Digital Data Communication Techniques

ITS323: Introduction to Data Communications

Sirindhorn International Institute of Technology
Thammasat University

Prepared by Steven Gordon on 7 June 2011
ITS323Y11S1L06, Steve/Courses/ITS323/Lectures/digitaldata.tex, r1798

Contents

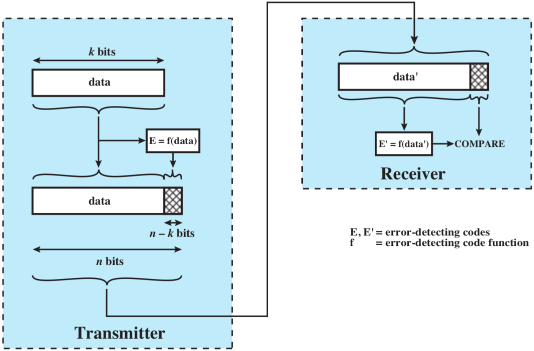
Bit Errors

- In digital transmission systems errors occur when a bit is altered between transmission and reception
- Single-bit errors
  - Only one bit altered, surrounding bits not affected
  - Caused by random noise
- Error burst
  - A group of bits near each other are affected (in error)
  - Caused by impulse noise or fading
  - Effects of burst errors are greater at higher data rates
- Require methods to detect errors, and correct where possible

Contents

Error Detection

- Transmitter adds extra information to transmitted data, i.e. an error– detecting code
- Receiver recalculates the error–detecting code from received data, and compares to received error-detecting code
- If the same, good. If not, then error (in data or code). Still a chance that an error is not detected



Error Detection with Parity Check

- Odd-parity check: append parity bit to block of data; resulting set of bits has odd number of ones
- Receiver detects an error if receiver bits has unexpected number of ones (transmitter and receiver both know parity scheme being used)

Parity Check Example

Assume character S is to be sent using odd-parity check. What is transmitted? What happens if the last bit is corrupted? What about the last two bits?

Error Detection with Cyclic Redundancy Check

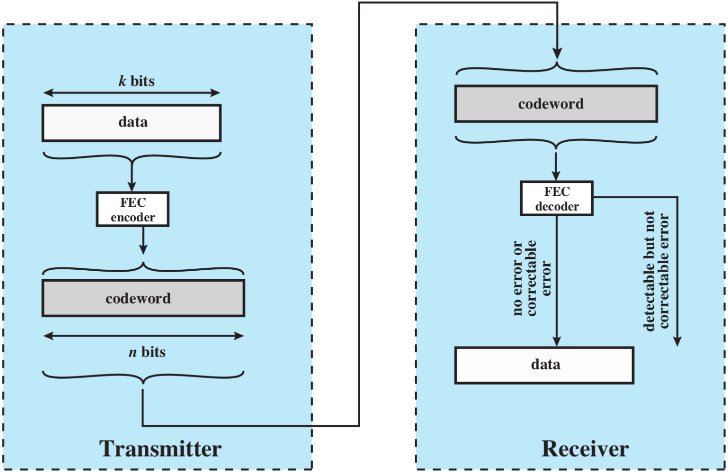
- Parity checks are not good when multiple bit errors occur
- CRC is a powerful, commonly used error detection scheme
- Approach:
  - k bits of data to send
  - Constant divisor known by transmitter/receiver, $n - k + 1$ bits
  - Append $n - k$ bits to data such that no remainder when divided by divisor
  - Transmit n bits
  - Receiver divides received n bits by divisor; if remainder, error detected
- Length and value of divisor is important for error detection capabilities (e.g. chance that one or more errors go undetected)
- CRC used in: Ethernet, HDLC, SATA, CDMA, PNG images, SD cards, ...

Contents

Error Correction

- What to do when error detected at receiver?
- Ask transmitted to send again, i.e. retransmit
  - Covered in Data Link Control Protocols lecture
  - Can be inadequate if link has high delay or many errors, e.g. wireless/satellite links
- Forward Error Correction: sender sends a codeword (instead of data); codeword chosen such that if error detected, receiver can correct the error without retransmission
- Depending on encoding scheme and pattern of errors, receiver may: detect and correct errors; detect, but not correct errors; not detect errors

Error Correction



FEC with Hamming Distance

Hamming Distance

- Number of bits of two n-bit sequences that differ
- $v_1 = 011011$, $v_2 = 110001$: $d(v1, v2) = 3$

Example FEC Encoder

- 2-bits of data mapped to 5-bit codeword ($k = 2$, $n = 5$)

Data	*Codeword*
00	00000
01	00111
10	11001
11	11110

- If received codeword invalid, assume valid codeword that is unique minimum Hamming distance from received codeword was transmitted

Error Correction Example 1

Data to send: 01; no transmission error

Error Correction Example 2

Data to send: 01; 3rd bit transmitted is in error

Error Correction Example 3

Data to send: 01; 1st and 4th bit transmitted in error

Performance of Error Detection/Correction

- Aim to detect/correct as many errors as possible
- But error detection/correction require extra bits to be sent
- k bits of useful data; n bits transmitted; efficiency $\frac{k}{n}$
- Tradeoff: for a given amount of data, k bits
  - Increase n, more errors detected/corrected (GOOD)
  - Increase n, lower efficiency of transmission (BAD)