General Errors, Uncountability, Self Reference, Counting 5

Computer Science Mentors 70

October 10 to October 14, 2016

1 General Errors (Berlekamp and Welch)

1.1 Introduction

Now instead of losing packets, we know that k packets are corrupted. Furthermore, we do not know which k packets are changed. Instead of sending k additional packets, we will send an additional $2k$.

$$\boxed{3}\ \boxed{1}\ \boxed{5}\ \boxed{0} \longrightarrow \boxed{4}\ \boxed{1}\ \boxed{5}\ \boxed{1}$$

Solomon-Reed Codes

1. Identical to erasure errors: Alice creates $n-1$ degree polynomial $P(x)$.

$$P(x) = p_0 + p_1 x + \ldots + p_{n-1} x^n$$

2. Alice sends $P(1), \ldots, P(n+2k)$

3. Bob receives $R(1), \ldots, R(n+2k)$

For how many points does $R(x) = P(x)$?

True or false: $P(x)$ is the unique degree $n-1$ polynomial that goes through at least $n+k$ of the received points.

Write the matrix view of encoding the points $P(1), \ldots, P(n+2k)$

Berlekamp Welch

How do we find the original polynomial $P(x)$?

Suppose that $m_1, \ldots, m_k$ are the corrupted packets. Let $E(x) = (x - m_1) \ldots (x - m_k)$

Then $P(i) * E(i) = r_i * E(i)$ for any i greater than 1 and less than $n + 2k$. Why?

Let $Q(i) = P(i)E(i)$ So we have $Q(i) = P(i)E(i) = r_i * E(i)$ where $1 \leq i \leq 2k + n$ What degree is $Q(i)$?

How many coefficients do we need to describe $Q(i)$?

What degree is $P(i)$?

How many unknown coefficients do we need to describe $E(i)$?

We can write $Q(i) = r_i E(i)$ for every$i$ that is $1 \leq i \leq 2k + n$.

How many equations do we have? How many unknowns?

Once we have the above described equations, how do we determine what $P(i)$ is?

1.2 Questions

1. (a) Alice sends Bob a message of length 3 on the Galois Field of 5 (modular space of mod 5). Bob receives the following message: (3, 2, 1, 1, 1). Assuming that Alice is sending messages using the proper general error message sending scheme, set up the linear equations that, when solved, give you the $Q(x)$ and $E(x)$ needed to find the original $P(x)$.

   (b) What is the encoded message that Alice actually sent? What was the original message? Which packet(s) were corrupted?

2 Uncountability

2.1 Introduction

1. (a.) What does it mean for a set to be countably infinite?

(b.) Does $\mathbb{N}$ and $\mathbb{Z}^+$ have the same cardinality? Does adding one element change cardinality?

(c.) Cantor-Bernstein Theorem: Suppose there is an injective function from set A to set B and there is an injective function from set B to set A. Then there is a bijection between A and B. Use this theorem to prove that Q is countable

2.2 Questions

1. Are these sets countably infinite/ uncountable infinite/ finite? If finite, what is the order of the set?
   (a) Finite bit strings of length n.
   (b) All finite bit strings of length 1 to n.
   (c) All finite bit strings
   (d) All infinite bit strings
   (e) All finite or infinite bit strings.
2. Find a bijection between N and the set of all integers congruent to $1 \mod n$, for a fixed n.
3. **True/False**
   (a) Every infinite subset of a countable set is countable
   (b) If A and B are both countable, then $A \times B$ is countable
   (c) Every infinite set that contains an uncountable set is uncountable.

3 Self Reference

3.1 Introduction

The Halting Problem: Does a given program ever halt when executed on a given input?

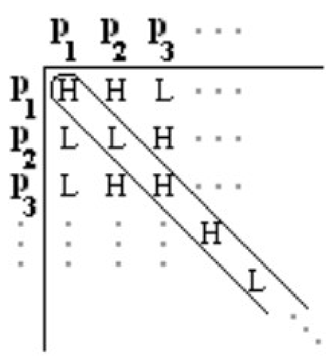
$$\texttt{TestHalt(P,x)} = \begin{cases} \text{“yes"}, & \text{if program } P \text{ halts on input } x \\ \text{“no"}, & \text{if program } P \text{ loops on input } x \end{cases}$$

How do we prove that `TestHalt` doesnt exist? Lets assume that it does, and hope we reach a contradiction.
Define another program:

```
Turing(P)
  if TestHalt(P,P) = “yes” then loop forever
  else halt
```

What happens when we call `Turing(Turing)`?

How is this just a reformulation of proof by diagonalization?



Therefore the Halting Problem is unsolvable. We can use this to prove that other problems are also unsolvable. Say we are asked if program M is solvable. To prove it is not, we just need to prove the following claim: If we can compute program M, then we could also compute the halting problem.
This would then prove that M can not exist, since the halting problem is not computable. This amounts to proof by contradiction.

3.2 Questions

1. Say that we have a program M that decides whether any input program halts as long as it prints out the string ABC as the first operation that it carries out. Can such a program exist?

4 Intro to Counting

4.1 Introduction

Rules of counting:

1. If your event is composed of different independent events then you can multiply together the probabilities of the independent events.
2. If order does not matter then count with order and then divide by the number of orderings/sorted objects

4.2 When Order Matters

1. (a) You have 15 chairs in a room and there are 9 people. How many different ways can everyone sit down?

   (b) How many ways are there to fill 9 of the 15 chairs? (We dont care who sits in them)

2. **Identical Digits** The numbers 1447, 1005, and 1231 have something in common. Each of them is a four digit number that begins with 1 and has two identical digits. How many numbers like this are there?

4.3 More Practice

1. At Starbucks, you can choose either a Tall, a Grande, or a Venti drink. Further, you can choose whether you want an extra shot of espresso or not. Furthermore, you can choose whether you want a Latte, a Cappuccino, an Americano, or a Frappuccino.

How many different drink combinations can you order?

2. We grab a deck of cards and its poker time. Remember, in poker, order doesnt matter.
   (a) How many ways can we have a hand with exactly one pair? This means a hand with ranks (a, a, b, c, d)
   (b) How many ways can we have a hand with four of a kind? This means a hand with ranks (a, a, a, a, b)
   (c) How many ways can we have a straight ? A straight is 5 consecutive cards, that dont all necessarily have the same suit. straight can be (2, 3, 4, 5, 6); (3, 4, 5, 6, 7); ; (10, J, Q, K, A) can start from 2 - 10, which is 9 possibilities each number in hand has 4 possibilities (suits)
   (d) How many ways can we have a hand of all of the same suit?
   (e) How many ways can we have a straight flush? This means we have a consecutive-rank hand of the same suit. For examples, (2, 3, 4, 5, 6), all of spades is a straight flush, while (2, 3, 5, 7, 8) of all spades is NOT, as the ranks are not consecutive.
3. How many solutions does $x+y+z = 10$ have, if all variables must be positive integers?
4. How many ways are there to arrange the letters of the word SUPERMAN
   (a) On a straight line?
   (b) On a straight line, such that SUPER occurs as a substring?
   (c) On a straight line, such that SUPER occurs as a subsequence (S U P E R appear in that order, but not necessarily next to each other)?
   (d) On a circle?
   (e) On a circle, such that SUPER occurs as a substring?
   (f) On a circle, such that SUPER occurs as a subsequence (S U P E R appear in that order, but not necessarily next to each other)?