Information-Theoretic Modeling

Lecture 2: Noisy Channel Coding

Teemu Roos

Department of Computer Science, University of Helsinki

Fall 2009

UNIVERSITY OF HELSINKI

Lecture 2: Noisy Channel Coding

1. What we will *not* talk about (except today)
   - Reliable communication
   - Error correcting codes
   - Repetition codes

1. What we will *not* talk about (except today)
    - Reliable communication
    - Error correcting codes
    - Repetition codes

2. Shannon's theorem
    - Channel capacity
    - Noise Channel Coding Theorem

1. What we will *not* talk about (except today)
   - Reliable communication
   - Error correcting codes
   - Repetition codes

2. Shannon's theorem
   - Channel capacity
   - Noise Channel Coding Theorem

3. Hamming Codes
   - Parity Check Codes
   - Hamming (7,4)

1. What we will *not* talk about (except today)
   - Reliable communication
   - Error correcting codes
   - Repetition codes

2. Shannon's theorem
   - Channel capacity
   - Noise Channel Coding Theorem

3. Hamming Codes
   - Parity Check Codes
   - Hamming (7,4)

Reliable communication

In practice, most media are not perfect — *noisy channels*:

Reliable communication

In practice, most media are not perfect — *noisy channels*:

- Modem line

Reliable communication

In practice, most media are not perfect — *noisy channels*:

- Modem line
- Satellite link

Reliable communication

In practice, most media are not perfect — *noisy channels*:

- Modem line
- Satellite link
- Hard disk

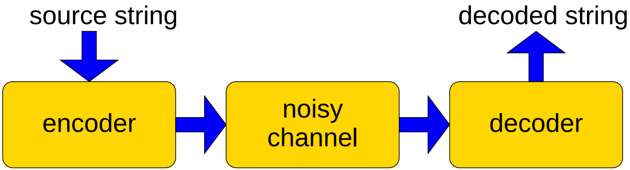
Reliable communication

In practice, most media are not perfect — *noisy channels*:

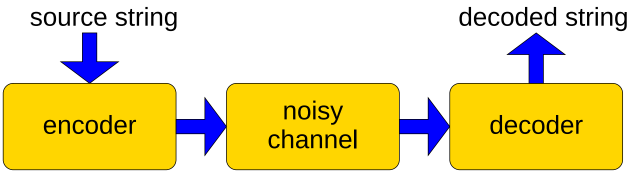
- Modem line
- Satellite link
- Hard disk

Can we recover the original message (without errors) from a noisy code string?

Error correcting codes



Error correcting codes



We want to minimize two things:

1. Length of the code string.
2. Probability of error.

Repetition codes

A simple idea: Just repeat the original string many times.

Repetition codes

A simple idea: Just repeat the original string many times.

Get it? Get it? Get it? Get it? Get it? Get it? Get it? G

Repetition codes

A simple idea: Just repeat the original string many times.

`T R A N S M I S S I O N`

Repetition codes

A simple idea: Just repeat the original string many times.

```
T R A N S M I S S I O N

TTTRRRAAANNNSSSMMMIIISSSSSSIIIOOONNN
```

Repetition codes

A simple idea: Just repeat the original string many times.

`T R A N S M I S S I O N`

`TTTRRRAAANNNSSSMMMIIISSSSSSIIIOOONNN`

`TTTHRRAAANNBSSSMMMIIISSSSWSPILOOONNG`

Repetition codes

A simple idea: Just repeat the original string many times.

T R A N S M I S S I O N

TTTRRRAAANNNSSSMMMIIISSSSSSIIIOOONNN

TTTHRRAAANNBSSSMMMIIISSSSWSPILOOONNG

T R A N S M I S S ? O N

Repetition codes

A simple idea: Just repeat the original string many times.

```
 T  R  A  N  S  M  I  S  S  I  O  N

TTTRRRAAANNNSSSMMMIIISSSSSSIIIOOONNN

TTTHRRAAANNBSSSMMMIIISSSSWSPILOOONNG

 T  R  A  N  S  M  I  S  S  ?  O  N
```

Transmission rate reduced to 1 : 3.

Repetition codes

A simple idea: Just repeat the original string many times.

```
T  R  A  N  S  M  I  S  S  I  O  N
TTTRRRAAANNNSSSMMMIIISSSSSSIIIOOONNN
TTTHRRAAANNBSSSMMMIIISSSSWSPILOOONNG
T  R  A  N  S  M  I  S  S  ?  O  N
```

Transmission rate reduced to 1 : 3.

If errors independent and symmetric, probability of error reduced to $3(1-p)p^2 + p^3 \approx 3p^2$, where p is the error rate of the channel.

1. What we will *not* talk about (except today)
   - Reliable communication
   - Error correcting codes
   - Repetition codes
2. Shannon's theorem
   - Channel capacity
   - Noise Channel Coding Theorem
3. Hamming Codes
   - Parity Check Codes
   - Hamming (7,4)

Channel Capacity

- Binary symmetric channel (BSC), error rate p:

$$\Pr[\hat{x} = 1 \mid x = 0] = \Pr[\hat{x} = 0 \mid x = 1] = p.$$

Channel Capacity

- Binary symmetric channel (BSC), error rate p:

$$\Pr[\hat{x} = 1 \mid x = 0] = \Pr[\hat{x} = 0 \mid x = 1] = p.$$

- *Channel capacity*

$$C(p) = 1 - H(p) = 1 - \left[p \log_2 \frac{1}{p} + (1-p) \log_2 \frac{1}{1-p} \right] .$$

Channel Capacity

- Binary symmetric channel (BSC), error rate p:

$$\Pr[\hat{x} = 1 \mid x = 0] = \Pr[\hat{x} = 0 \mid x = 1] = p.$$

- *Channel capacity*

$$C(p) = 1 - H(p) = 1 - \left[p \log_2 \frac{1}{p} + (1-p) \log_2 \frac{1}{1-p} \right] .$$

- For instance, $C(0.1) \approx 0.53$. Ratio about $1:2$.

Channel Capacity

- Binary symmetric channel (BSC), error rate p:

$$\Pr[\hat{x} = 1 \mid x = 0] = \Pr[\hat{x} = 0 \mid x = 1] = p.$$

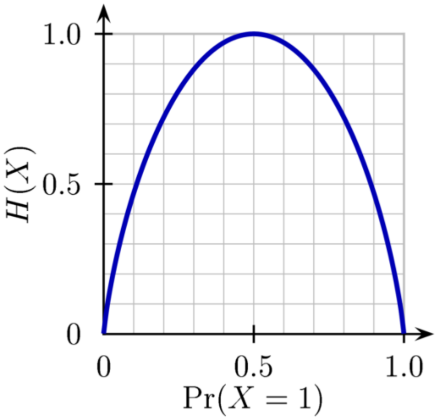
- *Channel capacity*

$$C(p) = 1 - H(p) = 1 - \left[p \log_2 \frac{1}{p} + (1-p) \log_2 \frac{1}{1-p} \right] .$$

- For instance, $C(0.1) \approx 0.53$. Ratio about $1:2$.

The quantity $H(p)$ is the binary *entropy*. We will get to know it very well during this course.

Channel Capacity



Noisy Channel Coding Theorem

Noisy Channel Coding Theorem

For rates less than channel capacity, the error probability can be made arbitrarily small (but not zero), and *vice versa*.

So what?

Noisy Channel Coding Theorem

Noisy Channel Coding Theorem

For rates less than channel capacity, the error probability can be made arbitrarily small (but not zero), and *vice versa*.

Assume you want to transmit data with probability of error 10^{-15} over a BSC, $p = 0.1$. Using a repetition code, we need to make the message **63** times as long as the source string.
(Exercise: Check the math. Hint: binomial distribution.)

Noisy Channel Coding Theorem

Noisy Channel Coding Theorem

For rates less than channel capacity, the error probability can be made arbitrarily small (but not zero), and *vice versa*.

Assume you want to transmit data with probability of error 10^{-15} over a BSC, $p = 0.1$. Using a repetition code, we need to make the message **63** times as long as the source string.

Shannon's result says twice as long is enough.

Noisy Channel Coding Theorem

Noisy Channel Coding Theorem

For rates less than channel capacity, the error probability can be made arbitrarily small (but not zero), and *vice versa*.

Assume you want to transmit data with probability of error 10^{-15} over a BSC, $p = 0.1$. Using a repetition code, we need to make the message **63** times as long as the source string.

If you want probability of error 10^{-100}, Shannon's result still says that twice is enough!

Noisy Channel Coding Theorem

Noisy Channel Coding Theorem

For rates less than channel capacity, the error probability can be made arbitrarily small (but not zero), and *vice versa*.

Assume you want to transmit data with probability of error 10^{-15} over a BSC, $p = 0.1$. Using a repetition code, we need to make the message **63** times as long as the source string.

If you want probability of error 10^{-100}, Shannon's result still says that twice is enough!

1. What we will *not* talk about (except today)
   - Reliable communication
   - Error correcting codes
   - Repetition codes
2. Shannon's theorem
   - Channel capacity
   - Noise Channel Coding Theorem
3. Hamming Codes
   - Parity Check Codes
   - Hamming (7,4)

Hamming Codes

Richard W. Hamming (11.2.1915–7.1.1998)

Parity Check Codes

One way to detect and correct errors is to add *parity checks* to the codewords:

Parity Check Codes

One way to detect and correct errors is to add *parity checks* to the codewords:

- If we add a parity check bit at the end of each codeword we can detect one (but not more) error per codeword.

Parity Check Codes

One way to detect and correct errors is to add *parity checks* to the codewords:

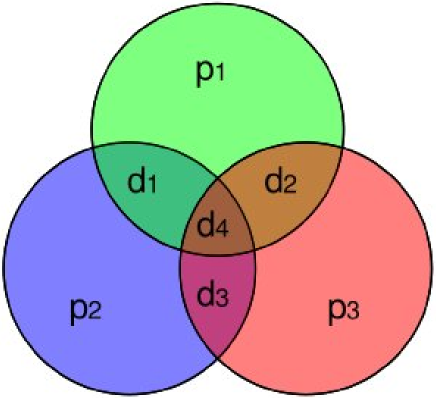
- If we add a parity check bit at the end of each codeword we can detect one (but not more) error per codeword.
- By clever use of more than one parity bits, we can actually identify where the error occurred and thus also *correct errors*.

Parity Check Codes

One way to detect and correct errors is to add *parity checks* to the codewords:

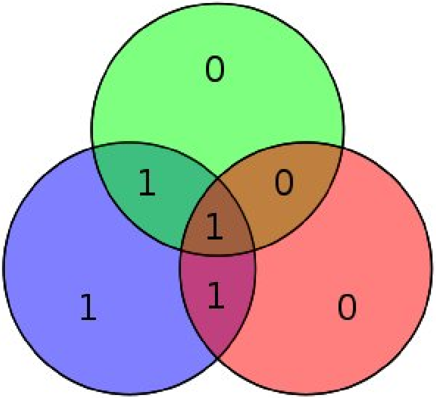
- If we add a parity check bit at the end of each codeword we can detect one (but not more) error per codeword.
- By clever use of more than one parity bits, we can actually identify where the error occurred and thus also *correct errors*.
- Designing ways to add as few parity bits as possible to correct and detect errors is a *really* hard problem.

Hamming (7,4)



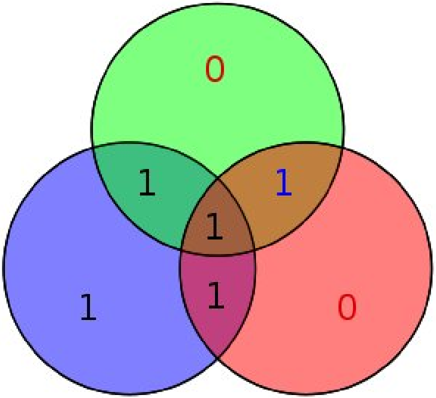
4 data bits (d_1, d_2, d_3, d_4), 3 parity bits (p_1, p_2, p_3)

Hamming (7,4)



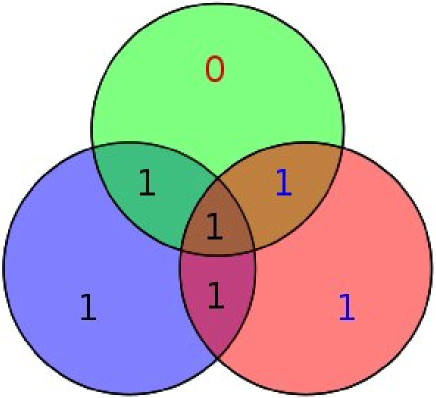
source string 1011, parity bits 010

Hamming (7,4)



error in data bit d_2 $(0 \mapsto 1)$ is identified and corrected

Hamming (7,4)



two errors can be detected but not corrected

Advanced Error Correcting Codes

The Hamming (7,4) code is an example of a code that can detect and correct errors at rate greater than $1:2$.

Advanced Error Correcting Codes

The Hamming (7,4) code is an example of a code that can detect and correct errors at rate greater than 1 : 2.

More complex Hamming codes, like Hamming (8,4), Hamming (11,7), etc. can correct and/or detect more errors.

Advanced Error Correcting Codes

The Hamming (7,4) code is an example of a code that can detect and correct errors at rate greater than 1 : 2.

More complex Hamming codes, like Hamming (8,4), Hamming (11,7), etc. can correct and/or detect more errors.

The present state-of-the-art is based on so called *low-density parity-check* (LDPC) codes, which likewise include a number of parity check bits.

Advanced Error Correcting Codes

The Hamming (7,4) code is an example of a code that can detect and correct errors at rate greater than 1 : 2.

More complex Hamming codes, like Hamming (8,4), Hamming (11,7), etc. can correct and/or detect more errors.

The present state-of-the-art is based on so called *low-density parity-check* (LDPC) codes, which likewise include a number of parity check bits.

Massive research effort: At ISIT-09 conference, 12 sessions (4 talks in each) about LDPC codes.

Next Lecture

We will not make any errors, so focus on noiseless channels.

Next Tuesday:

- crash course on math (calculus, probability).

Next Lecture

We will not make any errors, so focus on noiseless channels.

Next Tuesday:

- crash course on math (calculus, probability).

Next Friday:

- entropy (finally!) and AEP,

Next Lecture

We will not make any errors, so focus on noiseless channels.

Next Tuesday:

- crash course on math (calculus, probability).

Next Friday:

- entropy (finally!) and AEP,
- first exercises (questions anyone?)