(X)$.
  - i.e., the BSC output Y becomes independent of the transmitted X

Why?

- Since

$$p(Y = 0 \mid X = 0) = 1 - p = 0.5; \quad p(Y = 0 \mid X = 1) = p = 0.5,$$

Several Questions about Information Transfer over BSC(p)

Answer of Q2

- To answer Question 2, note that when $p = 0.5$, $p(X \mid Y) = p(X)$.
  - i.e., the BSC output Y becomes independent of the transmitted X

Why?

- Since

$$p(Y = 0 \mid X = 0) = 1 - p = 0.5; \quad p(Y = 0 \mid X = 1) = p = 0.5,$$

- We have the following:

$$\begin{aligned} p(Y = 0) &= p(Y = 0, X = 0) + p(Y = 0, X = 1) \\ &= p(Y = 0 \mid X = 0)p(X = 0) + p(Y = 0 \mid X = 1)p(X = 1) \\ &= 0.5 \times (1 - q + q) = 0.5 \end{aligned}$$

Several Questions about Information Transfer over BSC(p)

Answer of Q2

- To answer Question 2, note that when $p = 0.5$, $p(X \mid Y) = p(X)$.
  - i.e., the BSC output Y becomes independent of the transmitted X

Why?

- Since

$$p(Y = 0 \mid X = 0) = 1 - p = 0.5; \quad p(Y = 0 \mid X = 1) = p = 0.5,$$

- We have the following:

$$\begin{aligned} p(Y = 0) &= p(Y = 0, X = 0) + p(Y = 0, X = 1) \\ &= p(Y = 0 \mid X = 0)p(X = 0) + p(Y = 0 \mid X = 1)p(X = 1) \\ &= 0.5 \times (1 - q + q) = 0.5 \end{aligned}$$

- i.e., regardless of the value of q, $p(Y = 0) = p(Y = 1) = 0.5$

Several Questions about Information Transfer over BSC(p)

Answer of Q2

- Therefore,

$$\begin{aligned} p(X=0 \mid Y=0) &= p(Y=0 \mid X=0)p(X=0)/p(Y=0) \\ &= 0.5 \times p(X=0)/0.5 \\ &= p(X=0). \end{aligned}$$

Several Questions about Information Transfer over BSC(p)

Answer of Q2

- Therefore,

$$\begin{aligned} p(X=0 \,|\, Y=0) &= p(Y=0 \,|\, X=0)p(X=0)/p(Y=0) \\ &= 0.5 \times p(X=0)/0.5 \\ &= p(X=0). \end{aligned}$$

- Similarly, it can be shown than

$$\begin{aligned} p(X=1 \,|\, Y=0) &= p(X=1) \\ p(X=0 \,|\, Y=1) &= p(X=0) \\ p(X=1 \,|\, Y=1) &= p(X=1). \end{aligned}$$

Several Questions about Information Transfer over BSC(p)

Answer of Q2

- Therefore,

$$\begin{aligned} p(X=0 \mid Y=0) &= p(Y=0 \mid X=0)p(X=0)/p(Y=0) \\ &= 0.5 \times p(X=0)/0.5 \\ &= p(X=0). \end{aligned}$$

- Similarly, it can be shown than

$$\begin{aligned} p(X=1 \mid Y=0) &= p(X=1) \\ p(X=0 \mid Y=1) &= p(X=0) \\ p(X=1 \mid Y=1) &= p(X=1). \end{aligned}$$

- Thus, we have shown that $p(X \mid Y) = p(X)$. This, however, implies that the conditional entropy $H(X \mid Y) = H(X)$

Several Questions about Information Transfer over BSC(p)

Answer of Q2

- Therefore,

$$\begin{aligned} p(X=0 \mid Y=0) &= p(Y=0 \mid X=0)p(X=0)/p(Y=0) \\ &= 0.5 \times p(X=0)/0.5 \\ &= p(X=0). \end{aligned}$$

- Similarly, it can be shown than

$$\begin{aligned} p(X=1 \mid Y=0) &= p(X=1) \\ p(X=0 \mid Y=1) &= p(X=0) \\ p(X=1 \mid Y=1) &= p(X=1). \end{aligned}$$

- Thus, we have shown that $p(X \mid Y) = p(X)$. This, however, implies that the conditional entropy $H(X \mid Y) = H(X)$
- Therefore, the information transfer $I(X;Y) = H(X) - H(X \mid Y) = 0$.

Several Questions about Information Transfer over BSC(p)

Answer of Q2

- Therefore,

$$\begin{aligned} p(X=0 \mid Y=0) &= p(Y=0 \mid X=0)p(X=0)/p(Y=0) \\ &= 0.5 \times p(X=0)/0.5 \\ &= p(X=0). \end{aligned}$$

- Similarly, it can be shown than

$$\begin{aligned} p(X=1 \mid Y=0) &= p(X=1) \\ p(X=0 \mid Y=1) &= p(X=0) \\ p(X=1 \mid Y=1) &= p(X=1). \end{aligned}$$

- Thus, we have shown that $p(X \mid Y) = p(X)$. This, however, implies that the conditional entropy $H(X \mid Y) = H(X)$
- Therefore, the information transfer $I(X;Y) = H(X) - H(X \mid Y) = 0$.
  - Since Y becomes independent of X in this case, X fails to exert the influence on Y. The transmitter ceases to supply the information to the receiver.

Several Questions about Information Transfer over BSC(p)

Answer of Q1

- To answer Question 1, we show next that for the BSC(p), $H(X \mid Y) = H_b(p)$, where

$$H_b(p) = -(p \log_2(p) + (1-p) \log_2(1-p)$$

is the binary Entropy function

Several Questions about Information Transfer over BSC(p)

Answer of Q1

- To answer Question 1, we show next that for the BSC(p), $H(X \mid Y) = H_b(p)$, where

$$H_b(p) = -(p \log_2(p) + (1-p) \log_2(1-p)$$

is the binary Entropy function

  - Since

$$p(Y=0 \mid X=0) = 1-p; \quad p(Y=0 \mid X=1) = p,$$

and

$$p(X=0) = p(X=1) = 0.5$$

  - we obtain the following:

Several Questions about Information Transfer over BSC(p)

Answer of Q1

- To answer Question 1, we show next that for the BSC(p), $H(X \mid Y) = H_b(p)$, where

$$H_b(p) = -(p \log_2(p) + (1-p) \log_2(1-p)$$

is the binary Entropy function

  - Since

$$p(Y=0 \mid X=0) = 1-p; \quad p(Y=0 \mid X=1) = p,$$

and

$$p(X=0) = p(X=1) = 0.5$$

  - we obtain the following:

$$\begin{aligned} p(Y=0) &= p(Y=0, X=0) + p(Y=0, X=1) \\ &= p(Y=0 \mid X=0)p(X=0) + p(Y=0 \mid X=1)p(X=1) \\ &= 0.5 \times (1-p+p) \\ &= 0.5 \end{aligned}$$

Several Questions about Information Transfer over BSC(p)

Answer of Q1

- Therefore,

$$\begin{aligned} p(X=0 \mid Y=0) &= p(Y=0 \mid X=0)p(X=0)/p(Y=0) \\ &= (1-p) \times 0.5/0.5 = 1-p. \end{aligned}$$

- Similarly,

$$\begin{aligned} p(X=1 \mid Y=0) &= p(Y=0 \mid X=1)p(X=1)/p(Y=0) \\ &= p \times 0.5/0.5 = p. \end{aligned}$$

- Thus, when $Y=0$ is received, the information source X reduces to the Bernouilli(p) RV from Bernouilli($q=0.5$) RV
- Similarly, when $Y=1$ is received, the information source X again reduces to the Bernouilli(p) RV from Bernouilli($q=0.5$) RV
  - ▷ This is evident by the symmetry of the BSC, and it can be mathematically shown by repeating the above derivation for $Y=1$
- Therefore, $H(X \mid Y) = H_b(p)$ is proven

Several Questions about Information Transfer over BSC(p)

Answer of Q1

- Therefore,

$$\begin{aligned} p(X=0 \mid Y=0) &= p(Y=0 \mid X=0)p(X=0)/p(Y=0) \\ &= (1-p) \times 0.5/0.5 = 1-p. \end{aligned}$$

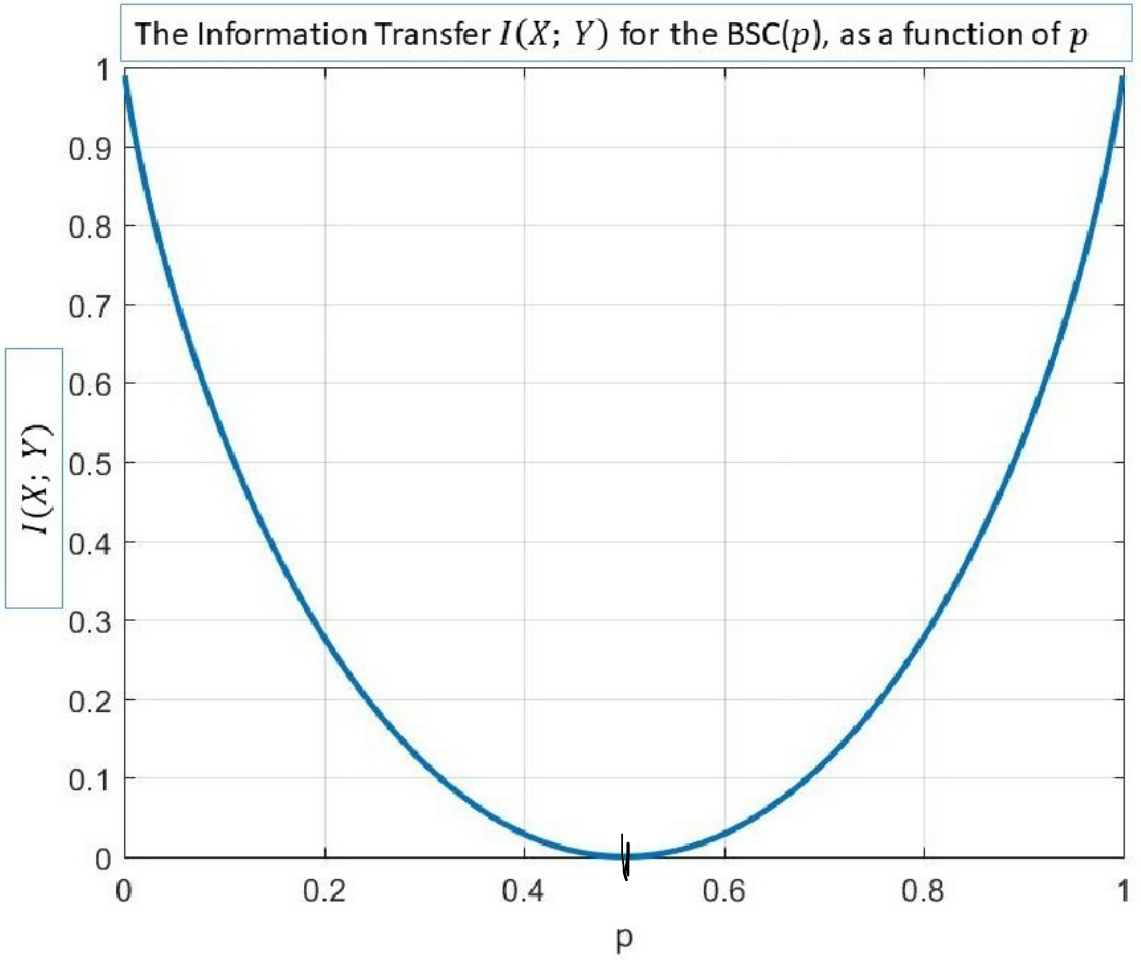
- Similarly,

$$\begin{aligned} p(X=1 \mid Y=0) &= p(Y=0 \mid X=1)p(X=1)/p(Y=0) \\ &= p \times 0.5/0.5 = p. \end{aligned}$$

- Thus, when $Y=0$ is received, the information source X reduces to the Bernouilli(p) RV from Bernouilli($q=0.5$) RV
- Similarly, when $Y=1$ is received, the information source X again reduces to the Bernouilli(p) RV from Bernouilli($q=0.5$) RV
  - ▷ This is evident by the symmetry of the BSC, and it can be mathematically shown by repeating the above derivation for $Y=1$
- Therefore, $H(X \mid Y) = H_b(p)$ is proven
- Since $H(X)=1$ bit, the information transfer equals $I(X;Y) = 1 - H_b(p)$ bits

Performance of the Ideal Channel Code

over BSC(p)



Repetition Code versus the Ideal Channel Code

over BSC(p)

- A theoretical possibility: over BSC($p = 0.1$), the information can be transmitted with arbitrarily low p_{error} with rate r as high as $1 - H_b(p = 0.1) = 0.53$!
  - $\rightarrow$ Rate r does not have to be reduced to near zero
  - $\rightarrow$ In fact, p_{error} can be reduced to 10^{-15}, or 10^{-100} or 10^{-1000} at the same rate of 0.53

Repetition Code versus the Ideal Channel Code

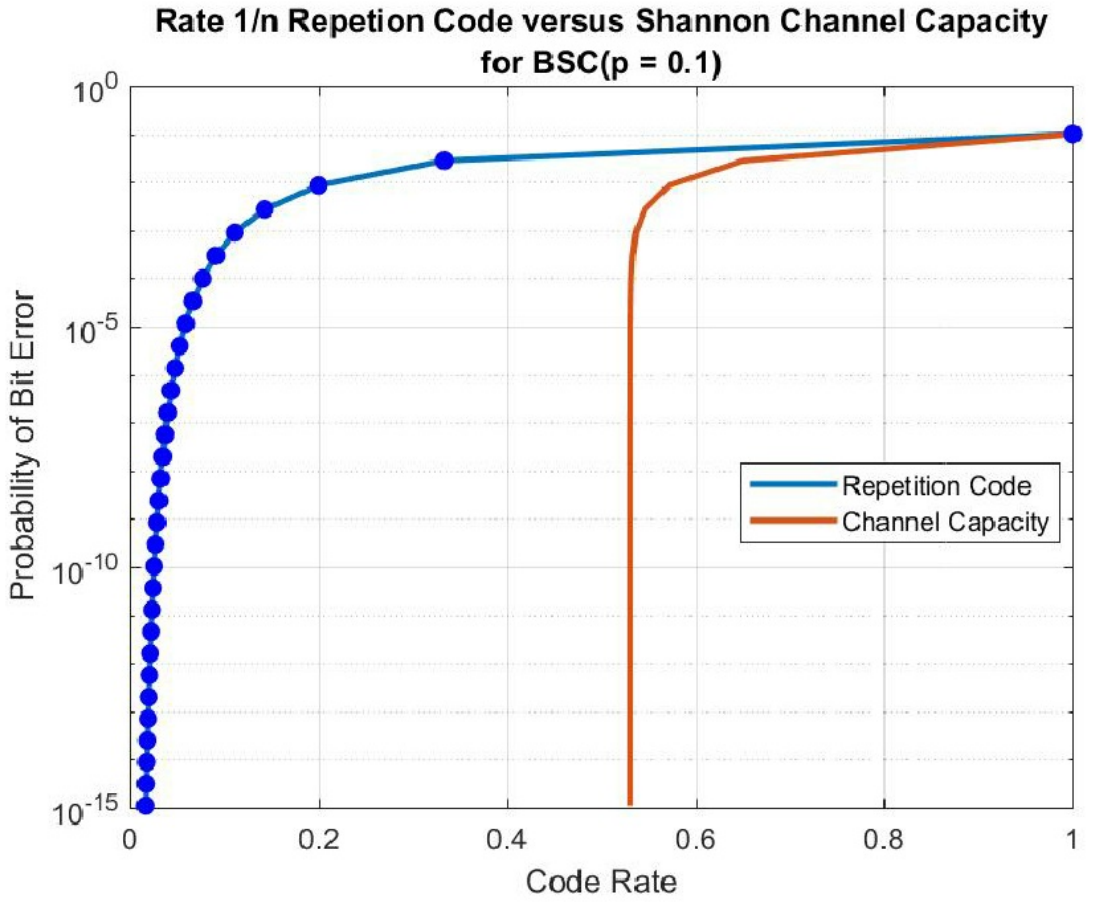
over BSC(p)

- A theoretical possibility: over BSC($p = 0.1$), the information can be transmitted with arbitrarily low p_{error} with rate r as high as $1 - H_b(p = 0.1) = 0.53$!
  - → Rate r does not have to be reduced to near zero
  - → In fact, p_{error} can be reduced to 10^{-15}, or 10^{-100} or 10^{-1000} at the same rate of 0.53
    - ▷ Prior to 1948, the general belief was that attempting to reduce $p_{error} \to 0$ requires $r \to 0$.
    - ▷ Shannon showed that nonzero, positive-valued, r is possible even if p_{error} is required to approach 0

Repetition Code versus the Ideal Channel Code

over BSC(p)

- The information transfer $I(X;Y)$ is called the channel capacity



Channel Coding Theorem

- A fundamental theorem of the Information Theory for the Noisy Channels [Shannon 1948 paper]:
  - → The rate r of the channel coding scheme $r \leq I(X;Y) = H(X) - H(X \mid Y)$.
    - ▷ Rate can be made arbitrarily close to, but not greater than, $I(X;Y)$

$r = \frac{k}{N} = \frac{\#\text{ of Inf. bits}}{\#\text{ of encoded bits}}$

Channel Coding Theorem

- A fundamental theorem of the Information Theory for the Noisy Channels [Shannon 1948 paper]:
  - → The rate r of the channel coding scheme $r \leq I(X;Y) = H(X) - H(X \mid Y)$.
    - ▷ Rate can be made arbitrarily close to, but not greater than, $I(X;Y)$
  - → This is the (Noisy) Channel Coding Theorem

Channel Coding Theorem

- A fundamental theorem of the Information Theory for the Noisy Channels [Shannon 1948 paper]:
  - → The rate r of the channel coding scheme $r \leq I(X;Y) = H(X) - H(X \mid Y)$.
    - ▷ Rate can be made arbitrarily close to, but not greater than, $I(X;Y)$
  - → This is the (Noisy) Channel Coding Theorem
- How to prove this theorem?

Design of Channel Coding

- We have seen the objective of the channel coding: to make a noisy channel appear noiseless

Design of Channel Coding

- We have seen the objective of the channel coding: to make a noisy channel appear noiseless
- We have also previewed the design of the channel coding: introduce the redundancies in the transmitted bits

Design of Channel Coding

- We have seen the objective of the channel coding: to make a noisy channel appear noiseless
- We have also previewed the design of the channel coding: introduce the redundancies in the transmitted bits
- Now we look at the practical design techniques for the channel coding, and also consider how the redundant bits that the channel encoder adds help in the noise removal

Number of Valid Codwords versus Number of Possible Received Words

- Notations:

Number of Valid Codwords versus Number of Possible Received Words

- Notations:
  - n: length of codeword in bits; $1 \leq k \leq n$: length of the information bit sequence in bits

Number of Valid Codwords versus Number of Possible Received Words

- Notations:
  - n: length of codeword in bits; $1 \leq k \leq n$: length of the information bit sequence in bits
  - $0 \leq r = k/n \leq 1$: the rate of the channel code

Number of Valid Codwords versus Number of Possible Received Words

- Notations:
  - n: length of codeword in bits; $1 \leq k \leq n$: length of the information bit sequence in bits
  - $0 \leq r = k/n \leq 1$: the rate of the channel code
  - We will interchangeably use small-case or capitalized notations k or K; n or N; except, for the rate r, for which the capitalized notation R is reserved for the source coding rate

Number of Valid Codwords versus Number of Possible Received Words

- Notations:
  - n: length of codeword in bits; $1 \leq k \leq n$: length of the information bit sequence in bits
  - $0 \leq r = k/n \leq 1$: the rate of the channel code
  - We will interchangeably use small-case or capitalized notations k or K; n or N; except, for the rate r, for which the capitalized notation R is reserved for the source coding rate
- A consequence of $k < n$:

Number of Valid Codwords versus Number of Possible Received Words

- Notations:
  - n: length of codeword in bits; $1 \leq k \leq n$: length of the information bit sequence in bits
  - $0 \leq r = k/n \leq 1$: the rate of the channel code
  - We will interchangeably use small-case or capitalized notations k or K; n or N; except, for the rate r, for which the capitalized notation R is reserved for the source coding rate
- A consequence of $k < n$:
  - Since the length of the information bit sequence equals k bits, there are total 2^k valid codewords that the encoder can generate

Number of Valid Codwords versus Number of Possible Received Words

- Notations:
  - n: length of codeword in bits; $1 \leq k \leq n$: length of the information bit sequence in bits
  - $0 \leq r = k/n \leq 1$: the rate of the channel code
  - We will interchangeably use small-case or capitalized notations k or K; n or N; except, for the rate r, for which the capitalized notation R is reserved for the source coding rate
- A consequence of $k < n$:
  - Since the length of the information bit sequence equals k bits, there are total 2^k valid codewords that the encoder can generate
  - However, since the length of the received bit sequence is n bits, there are total $2^n \geq 2^k$ bit sequences that the receiver can observe

Number of Valid Codwords versus Number of Possible Received Words

- Notations:
  - n: length of codeword in bits; $1 \leq k \leq n$: length of the information bit sequence in bits
  - $0 \leq r = k/n \leq 1$: the rate of the channel code
  - We will interchangeably use small-case or capitalized notations k or K; n or N; except, for the rate r, for which the capitalized notation R is reserved for the source coding rate
- A consequence of $k < n$:
  - Since the length of the information bit sequence equals k bits, there are total 2^k valid codewords that the encoder can generate
  - However, since the length of the received bit sequence is n bits, there are total $2^n \geq 2^k$ bit sequences that the receiver can observe
  - In the absence of the noise, the receiver observes only one of 2^k valid codewords; however, in the presence of noise, the receiver can also additionally observe any one of $2^n - 2^k$ invalid codewords

Number of Valid Codwords versus Number of Possible Received Words

- Notations:
  - n: length of codeword in bits; $1 \leq k \leq n$: length of the information bit sequence in bits
  - $0 \leq r = k/n \leq 1$: the rate of the channel code
  - We will interchangeably use small-case or capitalized notations k or K; n or N; except, for the rate r, for which the capitalized notation R is reserved for the source coding rate
- A consequence of $k < n$:
  - Since the length of the information bit sequence equals k bits, there are total 2^k valid codewords that the encoder can generate
  - However, since the length of the received bit sequence is n bits, there are total $2^n \geq 2^k$ bit sequences that the receiver can observe
  - In the absence of the noise, the receiver observes only one of 2^k valid codewords; however, in the presence of noise, the receiver can also additionally observe any one of $2^n - 2^k$ invalid codewords
  - The main idea of channel coding: make the receiver intelligent so that it can decide whether it is observing an invalid codeword (in which case it declares that "the channel has introduced an error") and possibly correct this error

Number of Valid Codwords versus Number of Possible Received Words

- Notations:
  - n: length of codeword in bits; $1 \leq k \leq n$: length of the information bit sequence in bits
  - $0 \leq r = k/n \leq 1$: the rate of the channel code
  - We will interchangeably use small-case or capitalized notations k or K; n or N; except, for the rate r, for which the capitalized notation R is reserved for the source coding rate
- A consequence of $k < n$:
  - Since the length of the information bit sequence equals k bits, there are total 2^k valid codewords that the encoder can generate
  - However, since the length of the received bit sequence is n bits, there are total $2^n \geq 2^k$ bit sequences that the receiver can observe
  - In the absence of the noise, the receiver observes only one of 2^k valid codewords; however, in the presence of noise, the receiver can also additionally observe any one of $2^n - 2^k$ invalid codewords
  - The main idea of channel coding: make the receiver intelligent so that it can decide whether it is observing an invalid codeword (in which case it declares that "the channel has introduced an error") and possibly correct this error
- To obtain a conceptual understanding of how the receiver can do the above, we introduce the notion of Hamming Distance among the codewords

Black square of previous slide

The probability of a "typical" sequence of length N bits out of Bernoulli(q) source X, as $N \to \infty$, is equal to: $(1-q)^{N-Nq} (q)^{Nq}$

Aditya
(224)

The set of all such typical sequences has a cardinality K, where K is such that

$$K \times q^{Nq} (1-q)^{N(1-q)} = 1$$

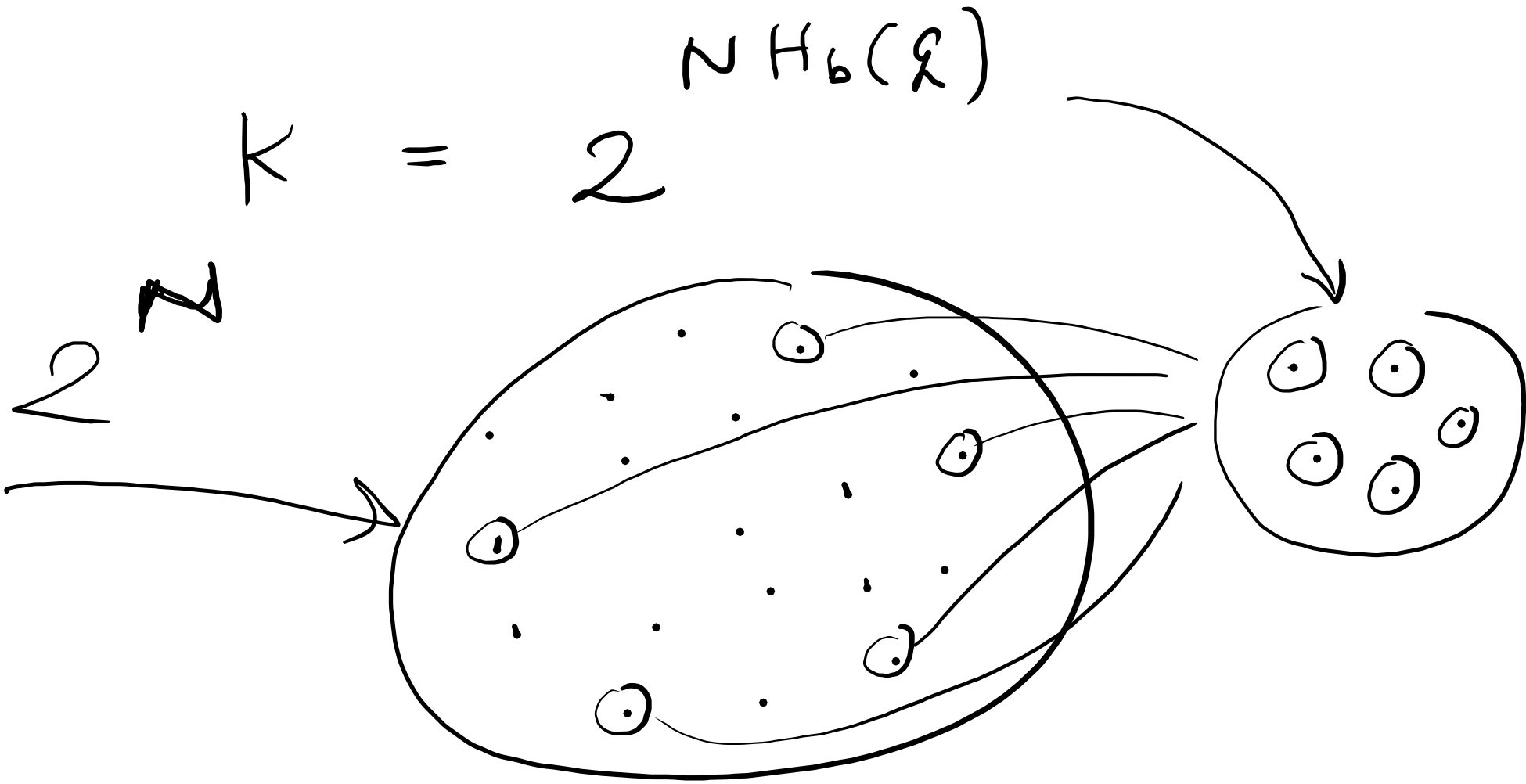
$$\Rightarrow K = q^{-Nq} (1-q)^{-N(1-q)}$$

$$\log_2(K) = \log_2$$

$$= \log_2\left(q^{-Nq} (1-q)^{-N(1-q)} \right)$$

$$= N\left(-q \log_2(q) - (1-q)\log_2(1-q) \right)$$

$$= N H_b(q)$$


$N H_6(2)$
$k = 2$
2^N

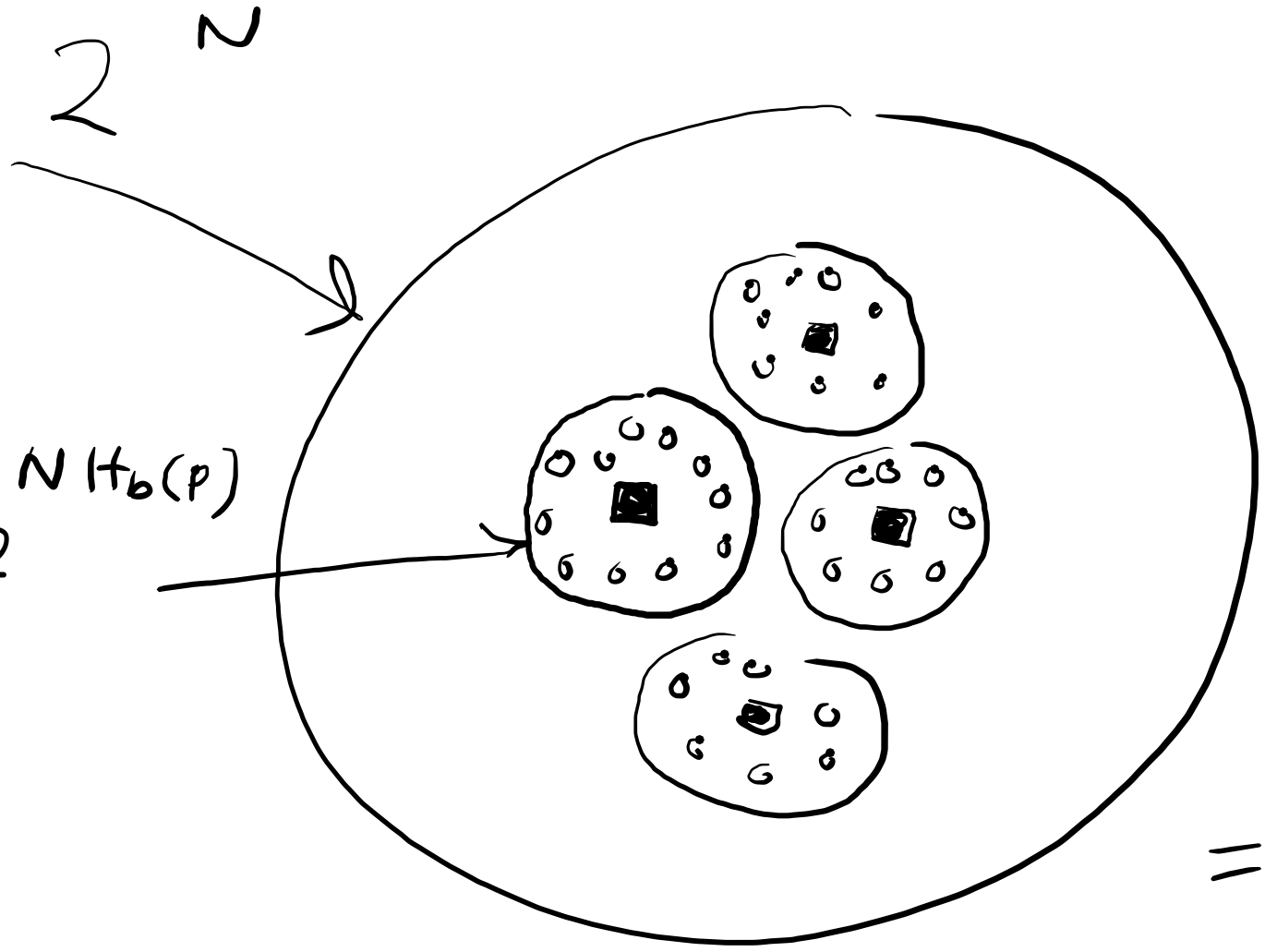
BSC(p):

$$Y = X + e$$

mod-2

Total # of sequences of the RV e:

$$2^{NH_b(p)}$$



$$\frac{2^N}{2^{NH_b(p)}} = 2^{N(1-H_b(p))} = 2^{NI(X;Y)}$$

Hamming Weight and Hamming Distance

for Binary Sequences

- Hamming Weight is simply the number of ones in the sequence

Hamming Weight and Hamming Distance

for Binary Sequences

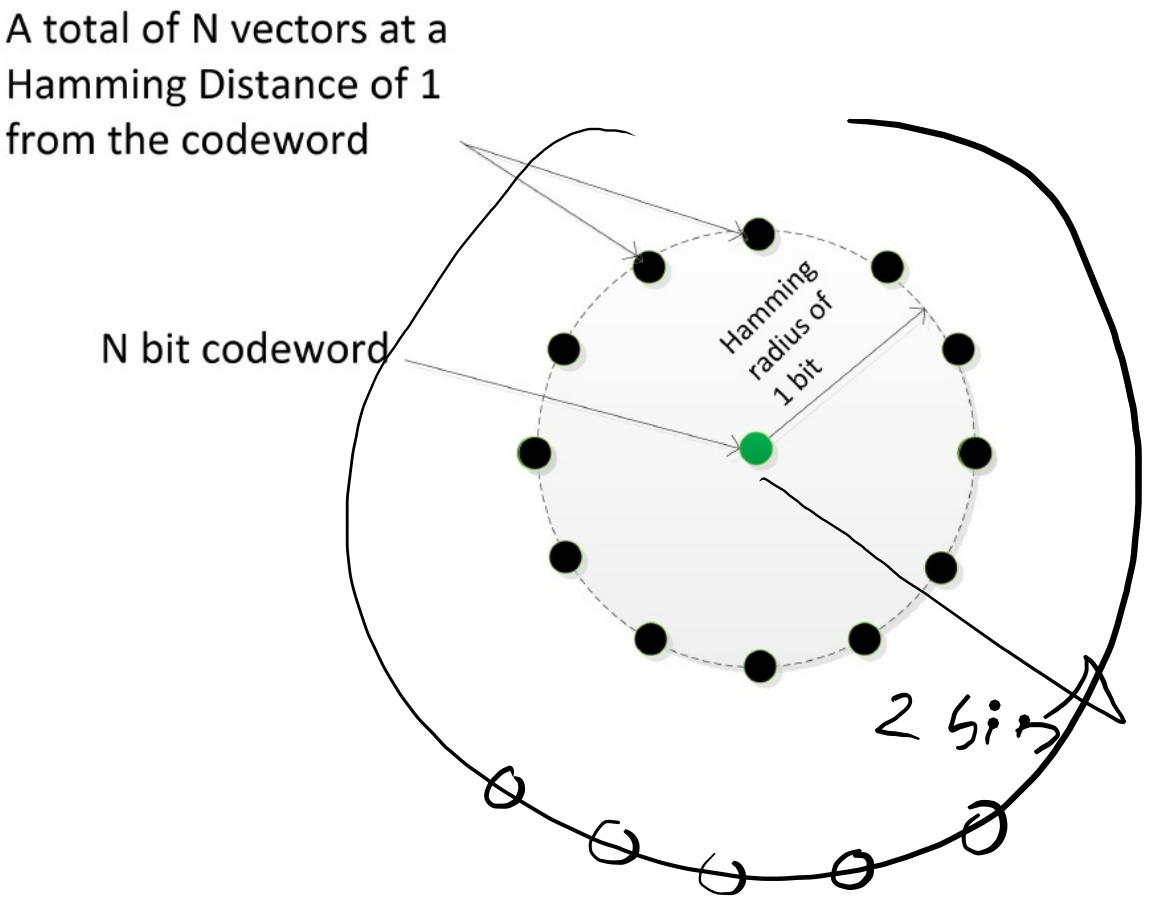
- Hamming Weight is simply the number of ones in the sequence
- Hamming Distance $d_H^{m,n}$:
  - → Let $\mathbf{c}_m$ and $\mathbf{c}_n$ be two codewords, and $\mathbf{e}_{m,n} = \mathbf{c}_m \oplus \mathbf{c}_n$ be the difference vector (also binary) between these codewords
    - ▷ $\mathbf{e}_{m,n}$ has ones only in those places where the bits of $\mathbf{c}_m$ and $\mathbf{c}_n$ are differing; $\mathbf{e}_{m,n}$ is zero otherwise
  - → Hamming Distance $d_H^{m,n}$ between $\mathbf{c}_m$ and $\mathbf{c}_n$ is the Hamming Weight of $\mathbf{e}_{m,n}$
    - ▷ Hamming Distance is simply the number of places in which two binary sequences differ

Minimum Hamming Distance d_{min}

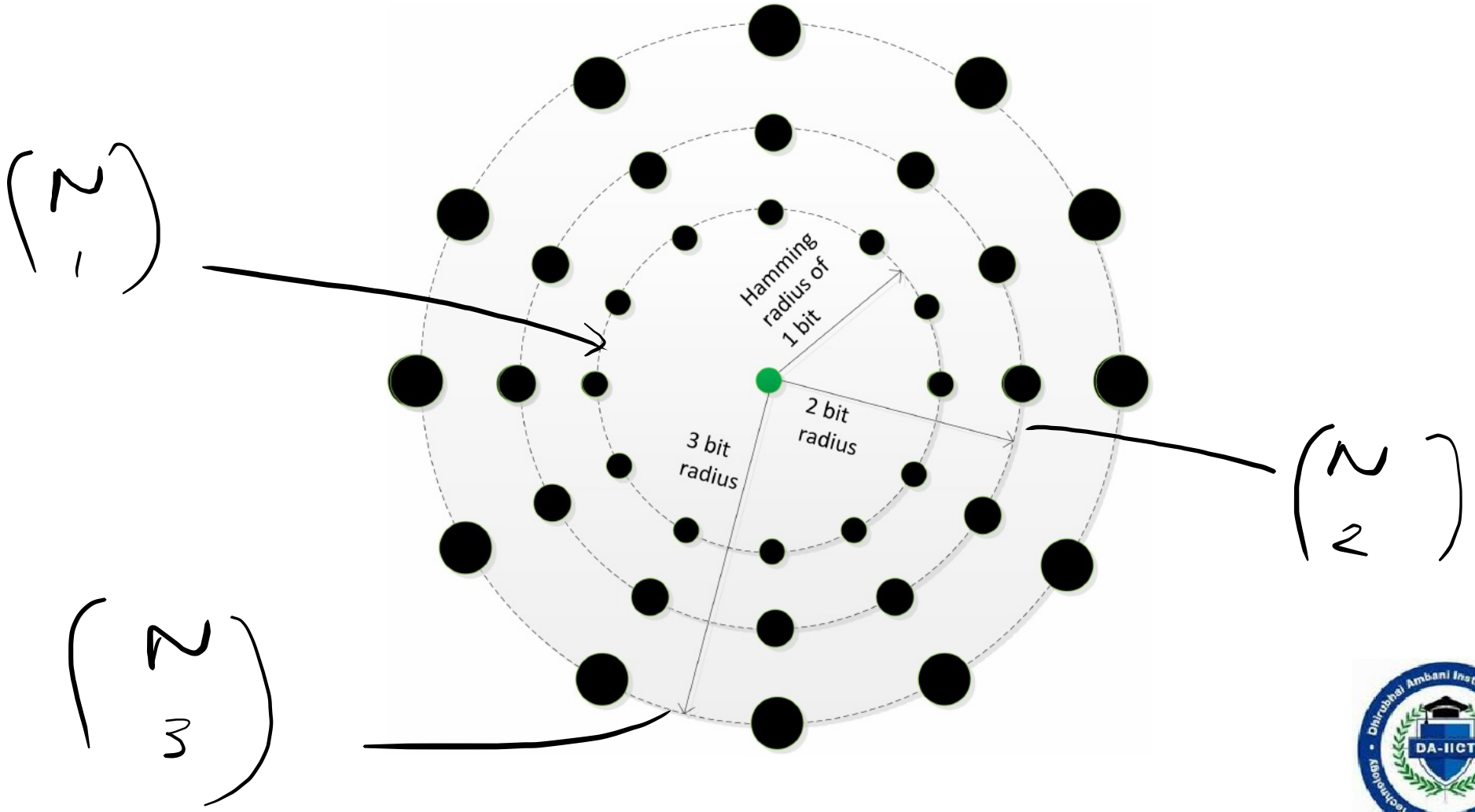
for A Channel Code

- $d_H^{\min}$ is defined for a channel code with rate $r = \frac{K}{N}$
  - $\rightarrow$ This code takes K bit information sequence and generates N bit codeword
    - Thus, there are a total of 2^K codewords
- $d_H^{\min}$ for this channel code is the minimum Hamming distance between any two pairs of this channel code
- $d_H^{\min}$ relates to the error detection and error correction capabilities of the channel code. This can be visualized by drawing Hamming Circles as shown next

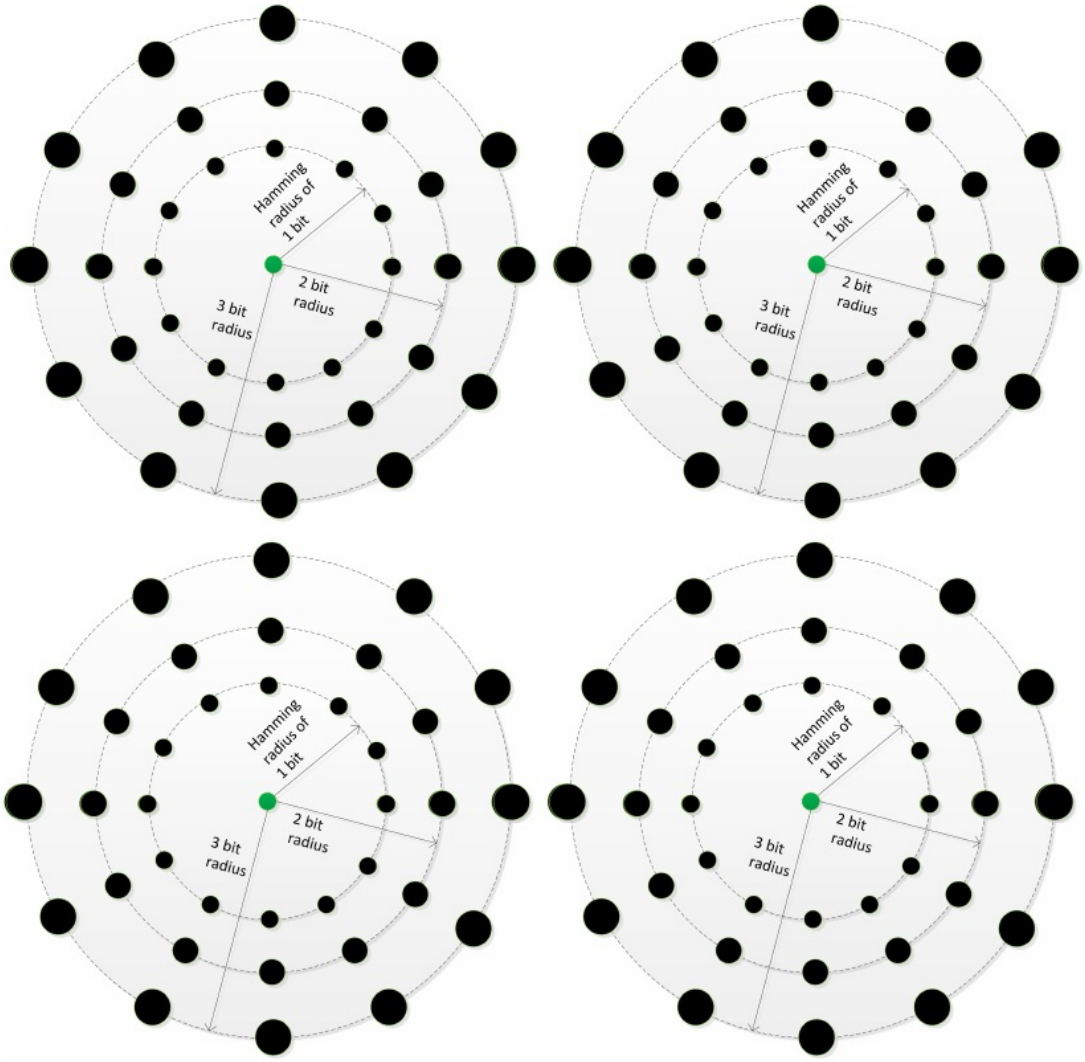
Hamming Circle of Radius 1 around a codeword



Hamming Circles around a codeword



Hamming Circles around several codewords



Minimum Hamming Distance Decoder

- Detects the transmitted codeword as the one at the center of the Hamming sphere in which the received word falls in

Minimum Hamming Distance Decoder

- Detects the transmitted codeword as the one at the center of the Hamming sphere in which the received word falls in
- Can be shown to be the optimal Bayesian (ML) receiver provided all 2^k transmitted codewords are equally likely to have been transmitted

Minimum Hamming Distance Decoder

- Detects the transmitted codeword as the one at the center of the Hamming sphere in which the received word falls in
- Can be shown to be the optimal Bayesian (ML) receiver provided all 2^k transmitted codewords are equally likely to have been transmitted
- We have actually proven this for the repetition code

Minimum Hamming Distance Decoder

- Detects the transmitted codeword as the one at the center of the Hamming sphere in which the received word falls in
- Can be shown to be the optimal Bayesian (ML) receiver provided all 2^k transmitted codewords are equally likely to have been transmitted
- We have actually proven this for the repetition code
- For practical channel coding design, this is the main challenge — since the both 2^k and 2^n become very large numbers as k and n increase. Therefore, a brute-force distance minimization becomes infeasible to implement

Error Detection t_d and Error Correction t_c Capability of a Channel Code

- Error detection capability t_d in bits:

Error Detection t_d and Error Correction t_c Capability of a Channel Code

- Error detection capability t_d in bits:
  - Defined as the maximum number of bits that can be flipped by the BSC above which the detector may not be able to identify that the BSC has introduced bit errors

Error Detection t_d and Error Correction t_c Capability of a Channel Code

- Error detection capability t_d in bits:
  - Defined as the maximum number of bits that can be flipped by the BSC above which the detector may not be able to identify that the BSC has introduced bit errors
- Error correction capability t_c in bits:

Error Detection t_d and Error Correction t_c Capability of a Channel Code

- Error detection capability t_d in bits:
  - Defined as the maximum number of bits that can be flipped by the BSC above which the detector may not be able to identify that the BSC has introduced bit errors
- Error correction capability t_c in bits:
  - Defined as the maximum number of bits that can be flipped by the BSC at or below which the minimum distance detector is guaranteed to decode the correct (i.e., the transmitted) codeword

Error Detection t_d and Error Correction t_c Capability of a Channel Code

- Error detection capability t_d in bits:
  - Defined as the maximum number of bits that can be flipped by the BSC above which the detector may not be able to identify that the BSC has introduced bit errors
- Error correction capability t_c in bits:
  - Defined as the maximum number of bits that can be flipped by the BSC at or below which the minimum distance detector is guaranteed to decode the correct (i.e., the transmitted) codeword
- Both t_d and t_c are functions of $d_H^{\min}$

Error Detection t_d and Error Correction t_c Capability of a Channel Code

- Error detection capability t_d in bits:
  - Defined as the maximum number of bits that can be flipped by the BSC above which the detector may not be able to identify that the BSC has introduced bit errors
- Error correction capability t_c in bits:
  - Defined as the maximum number of bits that can be flipped by the BSC at or below which the minimum distance detector is guaranteed to decode the correct (i.e., the transmitted) codeword
- Both t_d and t_c are functions of $d_H^{\min}$
- To see this, we will draw simpler, two-dimensional, versions of the Hamming spheres for different values of $d_H^{\min}$ (note that different $d_H^{\min}$ imply different coding schemes)

Error Detection and Error Correction Capabilities

of a Channel Code

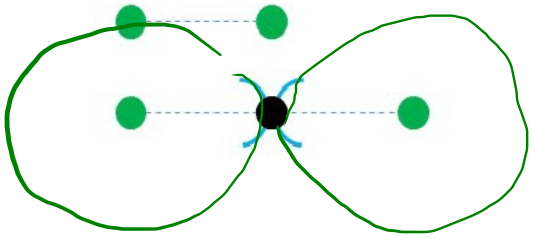
d_H^{min}	t_c	t_d
1 bit	0 bit	0 bit

: Valid codeword

: Not a valid codeword

Error Detection and Error Correction Capabilities

of a Channel Code



d_H^{min}	t_c	t_d
1 bit	0 bit	0 bit
2 bits	0 bit	1 bit

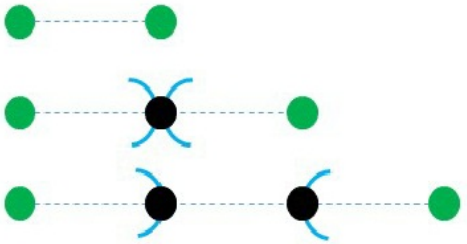
: Valid codeword

: Not a valid codeword

: Detection Boundary

Error Detection and Error Correction Capabilities

of a Channel Code



d_H^{min}	t_c	t_d
1 bit	0 bit	0 bit
2 bits	0 bit	1 bit
3 bits	1 bits	2 bit

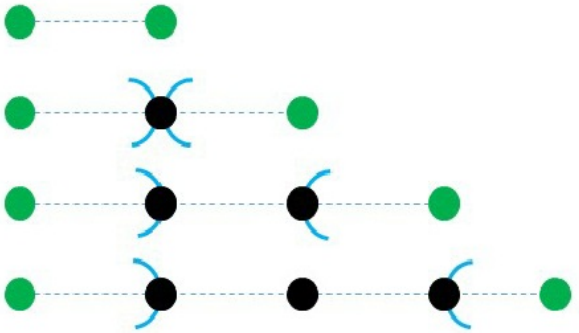
: Valid codeword

: Not a valid codeword

: Detection Boundary

Error Detection and Error Correction Capabilities

of a Channel Code



d_H^{min}	t_c	t_d
1 bit	0 bit	0 bit
2 bits	0 bit	1 bit
3 bits	1 bits	2 bit
4 bits	1 bits	3 bit

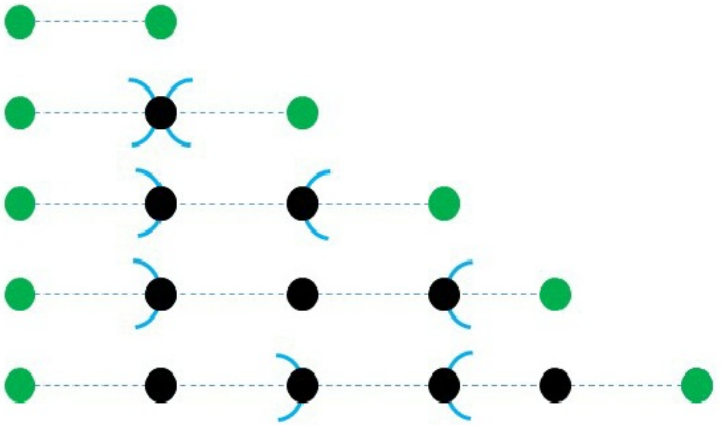
: Valid codeword

: Not a valid codeword

: Detection Boundary

Error Detection and Error Correction Capabilities

of a Channel Code



d_H^{min}	t_c	t_d
1 bit	0 bit	0 bit
2 bits	0 bit	1 bit
3 bits	1 bits	2 bit
4 bits	1 bits	3 bit
5 bits	2 bits	4 bit

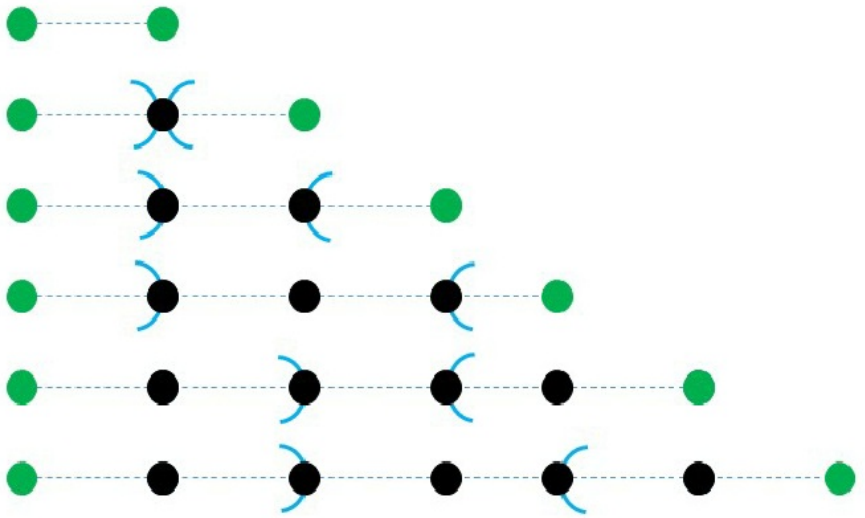
: Valid codeword

: Not a valid codeword

: Detection Boundary

Error Detection and Error Correction Capabilities

of a Channel Code



: Valid codeword

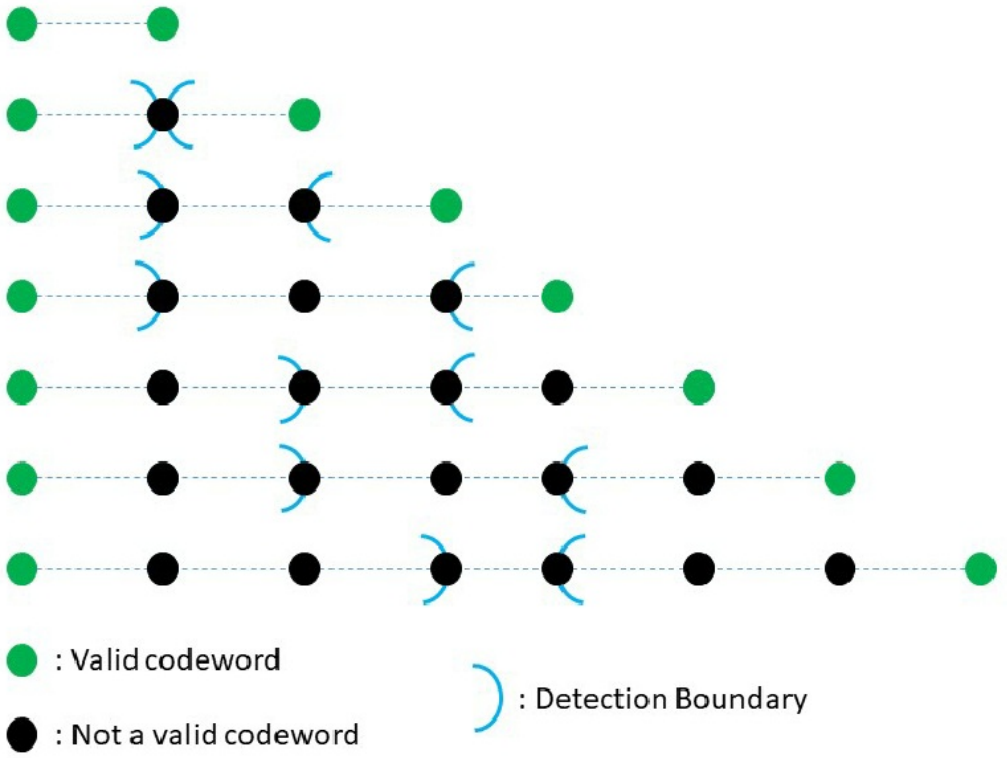
: Not a valid codeword

: Detection Boundary

d_H^{min}	t_c	t_d
1 bit	0 bit	0 bit
2 bits	0 bit	1 bit
3 bits	1 bits	2 bit
4 bits	1 bits	3 bit
5 bits	2 bits	4 bit
6 bits	2 bits	5 bit

Error Detection and Error Correction Capabilities

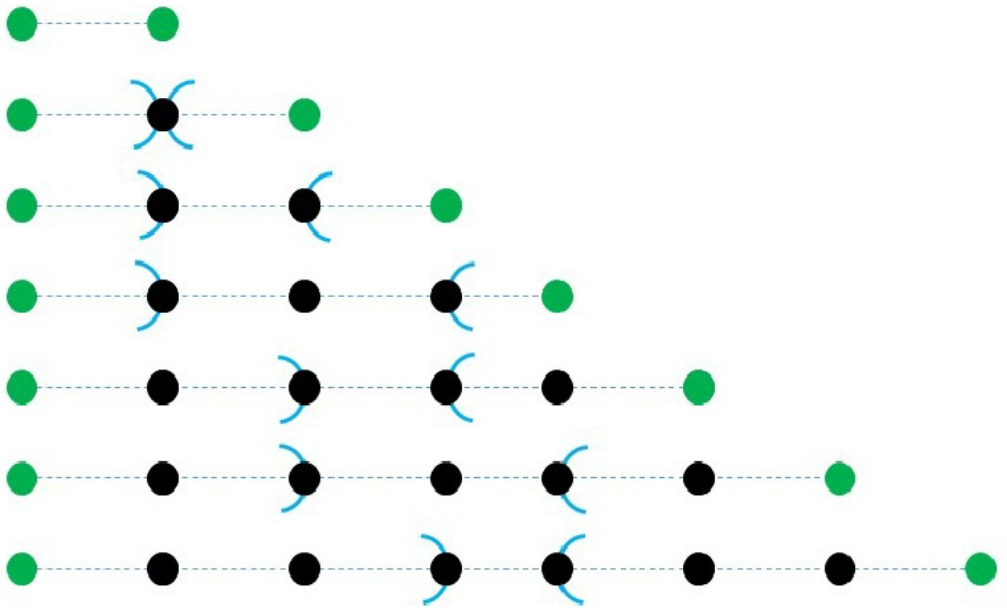
of a Channel Code



d_H^{min}	t_c	t_d
1 bit	0 bit	0 bit
2 bits	0 bit	1 bit
3 bits	1 bits	2 bit
4 bits	1 bits	3 bit
5 bits	2 bits	4 bit
6 bits	2 bits	5 bit
7 bits	3 bits	6 bit

Error Detection and Error Correction Capabilities

of a Channel Code



d_H^{min}	t_c	t_d
1 bit	0 bit	0 bit
2 bits	0 bit	1 bit
3 bits	1 bits	2 bit
4 bits	1 bits	3 bit
5 bits	2 bits	4 bit
6 bits	2 bits	5 bit
7 bits	3 bits	6 bit

- $t_d = d_H^{min} - 1$ **bits**
- $t_c = \left\lfloor \frac{d_H^{min} - 1}{2} \right\rfloor$ **bits**

Information and Encoded Bits as Vectors

- Consider the input block of k information bits and the output block of n encoded bits of a rate $r = k/n$ channel coding scheme as vectors:

Information and Encoded Bits as Vectors

- Consider the input block of k information bits and the output block of n encoded bits of a rate $r = k/n$ channel coding scheme as vectors:
  - $\rightarrow$ $\mathbf{m} = [m_0, m_1, \ldots, m_{k-1}]^T$: denotes a $k \times 1$ vector of message or information bits

Information and Encoded Bits as Vectors

- Consider the input block of k information bits and the output block of n encoded bits of a rate $r = k/n$ channel coding scheme as vectors:
  - $\rightarrow$ $\mathbf{m} = [m_0, m_1, \ldots, m_{k-1}]^T$: denotes a $k \times 1$ vector of message or information bits
    - Here, $\mathbf{v}^T$ denotes the transpose of a vector $\mathbf{v}$. In this case, it makes a horizontal $1 \times k$ vector a tall $k \times 1$ vector.

Information and Encoded Bits as Vectors

- Consider the input block of k information bits and the output block of n encoded bits of a rate $r = k/n$ channel coding scheme as vectors:
  - → $\mathbf{m} = [m_0, m_1, \ldots, m_{k-1}]^T$: denotes a $k \times 1$ vector of message or information bits
    - ▷ Here, $\mathbf{v}^T$ denotes the transpose of a vector $\mathbf{v}$. In this case, it makes a horizontal $1 \times k$ vector a tall $k \times 1$ vector.
  - → $\mathbf{c} = [c_0, c_1, \ldots, c_{n-1}]^T$: denotes an $n \times 1$ vector of encoded bits

Information and Encoded Bits as Vectors

- Consider the input block of k information bits and the output block of n encoded bits of a rate $r = k/n$ channel coding scheme as vectors:
  - → $\mathbf{m} = [m_0, m_1, \ldots, m_{k-1}]^T$: denotes a $k \times 1$ vector of message or information bits
    - ▷ Here, $\mathbf{v}^T$ denotes the transpose of a vector $\mathbf{v}$. In this case, it makes a horizontal $1 \times k$ vector a tall $k \times 1$ vector.
  - → $\mathbf{c} = [c_0, c_1, \ldots, c_{n-1}]^T$: denotes an $n \times 1$ vector of encoded bits
- We will relate $\mathbf{m}$ and $\mathbf{c}$ through matrix products. Several notes:

Information and Encoded Bits as Vectors

- Consider the input block of k information bits and the output block of n encoded bits of a rate $r = k/n$ channel coding scheme as vectors:
  - → $\mathbf{m} = [m_0, m_1, \ldots, m_{k-1}]^T$: denotes a $k \times 1$ vector of message or information bits
    - ▷ Here, $\mathbf{v}^T$ denotes the transpose of a vector $\mathbf{v}$. In this case, it makes a horizontal $1 \times k$ vector a tall $k \times 1$ vector.
  - → $\mathbf{c} = [c_0, c_1, \ldots, c_{n-1}]^T$: denotes an $n \times 1$ vector of encoded bits
- We will relate $\mathbf{m}$ and $\mathbf{c}$ through matrix products. Several notes:
  - → The input to the matrix products are binary (i.e., have elements that are either 0 or 1)

Information and Encoded Bits as Vectors

- Consider the input block of k information bits and the output block of n encoded bits of a rate $r = k/n$ channel coding scheme as vectors:
  - → $\mathbf{m} = [m_0, m_1, \ldots, m_{k-1}]^T$: denotes a $k \times 1$ vector of message or information bits
    - ▷ Here, $\mathbf{v}^T$ denotes the transpose of a vector $\mathbf{v}$. In this case, it makes a horizontal $1 \times k$ vector a tall $k \times 1$ vector.
  - → $\mathbf{c} = [c_0, c_1, \ldots, c_{n-1}]^T$: denotes an $n \times 1$ vector of encoded bits
- We will relate $\mathbf{m}$ and $\mathbf{c}$ through matrix products. Several notes:
  - → The input to the matrix products are binary (i.e., have elements that are either 0 or 1)
  - → All the elements of the matrices are also binary

Information and Encoded Bits as Vectors

- Consider the input block of k information bits and the output block of n encoded bits of a rate $r = k/n$ channel coding scheme as vectors:
  - → $\mathbf{m} = [m_0, m_1, \ldots, m_{k-1}]^T$: denotes a $k \times 1$ vector of message or information bits
    - ▷ Here, $\mathbf{v}^T$ denotes the transpose of a vector $\mathbf{v}$. In this case, it makes a horizontal $1 \times k$ vector a tall $k \times 1$ vector.
  - → $\mathbf{c} = [c_0, c_1, \ldots, c_{n-1}]^T$: denotes an $n \times 1$ vector of encoded bits
- We will relate $\mathbf{m}$ and $\mathbf{c}$ through matrix products. Several notes:
  - → The input to the matrix products are binary (i.e., have elements that are either 0 or 1)
  - → All the elements of the matrices are also binary
  - → The output of the matrix product is also turned into binary by taking modulo-two operation

Information and Encoded Bits as Vectors

- Consider the input block of k information bits and the output block of n encoded bits of a rate $r = k/n$ channel coding scheme as vectors:
  - → $\mathbf{m} = [m_0, m_1, \ldots, m_{k-1}]^T$: denotes a $k \times 1$ vector of message or information bits
    - ▷ Here, $\mathbf{v}^T$ denotes the transpose of a vector $\mathbf{v}$. In this case, it makes a horizontal $1 \times k$ vector a tall $k \times 1$ vector.
  - → $\mathbf{c} = [c_0, c_1, \ldots, c_{n-1}]^T$: denotes an $n \times 1$ vector of encoded bits
- We will relate $\mathbf{m}$ and $\mathbf{c}$ through matrix products. Several notes:
  - → The input to the matrix products are binary (i.e., have elements that are either 0 or 1)
  - → All the elements of the matrices are also binary
  - → The output of the matrix product is also turned into binary by taking modulo-two operation

A remark: in the advanced channel coding techniques, the above are not limited to be binary-valued. However, in this class, we will not consider non-binary channel codes

Generator and Parity Check Matrices

- The generator matrix **G** is a matrix of size $n \times k$ used to convert the message bit vector **m** to the encoded vector **c**

$$(\mathbf{c})_{n\times 1} = (\mathbf{G})_{n\times k}\,(\mathbf{m})_{k\times 1}$$

As noted on the prior slide, although an explicity modulo-two operation at the output of this matrix-vector product is not shown, but it is implicit

Generator and Parity Check Matrices

- The generator matrix $\mathbf{G}$ is a matrix of size $n \times k$ used to convert the message bit vector $\mathbf{m}$ to the encoded vector $\mathbf{c}$

$$(\mathbf{c})_{n\times 1} = (\mathbf{G})_{n\times k}\,(\mathbf{m})_{k\times 1}$$

  As noted on the prior slide, although an explicity modulo-two operation at the output of this matrix-vector product is not shown, but it is implicit
- The parity check matrix $\mathbf{H}$ is a matrix of size $u \times n$, such that

$$(\mathbf{H})_{u\times n}\,(\mathbf{c})_{n\times 1} = (\emptyset)_{u\times 1} \qquad (1)$$

  → Again the modulo-two operation is implicit
  → Here $u = n - k$ is the number of parity bits (i.e., the redundancy bits introduced by the encoding process (i.e., the matrix $\mathbf{G}$)
  → $\emptyset$ is an all-zero vector of size $u \times 1$

Encoder

Decoder

Generator and Parity Check Matrices

for the Rate $r = 1/5$ Repetition Code

- What are **G** and **H** matrices for the Rate $r = 1/5$ Repetition Code?

Rate $\frac{1}{5}$ repetition code :

$$k = 1 \;,\quad n = 5 \;,\quad u = n - k = 4$$

$$G_{5\times 1} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}_{5\times 1}$$

$$H_{u\times n} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}_{4\times 5}$$

$$C_{n\times 1} = G_{n\times k} \cdot m_{k\times 1}$$

$$H \cdot C = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}_{4\times 1}$$

Generator and Parity Check Matrices

for the Rate $r = 1/n$ Repetition Code

- For the rate $1/n$ repetition code ($k = 1, u = n - 1$):
  - The parity check matrix is as follows:
    - $\rightarrow$ $\mathbf{H} = \left[1_{u\times 1}, \mathbf{I}_{u\times u}\right]$, i.e., it is a matrix of size $u \times u + 1 = (n-1) \times n$
    - $\rightarrow$ The first column of $\mathbf{H}$ of the repetition code is a (tall or vertical) vector of u ones
    - $\rightarrow$ The Identity matrix $\mathbf{I}_{u\times u}$ populates last $u = n - 1$ columns and all u rows of $\mathbf{H}$
    - $\rightarrow$ The first bit (the message bit) of $\mathbf{c}$ is copied $n - 1$ times in the repetition code. Each of $n - 1$ rows of $\mathbf{H}$ checks that the sum of the message bit and each repeated bit is 0
  - The generator matrix is as follows:
    - $\rightarrow$ $\mathbf{G} = [1, 1, \ldots, 1]^T$ is a tall vector of size $n \times (k = 1)$
    - $\rightarrow$ This ensures that in the product $\mathbf{G}\,\mathbf{m}$, the single message bit m is repeated n times

Practical Channel Coding Techniques

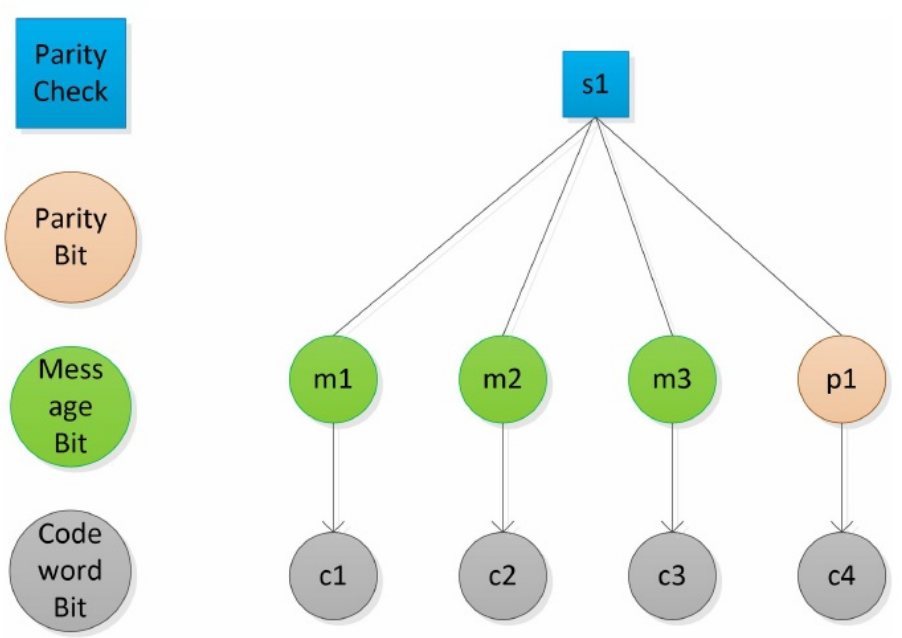
Coding Scheme 2: Single Parity Check (SPC) Code

- Encoding scheme:
  1. Take a block of $n-1 \geq 1$ bits as input.
  2. Add an extra n^{th} bit, called the parity check bit, which is 0 if the input block of n bits has even number of 1's, and it is 1 otherwise
  3. Transmit the resultant n bit long codeword
- Let $\{m_1, m_2, \ldots, m_{n-1}\}$ be the input block. The parity bit added is given as $p_n = \left(\sum_{k=1}^{n-1} m_k\right)_{mod-2}$
- Rate of this code is $r = \frac{n-1}{n}$.
  - $\rightarrow$ Rate r is the ratio of the number of information bits to total number of encoded bits

Practical Channel Coding Techniques

Single Parity Check (SPC) Code

- Notations:
  - $\{m_k\}$: info bits,
  - $\{p_k\}$: parity bits,
  - $\{s_k\}$: check bits,
  - $\{c_k\}$: codeword bits
  - If $\mathbf{c} \stackrel{\text{def}}{=} [c_1, c_2, \ldots, c_n]^T$ is an SPC codeword, $s_1 = \sum_{i=1}^{n} c_i$ has to be zero, where this sum is modulo two.



Practical Channel Coding Techniques

Single Parity Check (SPC) Code

- Decoding scheme:
  1. Take a block of n bits as input.
  2. Compute the modulo-two sum of these bits
  3. If this sum is zero, determine that zero or an even number of bit errors have occurred. If nonzero, determine that one or an odd number of bit errors have occurred
- Rate $(n-1)/n$ SPC code can detect one bit of error, but it cannot correct it

Generator and Parity Check Matrices

for the Rate $r = 4/5$ SPC Code

- What are **G** and **H** matrices for the Rate $r = 4/5$ SPC Code?

$c = G \times m$

$m = \begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \end{bmatrix}$

$G_{5\times 4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$ Imp

such that we bit are these but last bit is parity i.e. sum

$$c = G \times m$$

$$\begin{bmatrix} & G & \end{bmatrix} \begin{bmatrix} m \end{bmatrix}$$

$5 \times 4 \quad 4 \times 1$

$$c = \begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \\ m_0 + m_1 + m_2 + m_3 \end{bmatrix}$$

H Should be Simply Summation (XOR) Here

$H \times C = \phi$

Very

106

$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix}$

$C = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$

$H \times C = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \\ \text{parity} \end{bmatrix}$

$= m_0 + m_1 + m_2 + m_3 + \text{parity}$

Generator and Parity Check Matrices

for the Rate $r = (n-1)/n$ SPC Code

- For the rate $(n-1)/n$ SPC code $(k = n-1, u = 1)$:
  - The parity check matrix is as follows:
    - → $\mathbf{H} = [1, 1, \ldots, 1]$, i.e., it is just a matrix (a horizontal vector) of size $1 \times n$ consisting of all ones
    - → $\mathbf{H}\,\mathbf{c}$ in Eq. 1 denotes the single parity check (whose output is a 1×1 zero-valued bit)
    - → The coded bits $\mathbf{c}$, by definition of the SPC code, satisfy this check, hence $\mathbf{H}\,\mathbf{c} = 0$. In simple terms, this shows that all bits of c add up to 0 (modulo-two)
  - The generator matrix is as follows:
    - → $\mathbf{G} = \begin{bmatrix} \mathbf{I}_{k\times k} \\ 1_{1\times k} \end{bmatrix}$
    - → The Identity matrix $\mathbf{I}_{k\times k}$ populates first $k = n-1$ rows and all k columns of $\mathbf{G}$; the last row is a (horizontal) vector of size $1 \times k$ of all ones. The overall size of $\mathbf{G}$ is, thus, $(k+1) \times k = n \times (n-1)$, as required
    - → This ensures that in the product $\mathbf{G}\,\mathbf{m}$, the first $k = n-1$ bits are directly the message bits, and the last bit (i.e., the parity bit) is the modulo-two sum of all $n-1$ message bits
- Remarks:
  - → Practice the operation of $\mathbf{G}$ and $\mathbf{H}$ for the SPC and the repetition codes, for some example values of n, k and u
  - → Observe a duality between $\mathbf{G}$ and $\mathbf{H}$ matrices of the SPC and the repetition codes ($\mathbf{G}$ of one looks like $\mathbf{H}$ of the other, and vice versa)

Practical Channel Coding Techniques

Coding Scheme 3: Product Codes

- k (number of information bits that get encoded) is a perfect square number $(4, 9, 16, 25, \ldots)$
- $n = \left(\sqrt{k} + 1\right)^2 = k + 1 + 2\sqrt{k}$ is the number of encoded bits
- Encoding Strategy:
  - Place the information bits in $\sqrt{k} \times \sqrt{k}$ array
  - Encode each row with an SPC
  - Encode each column with an SPC

Practical Channel Coding Techniques

Product Codes

- k (number of information bits that get encoded) is a perfect square number $(4, 9, 16, 25, \ldots)$
- $n = \left(\sqrt{k} + 1\right)^2 = k + 1 + 2\sqrt{k}$ is the number of encoded bits
- Encoding Strategy:
  - Place the information bits in $\sqrt{k} \times \sqrt{k}$ array
  - Encode each row with an SPC
  - Encode each column with an SPC

Practical Channel Coding Techniques

An Example $(n = 9, k = 4, u = 5, r = 4/9)$ Product Code

- $p_1 = m_1 + m_2; p_2 = m_3 + m_4$
- $p_3 = m_1 + m_3; p_4 = m_2 + m_4$
- $p_5 = p_1 + p_2 = p_3 + p_4 = m_1 + m_2 + m_3 + m_4$

m1	m2	p1
m3	m4	p2
p3	p4	p5

- A homework assignment: determine the generator matrix **G** of size 9×4 and the parity check matrix of size 5×9 for this product code.

An Iterative Decoding Scheme

for the Product Code received over a BEC channel

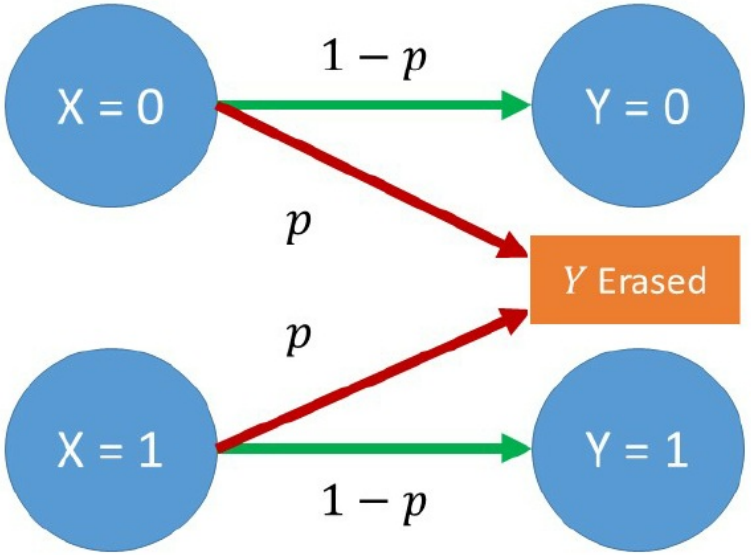
- For understanding how the product codes can be decoded, we will take the following example of $(9, 4)$ product code

m1= 1	m2= 0	p1= 1
m3= 1	m4= 1	p2= 0
p3= 0	p4= 1	p5= 1

An Iterative Decoding Scheme

for the Product Code received over a BEC channel

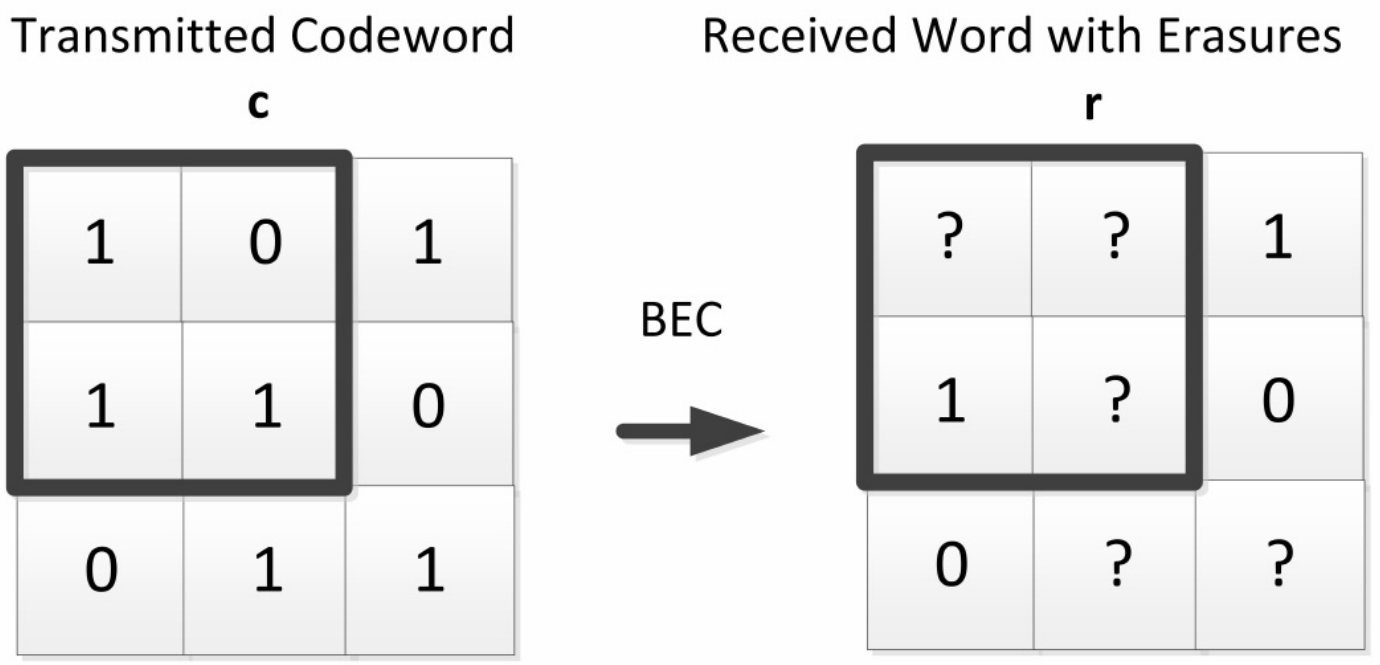
- The model of the BEC channel is as follows:



Iterative Methods

An Example Decoding for the Product Code

- We consider that the transmitted codeword is received over the BEC channel. In this example, as shown below, we take that as many as five bits are erased



Iterative Methods

An Example Decoding for the Product Code

- We now consider a simple iterative method that performs the SPC decoding on each row, followed by each column. This counts as one iteration. The method iterates over multiple iterations until either the transmitted codeword is recovered or the iterations do not make any changes
  - Since we are considering the BEC channel, and there are three bits in each row or column, the SPC is able to correct a row or column if it has no more than one bit erased
  - Work the example on the next slide on your own to improve the understanding

Iterative Methods

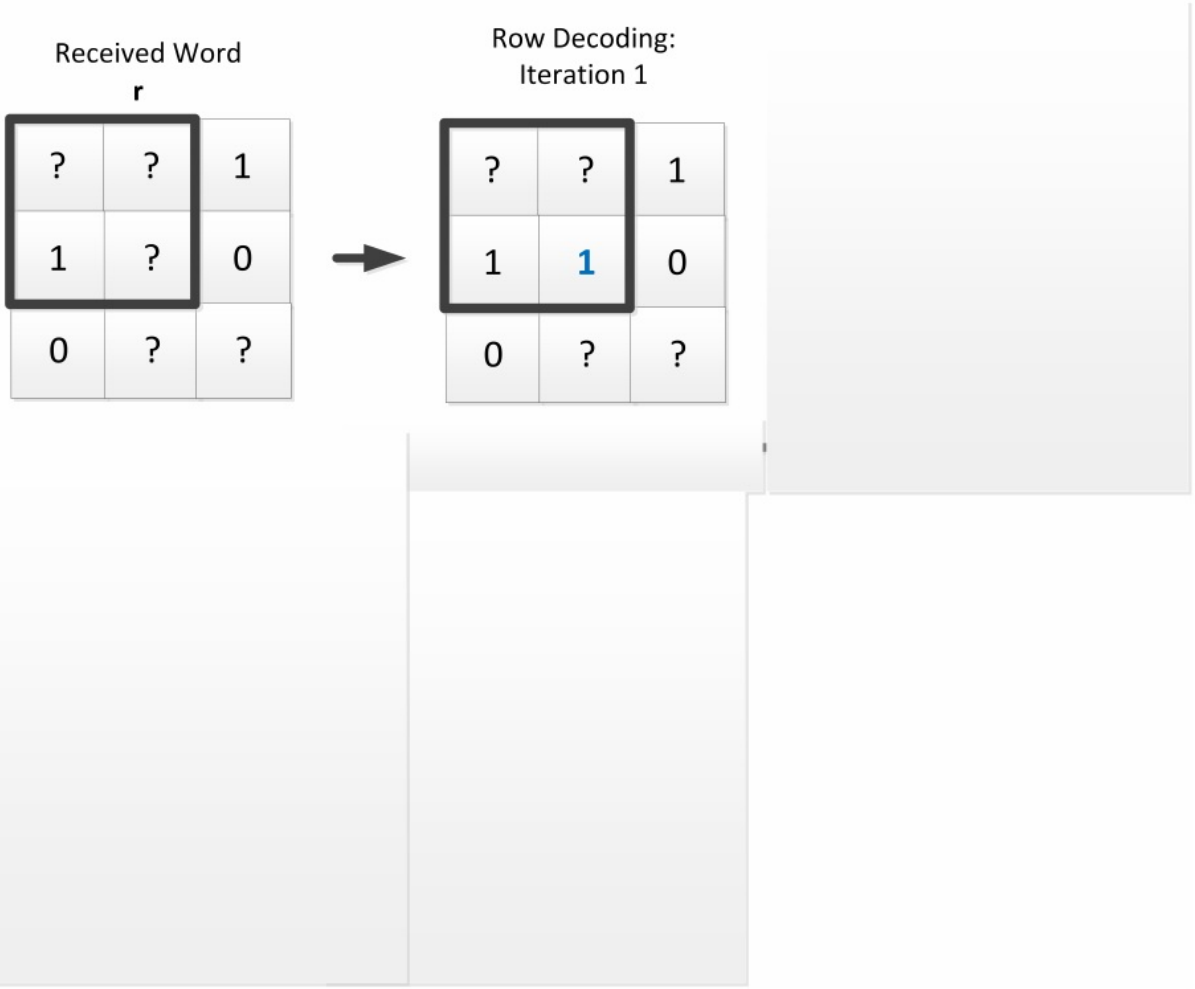
An Example Decoding for the Product Code

Received Word
r

?	?	1
1	?	0
0	?	?

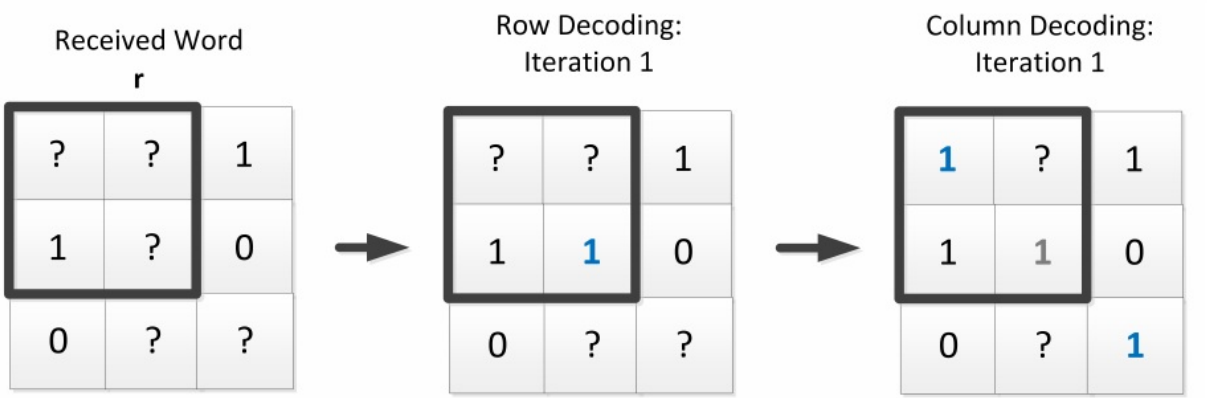
Iterative Methods

An Example Decoding for the Product Code



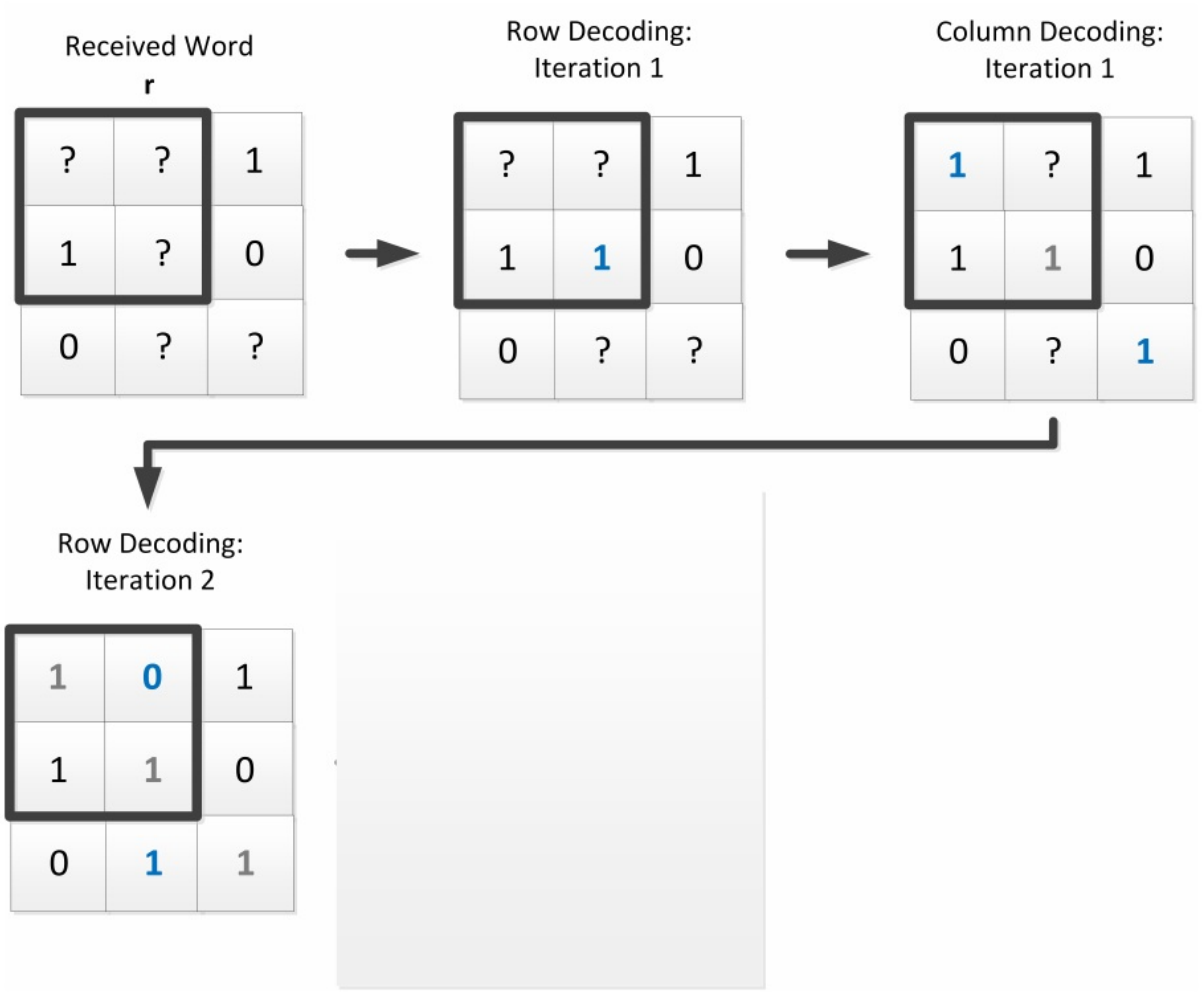
Iterative Methods

An Example Decoding for the Product Code



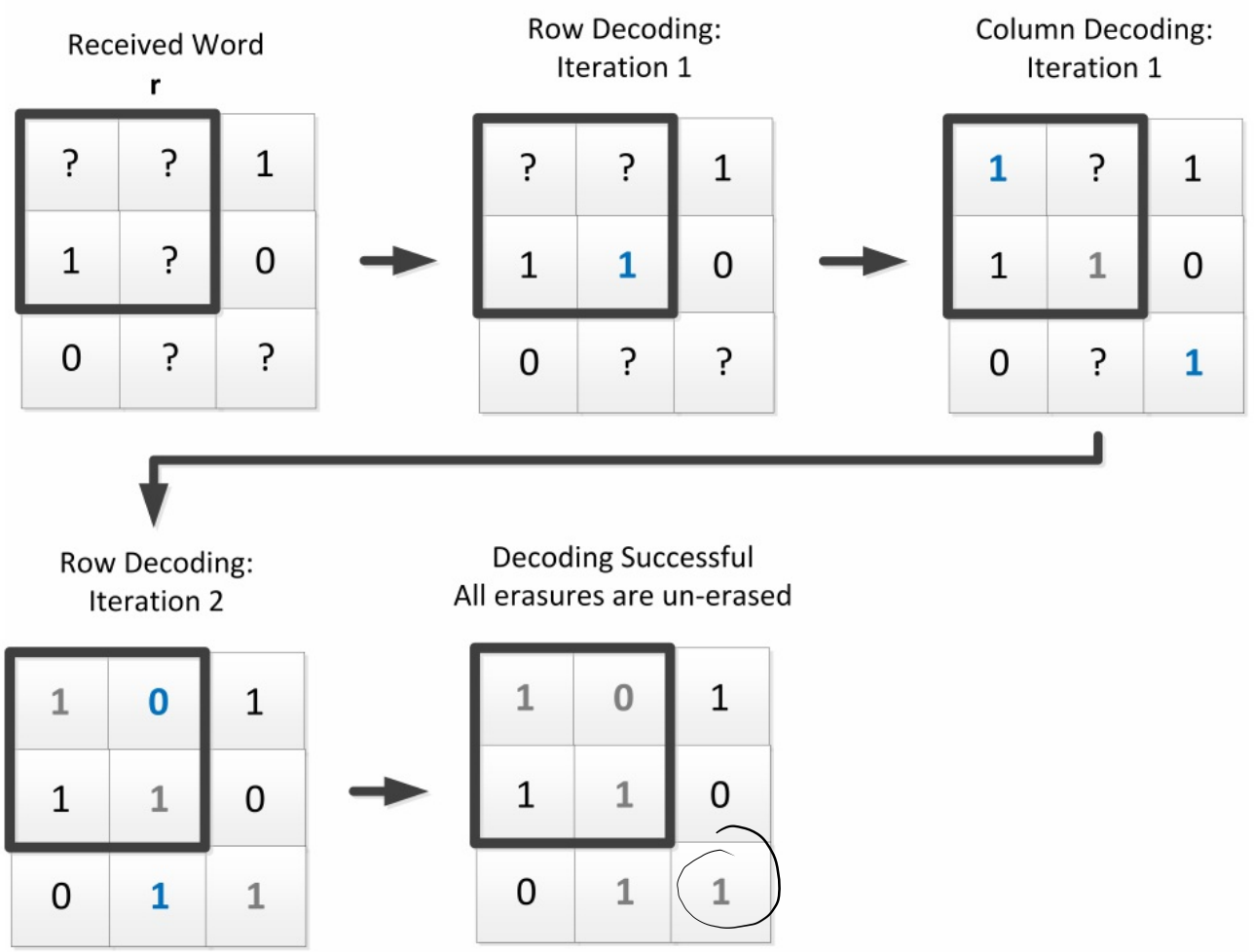
Iterative Methods

An Example Decoding for the Product Code



Iterative Methods

An Example Decoding for the Product Code



Practical Channel Coding Techniques

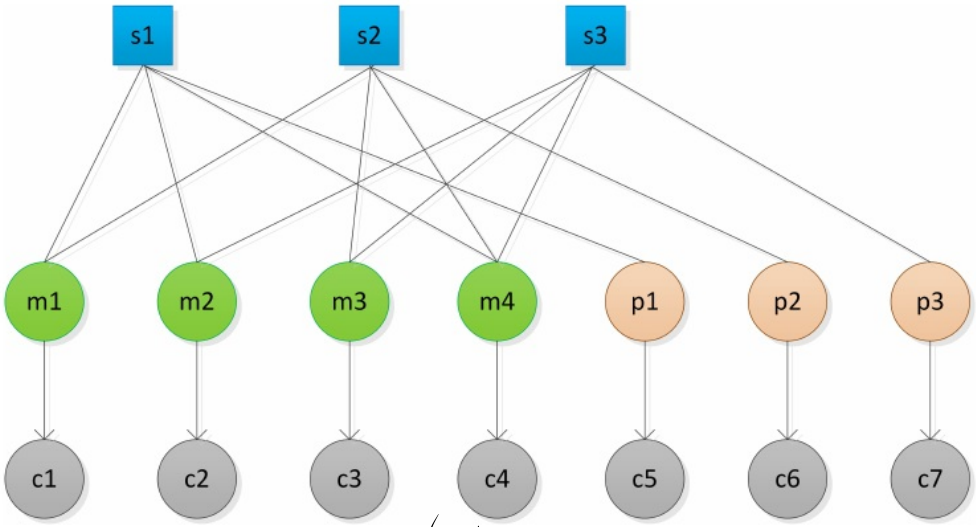
Coding Scheme 4: Rectangular Parity Codes

- Product Codes mentioned above are one of a family of codes known as Rectangular Parity Codes (RPC)
  - → The code defined on the prior two slides can be considered a Square Parity Code
- See Fig. 6-3 of MIT Lecture 6 (posted on Google Classroom) for an example Rectangular Parity Code with three rows and five columns $(n = 14, k = 8, u = 6, r = 4/7)$

Practical Channel Coding Techniques

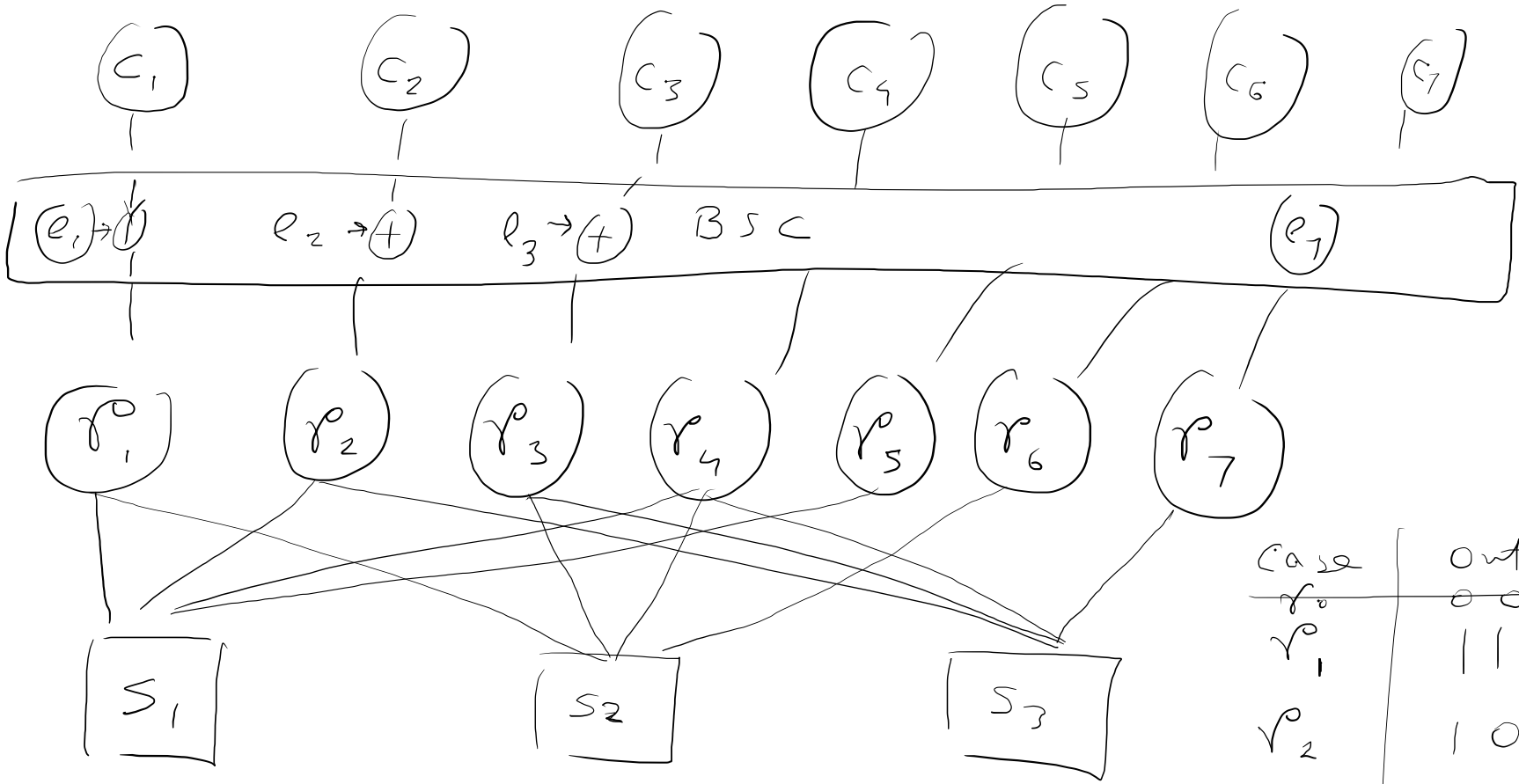
Coding Scheme 5: Hamming Code

- Hamming $(n = 7, k = 4, u = 3, r = 4/7)$ code is essentially three SPC codes:



$$s_1 = m_1 + m_2 + m_4 + p_1$$
$$s_2 = m_1 + m_3 + m_4 + p_2$$
$$s_3 = m_2 + m_3 + m_4 + p_3$$

- A homework/Lab assignment: determine the generator matrix **G** of size 7×4 and the parity check matrix of size 3×7 for the Hamming code. Also answer the following questions:
  - → The columns of **H** together make an interesting pattern. What is that?
  - → Given **H**, the matrix **G** can be determined (and vice versa). How?



Case	Output
r_0	0 0 0
r_1	1 1 0
r_2	1 0 1
r_3	0 1 1 ←
r_4	1 1 1
r_5	1 0 0
r_6	0 1 0
r_7	0 0 1

$$H_{3\times 7}\ C_{7\times 1} = \begin{bmatrix}\bar{0}\end{bmatrix}_{3\times 1}$$

$$\bar{r} = \bar{c} + \bar{e}$$

$$\underset{3\times 7}{\bar{H}} \cdot \underset{7\times 1}{\bar{r}} = H(\bar{c} + \bar{e})$$

$$= H\bar{e} = 0 \text{ 1 1}$$

Practical Channel Coding Techniques

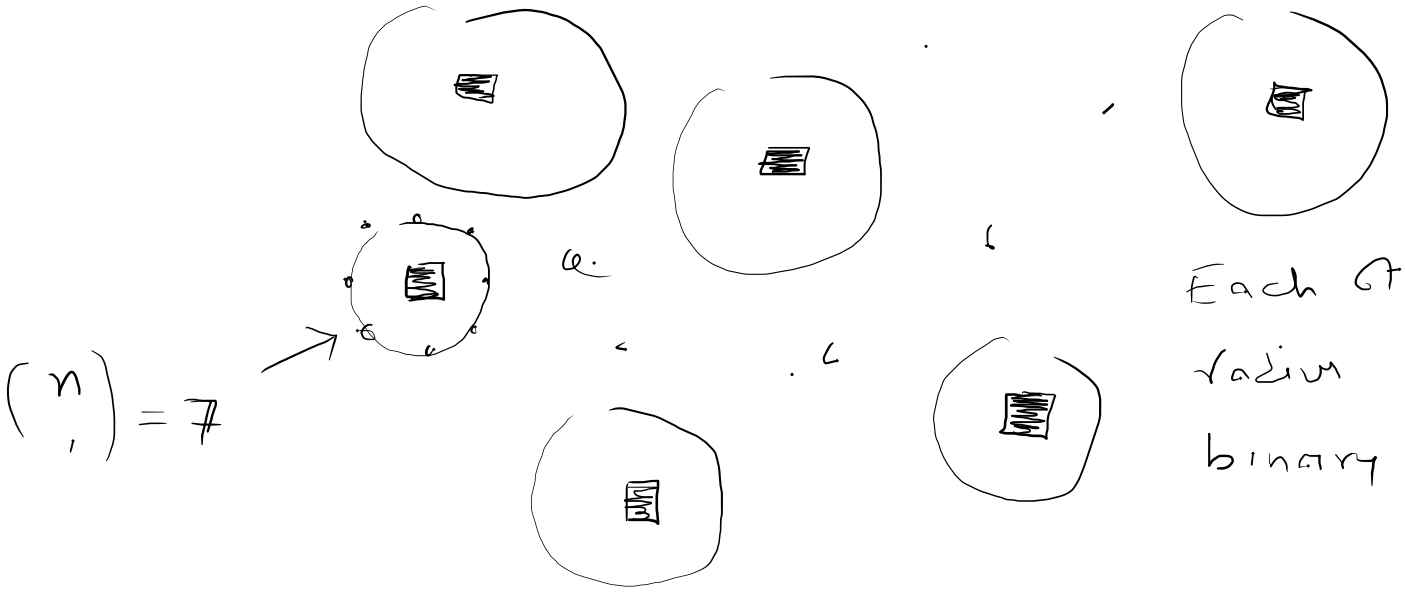
Hamming Code versus Rectangular Parity Code (RPC)

- The parameters n, k, u and r of the Hamming Code matches with that of the RPC code
- Furthermore, $d_H^{\min}$ for both these codes is 3 bits, i.e., $t_c = 1$ bit
- However, Hamming codes are more efficient. Each of the three parity bits does the same amount of "work" (in protecting three message bits). In the RPC codes, some parity bits do more "work" and the others less

Perfect Channel Codes :

Consider a rate $r = 4/7$ channel code where $K = 4$, $n = 7$. Assume That $d_H^{min} = 3$ bits.

$t_c = 1$ bit



$\binom{n}{1} = 7$

Each of The spheres of radius 1 bit has $7+1=8$ binary sequences.

To determine whether a rate

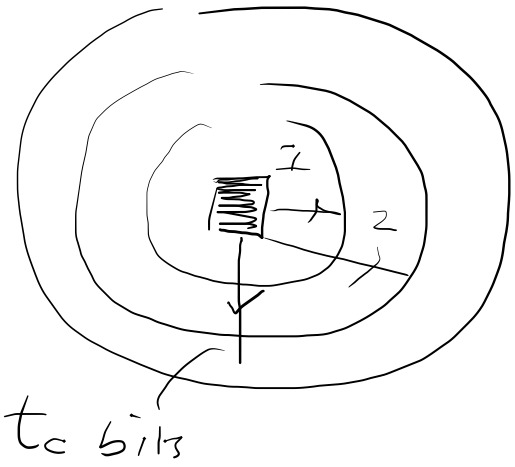
$\rho = \frac{k}{n}$ channel code with k information

bits, n encoded bits and error correction

of t_c bits is a perfect code:

check if the following equality holds

$$2^{k} \cdot V(n, t_c) = 2^{n}$$



$V(n, t_c)$

$$V(n,1) = n+1 = \binom{n}{1} + \binom{n}{0}$$

$$V(n,2) = \binom{n}{2} + \binom{n}{1} + \binom{n}{0}$$

$$\vdots$$

$$V(n,t_c) = \sum_{i=0}^{t_c} \binom{n}{i}$$

Since There are 16 Hamming spheres,

which all have to be non-overlapping,

There is a need of having 16×8 unique

binary sequences, i.e., 128 sequences.

Since $n = 7$, we will have $2^n = 128$ sequences.

Is $r = \frac{4}{7}$ RPC code with

$k = 8$ and $n = 14$ and $t_c = 1$ bit

a perfect code ?

$\Rightarrow$ $2^8 \times 15$ is  a power of two,

cannot be 2^{14}