CSE/EE 461 – Lecture 3
Error Detection and Correction

Janet Davis
jlnd@cs.washington.edu
January 9, 2004

Last Time

• A brief introduction to layering
• Sending a message across a wire
  – Coding: turning bits into a signal
  – Framing: where do messages start and end?

Muddiest point

• Too fast!
• Layering
  – we’ll return to this later
• 4B/5B
• XOR
• Efficiency
  – We’ll do some examples
• Others
  – Read, talk to people, come to office hours!

NRZI

0 = stay same
1 = transition

\[
\begin{array}{cccccccc}
| & | & | & | & | & | & | & | \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\
\end{array}
\]
4B/5B

Original:
0 0 0 0 1 1 0 1

4B/5B encoded:
1 1 1 1 0 1 1 0 1 1

NRZI:

Efficiency:

XOR

Original:
0 0 0 0 1 1 0 1

Pseudo-random string (always use same one!):
1 0 1 1 0 1 1 0

XOR encoded:

This Lecture

- Error detection and correction
- Focus: How do we detect and correct errors in messages that are garbled during transmission?

Errors and Redundancy

- Noise can flip some of the bits we receive
  - Maybe just 1 bit:
  - Often several in a row:
- Basic approach: add redundant data
  - Error detection
  - Error correction
A Simple Error Detection Scheme

- Just send 2 copies of everything.
- Differences imply errors.
- Issues with this?
  -

Block codes

- K bits in, N bits out is a (N,K) code
- Simple, memoryless mapping

The Hamming Distance

- Errors must not turn one valid codeword into another valid codeword, or we cannot detect/correct them.
- Hamming distance of a code is the smallest number of bit differences that turn any one codeword into another.
  - e.g., code 000 for 0, 111 for 1
  - Hamming distance is

Hamming Distance (2)

- For code with distance d+1:
  - errors of d bits can be detected, e.g., 001, 010, 110, 101, 011

- For code with distance 2d+1:
  - Errors of d bits can be corrected, e.g., 001 \rightarrow 000
Parity

- Start with \( n \) bits and add one more so that the total number of 1s is even (even parity)
  - e.g. 0110010 \( \rightarrow \) 01100101
  - Easy to compute as XOR of all input bits

- Hamming distance =
  - Detect:
  - Correct:

2D Parity

- Add parity row/column to array of bits

\[
\begin{array}{cccccc}
0 & 1 & 0 & 0 & 1 & 1 \\
1 & 1 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 0 \\
1 & 1 & 1 & 0 & 1 & 1 \\
\end{array}
\]

Hamming distance =
- Detect:
- Correct:
- Overhead:

Detection vs. Correction

- Two strategies to correct errors:
  - Detect and retransmit, or Automatic Repeat reQuest (ARQ)
  - Error correcting codes, or Forward Error Correction (FEC)
- Why not always just use FEC?
Checksums

- Basic Idea: Add up the data and send the sum
- Algorithm:
  - Break data into 16-bit blocks
  - Compute running sum (ones complement) of all blocks
  - Compute ones complement of sum

Ones complement

- Definition
  - Flip all bits to make number negative
    +5 =
    -5 =
- Consequence
  - Adding requires carryout to be added to result
    \[ 1010 + 1100 = 10110 \]

Checksums

- Overhead:

- Hamming distance =
- Detect:

- Correct:

CRCs (Cyclic Redundancy Check)

- Stronger protection than checksums
  - Used widely in practice, e.g., Ethernet CRC-32
  - Implemented in hardware (XORs and shifts)
- Based on mathematics of finite fields
  - "Numbers" correspond to polynomials, use modulo arithmetic
  - e.g., interpret 10011010 as \[ x^7 + x^4 + x^3 + x^1 \]
**CRCs (Cyclic Redundancy Check)**

Algorithm:
- Given n bits of data,
- use polynomial long division to generate a k bit check sequence
- that gives a combined n + k bits that are divisible by a chosen divisor C(x)

**How is C(x) Chosen?**

- Mathematical properties:
  - All 1-bit errors if non-zero $x^k$ and $x^0$ terms
  - All 2-bit errors if C(x) has a factor with at least three terms
  - Any odd number of errors if C(x) has $(x + 1)$ as a factor
  - Any burst error < k bits
- There are several standard ones!
  - e.g., Ethernet CRC-32:
    10000100110000010001110110110111

**Retransmissions vs. FEC**

- The better option depends on the kind of errors and the cost of recovery
- Example: Message with 1000 bits, Prob(bit error) 0.001
  - Case 1: random errors
  - Case 2: bursts of 1000 errors
  - Case 3: real-time application (e.g., teleconference)

**Key Concepts**

- Redundant bits are added to messages to protect against transmission errors.
- The Hamming distance tells us how much error can safely be tolerated.
- Retransmission is complementary to error detection & correction.