CS 70 Discrete Mathematics and Probability Theory
Spring 2016 Rao and Walrand Discussion 6B Solution

1. **GCD of Polynomials**
Let $A(x)$ and $B(x)$ be polynomials (with coefficients in $\mathbb{R}$ or $GF(m)$). We say that $\gcd(A(x),B(x)) = D(x)$ if $D(x)$ divides $A(x)$ and $B(x)$, and if every polynomial $C(x)$ that divides both $A(x)$ and $B(x)$ also divides $D(x)$. For example, $\gcd((x-1)(x+1),(x-1)(x+2)) = x-1$. Incidentally, $\gcd(A(x),B(x))$ is the highest degree polynomial that divides both $A(x)$ and $B(x)$.

   (a) Write a recursive program to compute $gcd(A(x),B(x))$. You may assume you already have a subroutine for dividing two polynomials.

   Answer: Specifically, we wish to find a gcd of two polynomials $A(x)$ and $B(x)$, assuming that that $\deg A(x) \geq \deg B(x) > 0$. Here, $\deg A(x)$ denotes the degree of $A(x)$.
   We can find two polynomials $Q_0(x)$ and $R_0(x)$ by polynomial long division (see lecture note 8) which satisfy

$$A(x) = B(x)Q_0(x) + R_0(x), \qquad 0 \leq \deg R_0(x) < \deg B(x)$$

   Notice that a polynomial $C(x)$ divides $A(x)$ and $B(x)$ iff it divides $B(x)$ and $R_0(x)$.
   [Proof: $C(x)$ divides $A(x), B(x)$, there $\exists S(x)$ and $S'(x)$ s.t. $A(x) = C(x)S(x)$ and $B(x) = C(x)S'(x)$, so $R_0(x) = A(x) - B(x)Q_0(x) = C(x)(S(x) - S'(x)Q_0(x))$, therefore $C(x)$ divides $R_0(x)$ or $R_0(x) = 0$.]
   We deduce that

$$gcd(A(x),B(x)) = gcd(B(x),R(x))$$

   and set $A_1(x) = B_1(x), B_1(x) = R_0(x)$; we then repeat to get new polynomials $Q_1(x), R_1(x), A_2(x), B_2(x)$ and so on. The degrees of the polynomials keep getting smaller and will eventually reach a point at which $B_N(x) = 0$; and we will have found our gcd:

$$gcd(A(x),B(x)) = gcd(A_1(x),B_1(x)) = \cdots = gcd(A_N(x),0) = A_N(x)$$

   Here, we have the function that can perform the polynomial long division on $A(x)$ and $B(x)$ and return both the quotient $Q(x)$ and the remainder $R(x)$, i.e. [$Q(x)$, $R(x)$] = div($A(x), B(x)$). The algorithm can be extended from the original integer-based GCD as follows:

```
function gcd(A(x), B(x)):
  if B(x) = 0:
    return A(x)
  else if deg A(x) < deg B(x):
    return gcd(B(x), A(x))
  else:
    (Q(x), R(x)) = div(A(x), B(x))
    return gcd(B(x), R(x))
```

(b) Let $P(x) = x^4 - 1$ and $Q(x) = x^3 + x^2$ in standard form. Prove there are no polynomials $A(x)$ and $B(x)$ such that $A(x)P(x) + B(x)Q(x) = 1$ for all x.

Answer: We can compute a gcd of $P(x)$ and $Q(x)$ using the algorithm in part (a), and show that it is not 1.

$$\begin{aligned} gcd(x^4-1, \quad x^3+x^2) \quad &// \quad x^4-1 = (x^3+x^2)(x-1)+(x^2-1) \\ gcd(x^3+x^2, \quad x^2-1) \quad &// \quad x^3+x^2 = (x^2-1)(x+1)+(x+1) \\ gcd(x^2-1, \quad x+1) \quad &// \quad x^2-1 = (x+1)(x-1)+0 \\ gcd(x+1, \quad 0) \quad &// \quad D(x) = x+1 \end{aligned}$$

We can also derive that $gcd(P(x), Q(x))$ has the smallest degree among all the polynomials that can be expressed as a linear combination of $P(x)$ and $Q(x)$ (see proof below). And since 1 is of degree 0, which is smaller than 1, the degree of $gcd(P(x), Q(x)) = x+1$, there exists no such linear combination.

[Consider the following set:

$$I = \{S(x)P(x) + T(x)Q(x) : S(x), T(x) \text{in the same field as} P(x), Q(x)\}$$

Pick a polynomial $D(x) \in I$ of the smallest degree. We have

$$D(x) = S(x)P(x) + T(x)Q(x) \tag{1}$$

We want to show that

- $D(x)$ is a common divisor of $P(x)$ and $Q(x)$.
- Any common divisor of $P(x)$ and $Q(x)$ must divides $D(x)$.

If these two properties hold, $D(x) = gcd(P(x), Q(x))$.

From polynomial long division of $P(x)$ and $D(x)$, we also obtain

$$P(x) = D(x)E(x) + R(x) \tag{2}$$

where $E(x)$ is the quotient and the remainder $R(x)$ can either be 0 or has $\deg R(x) < \deg D(x)$.From (1) and (2), it follows that

$$\begin{aligned} R(x) = P(x) - D(x)E(x) &= P(x) - [S(x)P(x) + T(x)Q(x)] \cdot E(x) &(3) \\ &= [1 - S(x)E(x)] \cdot P(x) - [T(x)E(x)] \cdot Q(x) &(4) \end{aligned}$$

So $R(x)$ is also a linear combination of $P(x)$ and $Q(x)$, but $D(x)$ is defined to have the smallest degree; therefore $R(x) = 0$, which means that $D(x)$ divides $P(x)$. A similar argument shows that $D(x)$ divides $Q(x)$.

We now want to show that any common divisor $C(x)$ of $P(x)$ and $Q(x)$ must divide $D(x)$.

Let $P(x) = C(x)P'(x)$ and $Q(x) = C(x)Q'(x)$. We have that $D(x) = S(x)C(x)P'(x) + T(x)C(x)Q'(x) = C(x)[S(x)P'(x) + T(x)Q'(x)]$, so C(x) divides D(x).

Therefore, $D(x)$ is the greatest common divisor of $P(x)$ and $Q(x)$, and is of the form $S(x)P(x) + T(x)Q(x)$.

Alternative proof.

Proof by contradiction. Assume that there is $A(x)$ and $B(x)$ such that $A(x)P(x) + B(x)Q(x) = 1$.

We know that $gcd(P,Q) = x+1$, so:

$$A(x)P(x)+B(x)Q(x) = (x+1)[A(x)P'(x)+B(x)Q'(x)] = 1$$

Let $A(x)P'(x)+B(x)Q'(x) = Z(x)$. $(x+1)Z(x)$ is then a polynomial of degree at least 1. However, there is no polynomial $Z(x)$ such that $(x+1)Z(x) = 1$. Contradict! Therefore, there is no $A(x)$ and $B(x)$ such that $A(x)P(x)+B(x)Q(x) = 1$.]

(c) Find polynomials $A(x)$ and $B(x)$ such that $A(x)P(x)+B(x)Q(x) = x+1$ for all x.

Answer: Using extended gcd for polynomials, we can work our way backwards from the result of part (b) to find $A(x)$ and $B(x)$. We know that

$$x+1 = (x^3+x^2)-(x+1)(x^2-1)$$

Plugging in the formula for x^2-1, we get

$$\begin{aligned} x+1 &= (x^3+x^2)-(x+1)[(x^4-1)-(x^3+x^2)(x-1)] \\ &= -(x+1)(x^4-1)+x^2(x^3+x^2) \end{aligned}$$

So therefore, $A(x) = -(x+1)$ and $B(x) = x^2$.

2. **(Berlekamp–Welch algorithm)**

In this question we will go through an example of error-correcting codes with general errors. We will send a message (m_0,m_1,m_2) of length $n = 3$. We will use an error-correcting code for $k = 1$ general error, doing arithmetic modulo 5.

(a) Suppose $(m_0,m_1,m_2) = (4,3,2)$. Use Lagrange interpolation to construct a polynomial $P(x)$ of degree 2 (remember all arithmetic is $\bmod 5$) so that $(P(0),P(1),P(2)) = (m_0,m_1,m_2)$. Then extend the message to lengeth $n+2k$ by appending $P(3),P(4)$. What is the polynomial $P(x)$ and what is the message $(c_0,c_1,c_2,c_3,c_4) = (P(0),P(1),P(2),P(3),P(4))$ that is sent?

Answer: We use Lagrange interpolation to construct the unique quadratic polynomial $P(x)$ such that $P(0) = m_0 = 4, P(1) = m_1 = 3, P(2) = m_2 = 2$.

$$\begin{aligned} \Delta_0(x) &= \frac{(x-1)(x-2)}{(0-1)(0-2)} = \frac{x^2-3x+2}{2} \\ \Delta_1(x) &= \frac{(x-0)(x-2)}{(1-0)(1-2)} = \frac{x^2-2x}{-1} \\ \Delta_2(x) &= \frac{(x-0)(x-1)}{(2-0)(2-1)} = \frac{x^2-x}{2} \\ P(x) &= m_0\Delta_0(x)+m_1\Delta_1(x)+m_2\Delta_2(x) \\ &= 4\Delta_0(x)+3\Delta_1(x)+2\Delta_2(x) \\ &= -x+4 \end{aligned}$$

[Note that all arithmetic is $\bmod 5$, so for example $2^{-1} \equiv 3 \pmod 5$]. Then we compute $P(3) = 1$ and $P(4) = 0$, so our message is 43210.

(b) Suppose the message is corrupted by changing c_0 to 0. We will locate the error using the Berlekamp–Welsh method. Let $E(x) = x + b_0$ be the error-locator polynomial, and $Q(x) = P(x)E(x) = a_3x^3 + a_2x^2 + a_1x + a_0$ be a polynomial with unknown coefficients. Write down the system of linear equations (involving unknowns a_0, a_1, a_2, a_3, b_0) in the Berlekamp–Welsh method. You need not solve the equations.

Answer: The message received is $(c_0', c_1', c_2', c_3', c_4') = (0,3,2,1,0)$. Let $R(x)$ be the function such $R(i) = c_i'$ for $0 \le i < 5$. Let $E(x) = x + b_0$ be the error-locator polynomial, and $Q(x) = P(x)E(x) = a_3x^3 + a_2x^2 + a_1x + a_0$. Since $Q(i) = P(i)E(i) = R(i)E(i)$ for $1 \le i < 5$, we have the following equalities $\pmod 5$:

$$
\begin{aligned}
Q(0) &= 0E(0)\\
Q(1) &= 3E(1)\\
Q(2) &= 2E(2)\\
Q(3) &= 1E(3)\\
Q(4) &= 0E(4)
\end{aligned}
$$

They lead to the following system of linear equations:

$$
\begin{array}{rcrcrcrcrcl}
 & & & & & & a_0 & & & = & 0\\
a_3 & + & a_2 & + & a_1 & + & a_0 & - & 3b_0 & = & 3\\
8a_3 & + & 4a_2 & + & 2a_1 & + & a_0 & - & 2b_0 & = & 4\\
27a_3 & + & 9a_2 & + & 3a_1 & + & a_0 & - & b_0 & = & 3\\
64a_3 & + & 16a_2 & + & 4a_1 & + & a_0 & & & = & 0
\end{array}
$$

(c) The solution to the equations in part (b) is $b_0 = 0, a_0 = 0, a_1 = 4, a_2 = 4, a_3 = 0$. Show how the recipient can recover the original message (m_0, m_1, m_2).

Answer: From the solution, we know

$$
\begin{aligned}
Q(x) &= a_3x^3 + a_2x^2 + a_1x + a_0 = -x^2 + 4x\\
E(x) &= x + b_0 = x
\end{aligned}
$$

Since $Q(x) = P(x)E(x)$, the recipient can compute $P(x) = Q(x)/E(x) = -x + 4$ [note that this is the same polynomial $P(x)$ from part (a) used by the sender]. The recipient may deduce the location of the error from $E(x)$ as follows. There is only one error at location e_1, we have $E(x) = (x - e_1) = x$, so $e_1 = 0$ and the error is at position 0. To correct the error we evaluate $P(0) = 4$. Since the other two positions m_1, m_2 of the message are uncorrupted, we recover the original message $(m_0, m_1, m_2) = (4,3,2)$.

3. **Error-correcting codes: An example**

In this question we will go through an example of error-correcting codes with general errors. Since we will do this by hand, the message we will send is going to be short, consisting of $n = 3$ numbers, each modulo 5, and the number of errors will be $k = 1$.

(a) First, construct the message. Let $a_0 = 4$, $a_1 = 3$, and $a_2 = 2$; use the polynomial interpolation formula to construct a polynomial $P(x)$ of degree 2 (remember that all arithmetic is mod 5) so that $P(0) = a_0$, $P(1) = a_1$, and $P(2) = a_2$; then extend the message to length $n + 2k$ by adding $P(3)$ and $P(4)$. What is the polynomial $P(x)$ and what is the message that is sent?

(b) Suppose the message is corrupted by changing a_0 to 0. Use the Berlekamp-Welsh method to detect the location of the error and to reconstruct the original message $a_0a_1a_2$. Show clearly all your work.

Answer:

(i) We use Lagrange interpolation to construct the unique quadratic polynomial $P(x)$ such that $P(0) = a_0 = 4$, $P(1) = a_1 = 3$, and $P(2) = a_2 = 2$.

$$\begin{aligned}
\Delta_0(x) &= \frac{(x-1)(x-2)}{(0-1)(0-2)} = \frac{x^2-3x+2}{2} \\
\Delta_1(x) &= \frac{(x-0)(x-2)}{(1-0)(1-2)} = \frac{x^2-2x}{-1} \\
\Delta_2(x) &= \frac{(x-0)(x-1)}{(2-0)(2-1)} = \frac{x^2-x}{2}
\end{aligned}$$

$$\begin{aligned}
P(x) &= a_0\Delta_0(x) + a_1\Delta_1(x) + a_2\Delta_2(x) \\
&= 4\Delta_0(x) + 3\Delta_1(x) + 2\Delta_2(x) \\
&= 4(\tfrac{x^2-3x+2}{2}) + 3(\tfrac{x^2-2x}{-1}) + 2(\tfrac{x^2-x}{2}) \\
&= -x+4
\end{aligned}$$

[Note that all arithmetic is mod 5, and that $2^{-1} = 3 \bmod 5$.] Then we compute $P(3) = 1$ and $P(4) = 0$, so our message is $m = m_0\ m_1\ m_2\ m_3\ m_4 = 4\ 3\ 2\ 1\ 0$.

(ii) The message received is $m' = m'_0\ m'_1\ m'_2\ m'_3\ m'_4 = 0\ 3\ 2\ 1\ 0$. Let $R(x)$ be a function such that $R(i) = m'_i$ for $i = 0, \ldots, 4$. Since $k = 1$, the error-locator polynomial $E(x) = (x - e_1)$ has degree 1. Let $Q(x) = P(x)E(x)$. Then $Q(x)$ is a polynomial of degree 3 as it is the product of a polynomial of degree 2 and a polynomial of degree 1. Since we know the degrees of $Q(x)$ and $E(x)$ and we also know that the coefficient of the highest-order term of E(x) is 1, we can express these polynomials as

$$\begin{aligned}
Q(x) &= a_3x^3 + a_2x^2 + a_1x + a_0 \\
E(x) &= x + b_0
\end{aligned}$$

Since $Q(i) = P(i)E(i) = R(i)E(i)$ for $i = 0, \ldots, 4$, we know that the following equalities hold (mod 5):

$$\begin{aligned}
Q(0) &= 0E(0) \\
Q(1) &= 3E(1) \\
Q(2) &= 2E(2) \\
Q(3) &= 1E(3) \\
Q(4) &= 0E(4)
\end{aligned}$$

For our example, this leads to the following equations for the coefficients:

$$\begin{array}{rcrcrcrcrcl}
 & & & & & & a_0 & & & = & 0 \\
a_3 & + & a_2 & + & a_1 & + & a_0 & - & 3b_0 & = & 3 \\
8a_3 & + & 4a_2 & + & 2a_1 & + & a_0 & - & 2b_0 & = & 4 \\
27a_3 & + & 9a_2 & + & 3a_1 & + & a_0 & - & b_0 & = & 3 \\
64a_3 & + & 16a_2 & + & 4a_1 & + & a_0 & & & = & 0
\end{array}$$

For completeness we will spell out the solution to these equations. The first step is to eliminate a_0.

$$\begin{array}{rcrcrcrcl} a_3 & + & a_2 & + & a_1 & - & 3b_0 & = & 3 \\ 8a_3 & + & 4a_2 & + & 2a_1 & - & 2b_0 & = & 4 \\ 27a_3 & + & 9a_2 & + & 3a_1 & - & b_0 & = & 3 \\ 64a_3 & + & 16a_2 & + & 4a_1 & & & = & 0 \end{array}$$

Next we use Guassian elimination to diagonalize and solve this system of equations.

$$\left[\begin{array}{cccc|c} 1 & 1 & 1 & -3 & 3 \\ 8 & 4 & 2 & -2 & 4 \\ 27 & 9 & 3 & -1 & 3 \\ 64 & 16 & 4 & 0 & 0 \end{array}\right]$$

Subtracting 8 times the first row from the second row, 27 times the first row from the third row, and 64 times the first row from the fourth row, we obtain

$$\left[\begin{array}{cccc|c} 1 & 1 & 1 & -3 & 3 \\ 0 & -4 & -6 & 22 & -20 \\ 0 & -18 & -24 & 80 & -78 \\ 0 & -48 & -60 & 192 & -192 \end{array}\right]$$

Multiplying the second row by $-\frac{1}{4}$, we obtain

$$\left[\begin{array}{cccc|c} 1 & 1 & 1 & -3 & 3 \\ 0 & 1 & \frac{3}{2} & -\frac{11}{2} & 5 \\ 0 & -18 & -24 & 80 & -78 \\ 0 & -48 & -60 & 192 & -192 \end{array}\right]$$

Adding 18 times the second row to the third row and 48 times the second row to the fourth row, we obtain

$$\left[\begin{array}{cccc|c} 1 & 1 & 1 & -3 & 3 \\ 0 & 1 & \frac{3}{2} & -\frac{11}{2} & 5 \\ 0 & 0 & 3 & -19 & 12 \\ 0 & 0 & 12 & -72 & 48 \end{array}\right]$$

Multiplying the third row by $\frac{1}{3}$ and subtracting 12 times the new second row from the third row, we obtain

$$\left[\begin{array}{cccc|c} 1 & 1 & 1 & -3 & 3 \\ 0 & 1 & \frac{3}{2} & -\frac{11}{2} & 5 \\ 0 & 0 & 1 & -\frac{19}{3} & 4 \\ 0 & 0 & 0 & 4 & 0 \end{array}\right]$$

The fourth row of the matrix tells us that $4b_0$ and hence $b_0 = 0$.
The third row tells us that $a_1 - \frac{19}{3}b_0 = 4$ and hence $a_1 = 4$.
The second row tells us

$$\begin{aligned} a_2 + \frac{3}{2}a_1 - \frac{11}{2}b_0 &= 5 \\ a_2 + (\frac{3}{2})(4) - (\frac{11}{2})(0) &= 5 \\ a_2 &= -1. \end{aligned}$$

The first row tells us

$$\begin{aligned} a_3 + a_2 + a_1 - 3b_0 &= 3 \\ a_3 - 1 + 4 - (3)(0) &= 3 \\ a_3 &= 0 \end{aligned}$$

Thus,

$$\begin{aligned} Q(x) &= a_3x^3 + a_2x^2 + a_1x + a_0 \\ &= (0)x^3 + (-1)x^2 + 4x + 0 \\ &= -x^2 + 4x \end{aligned}$$

and

$$\begin{aligned} E(x) &= x + b_0 \\ &= x + 0 \\ &= x \end{aligned}$$

Since $Q(x) = P(x)E(x)$, we know that $P(x) = \frac{Q(x)}{E(x)}$. Thus $P(x) = -x+4$. [To check our working, we note that this is the same polynomial $P(x)$ that we know, from part (a), was used by the sender.]
Since we now know $E(x)$, we know the location of all the errors: in this case, since there is only one error location $x = e_1$, we have $E(x) = (x - e_1) = x + b_0 = x + 0$, so we deduce that $e_1 = 0$ and the only error is at position 0. So to correct the error in position 0 we evaluate $P(0) = 4$. Since we also know that the other two positions m_1 and m_2 of the message are not corrupted, we recover the original message as $m_1 = 4$, $m_2 = 3$, $m_3 = 2$. This completes the error correction.

4. **Counting Cantor**
Show that the Cantor set is uncountably infinite.

HINT: There are two standard ways to prove that something is uncountable: Find a bijection between it and some other uncountable set; or, use diagonalization.

Also, you might find it useful to know the following alternative definition of the Cantor set: S is the set of real numbers $x \in [0,1]$ that can be represented in base 3 (trinary) using only 0's and 2's (i.e., no 1's). (Be warned that there is some ambiguity in trinary representations: $1/3$ could be represented as either $0.10000\ldots$ or $0.02222\ldots$. For this definition, we require that the ambiguity be resolved by always using representations that end in $02222\ldots$ rather than $10000\ldots$, whenever you have a choice.)

Answer: First, I argue that the two definitions of the Cantor set are equivalent. The initial interval contains all the elements between 0 and 1, each of which can be represented as a ternary string that starts with either 0.0, 0.1, or 0.2. There are an equal number of elements that start with each of these, since the remaining substring can take on the same values in each case. Also, all numbers that start with 0.1 are greater than any that start with 0.0 and less than any that start with 0.2 (ignoring the endpoints). Thus the middle third of the elements are those that start with 0.1, so removing these results in only those that start with 0.0 and 0.2.

We repeat this reasoning in the second iteration to eliminate all strings that start with 0.01 and 0.21, in the third to eliminate all those that start with 0.001, 0.021, 0.201, and 0.221, and so on *ad inifinitum*. What we are left with are those strings that only contain the digits 0 or 2.

Now we can perform a diagonalization to show that the Cantor set is uncountably infinite. Suppose we have a bijection between S and $\mathbb{N}$, an ordered list of the elements of S, $(s_1, s_2, \cdots)$. We can construct

another number s' such that in the ith position after the decimal, s' has a 0 if s_i has a 2 in that position, and s' has a 2 if s_i has a 0 in the ith position. Thus s' differs from every $s_i \in S$. But notice that s' contains only 0's and 2's (no 1's), so it is in S. This is a contradiction, so no such bijection can exist. Thus S is uncountably infinite.

An alternative solution is to give a bijection between S and an uncountable set. Recall that the set of real numbers $R_{0,1} = [0,1]$ is uncountable. A simple bijection between S and $R_{0,1}$ is to replace each 2 that appears in an element of S by a 1, to get a real number in the range $[0,1]$ that is encoded in binary. It is easy to see that this is a bijection. Thus S is uncountable.