

# CS 330: Homework 5

Michael Lin

April 19, 2016

## Problem 1

The dual to the linear program is as follows:

$$\begin{aligned} \min & 3y_1 + 5y_2 \\ & 2y_1 + y_2 \geq 1 \\ & y_1 + 3y_2 \geq 1 \\ & y_1, y_2 \geq 0 \end{aligned}$$

## Solving Primal LP

Solve the primal LP using simplex method. First, convert constraints to equational form by introducing slack variables  $x_3, x_4$ :

$$\begin{aligned} 2x_1 + x_2 + x_3 &= 3 \\ x_1 + 3x_2 + x_4 &= 5 \end{aligned}$$

Begin with basic feasible solutions  $x_1, x_2 = 0$ . Note,  $z$  is the variable we want to maximize. First tableau:

$$\begin{aligned} x_3 &= 3 - 2x_1 - x_2 \\ x_4 &= 5 - x_1 - 3x_2 \\ z &= x_1 + x_2 \end{aligned} \tag{1}$$

We can increase  $z$  by setting  $x_1$  to 1.5. Now we re-express the equations in terms of  $x_1, x_4$  in second tableau:

$$\begin{aligned} x_1 &= 1.5 - 0.5x_2 - 0.5x_3 \\ x_4 &= 3.5 - 2.5x_2 + 0.5x_3 \\ z &= 1.5 + 0.5x_2 - 0.5x_3 \end{aligned} \tag{2}$$

We can increase  $z$  by setting  $x_2$  to 1.4. Now we re-express the equations in terms of  $x_1, x_2$  in third tableau:

$$\begin{aligned} x_1 &= 0.8 - 0.6x_3 + 0.2x_4 \\ x_2 &= 1.4 + 0.2x_3 - 0.4x_4 \\ z &= 2.2 - 0.4x_3 - 0.2x_4 \end{aligned} \tag{3}$$

Since we can no longer increase  $z$ , we stop, and conclude  $x_1 = 0.8, x_2 = 1.4$ , which gives  $z = 2.2$ .

## Solving Dual LP

Solve the dual LP using simplex method. First, convert objective function to maximization problem, and convert constraints to equational form by introducing slack variables  $y_3, y_4$ :

$$\begin{aligned} \max & \{-3y_1 - 5y_2\} \\ & 2y_1 + y_2 - y_3 = 1 \\ & y_1 + 3y_2 - y_4 = 1 \end{aligned}$$

Begin with basic feasible solutions  $y_1 = 1, y_2 = 0$ . Note,  $z$  is the variable we want to maximize. First tableau:

$$\begin{aligned}y_1 &= 1 - 3y_2 + y_4 \\y_3 &= 1 - 5y_2 + 2y_4 \\z &= -3 + 4y_2 - 3y_4\end{aligned}\tag{4}$$

We can increase  $z$  by setting  $y_2$  to 0.2. Now we re-express the equations in terms of  $y_1, y_2$  in second tableau:

$$\begin{aligned}y_1 &= 0.4 + 0.6y_3 + 1.6y_4 \\y_2 &= 0.2 - 0.2y_3 - 0.2y_4 \\z &= -2.2 - 0.8y_3 - 3.8y_4\end{aligned}\tag{5}$$

Since we can no longer increase  $z$ , we stop, and conclude  $y_1 = 0.4, y_2 = 0.2$ , which gives  $z = -2.2$ . In terms of the original problem, we want  $-z$  as the objective function, so the objective function attains minimum at 2.2.

## Problem 2

- (i) The payoff matrix from player  $R$ 's perspective is:

	$H_C$	$T_C$
$H_R$	1	-1
$T_R$	-1	1

Call this matrix  $M$ .

- (ii) From lecture, the value of the game is  $x^T M y$  where  $x, y$  are strategies of player  $R, C$ , respectively. Suppose  $C$  is going to choose a strategy that will minimize  $R$ 's winning, i.e.,

$$\min_y x^T M y$$

Let's call this value  $z$ . Given that we know  $C$  will choose this strategy, from  $R$ 's perspective, we want to maximize our winning, which means maximizing  $z$ . Since  $z$  is the minimum winning, we know that given  $C$ 's play,  $R$ 's winning will be at least  $z$ . We thus have the linear program:

$$\max z$$

$$x_1 - x_2 \geq z \tag{6}$$

$$-x_1 + x_2 \geq z \tag{7}$$

$$x_1 + x_2 = 1 \tag{8}$$

$$x_1, x_2 \geq 0$$

Constraint (6) is if  $C$  plays heads, and constraint (7) is if  $C$  plays tails. Constraint (8) ensures strategy  $x$  is a probability distribution that sums to 1. Solving constraint (8) for  $x_2$  and substituting the result into constraints (6) and (7) yields the following:

$$z \leq 2x_1 - 1 \tag{9}$$

$$z \leq 1 - 2x_1 \tag{10}$$

Since the optimal solution is at a vertex (i.e. intersection of constraint lines), and we have only two lines, we know that the optimal solution is just the point of intersection of (9) and (10). Solve, we get  $x_1 = 1/2$ ,  $z = 0$ , and substituting into (8) yields  $x_2 = 1/2$ . Therefore, the value of the game is 0, and the optimal strategy for  $R$  is even split between heads and tails (1/2 each). By symmetry, the optimal strategy for  $C$  is the same.

### Problem 3

- (i) We can set up an integral linear program to find the minimum vertex cover. Let  $x$  be a vector for the vertex cover, where  $x_j$  equals 1 if vertex  $j$  is in the vertex cover and 0 otherwise. Let  $A$  be a matrix with dimensions  $|E| \times |V|$ , and let entry  $A(i, j)$  equal 1 if edge  $i$  connects vertex  $j$ , and 0 otherwise. This means each row of the matrix has exactly two entries of 1's and the rest 0's, which makes sense because each edge has only 2 connected vertices. Then, the linear program becomes:

$$\begin{aligned} \min \mathbf{1}^T x \\ Ax \geq \mathbf{1} \\ x \geq 0, x \in \mathbb{Z} \end{aligned}$$

where  $\mathbf{1}$  is the vector of 1's of length  $|V|$ . The constraint  $Ax \geq \mathbf{1}$  means that for a given edge (i.e. a row), at least one of the vertices connected by it must be in the vertex cover. The constraint  $x \geq 0$  implies that some vertices are in the vertex cover while others are not. To match the sign of the form given in problem statement, the constraint  $Ax \geq \mathbf{1}$  can be written as  $-Ax \leq -\mathbf{1}$ .

- (ii) Consider the cyclic graph with vertices  $x, y, z$ . The linear program is as follows:

$$\begin{aligned} \min x + y + z \\ x + y \geq 1 \\ y + z \geq 1 \\ x + z \geq 1 \\ x, y, z \geq 0 \end{aligned}$$

The optimal solution is  $(x, y, z) = (0.5, 0.5, 0.5)$ , which is not an integer solution.

## Problem 4

Since the product of the two polynomials will be degree 4, we choose  $n = 4$ , and the 4th roots of unity are:

$$\omega_{0,4} = 1 \quad \omega_{1,4} = i \quad \omega_{2,4} = -1 \quad \omega_{3,4} = -i$$

Denote  $A(x) = 1 + x + 2x^2$ , and  $B(x) = 2 + 3x$ .

### 4.1 FFT for $A(x)$

We will evaluate  $A$  at the 4th roots of unity using the recursive FFT algorithm. First, observe the following decomposition:

$$A(x) = A_E(x^2) + xA_O(x^2)$$

where  $A_E(x) = 1 + 2x$  and  $A_O(x) = 1 - x$ .  $A_E(x)$  can be further decomposed into  $(A_E)_E(x) = 1$  and  $(A_E)_O(x) = 2$  such that

$$A_E(x) = (A_E)_E(x^2) + x(A_E)_O(x^2)$$

which is easy to evaluate.

#### 4.1.1 Evaluate $A_E$ at 2nd roots of unity

$$\begin{aligned} A_E(\omega_{0,2}) &= (A_E)_E(\omega_{0,2}) + \omega_{0,2}(A_E)_O(\omega_{0,2}) = 1 + 1 \cdot 2 = 3 \\ A_E(\omega_{1,2}) &= (A_E)_E(\omega_{1,2}) - \omega_{1,2}(A_E)_O(\omega_{1,2}) = 1 - 1 \cdot 2 = -1 \end{aligned}$$

#### 4.1.2 Evaluate $A$ at 4th roots of unity

Now we can evaluate  $A$  using  $A_E$  and  $A_O$ .

$$\begin{aligned} A(\omega_{0,4}) &= A_E(\omega_{0,4}) + \omega_{0,4}A_O(\omega_{0,4}) = A_E(\omega_{0,2}) + 1 \cdot A_O(\omega_{0,2}) = 3 + 1 = 4 \\ A(\omega_{1,4}) &= A_E(\omega_{1,4}) + \omega_{1,4}A_O(\omega_{1,4}) = A_E(\omega_{1,2}) + i \cdot A_O(\omega_{1,2}) = -1 + i \\ A(\omega_{2,4}) &= A_E(\omega_{0,4}) - \omega_{0,4}A_O(\omega_{2,4}) = A_E(\omega_{0,2}) - 1 \cdot A_O(\omega_{0,2}) = 3 - 1 = 2 \\ A(\omega_{3,4}) &= A_E(\omega_{1,4}) - \omega_{1,4}A_O(\omega_{3,4}) = A_E(\omega_{1,2}) - i \cdot A_O(\omega_{1,2}) = -1 - i \end{aligned}$$

### 4.2 FFT for $B(x)$

We will similarly evaluate  $B$  at the 4th roots of unity using the recursive FFT algorithm. First, observe the following decomposition:

$$B(x) = B_E(x^2) + xB_O(x^2)$$

where  $B_E(x) = 2$  and  $B_O(x) = 3$

#### 4.2.1 Evaluate $B$ at 4th roots of unity

Now we can evaluate  $B$  using  $B_E$  and  $B_O$ .

$$\begin{aligned} B(\omega_{0,4}) &= B_E(\omega_{0,4}) + \omega_{0,4}B_O(\omega_{0,4}) = B_E(\omega_{0,2}) + 1 \cdot B_O(\omega_{0,2}) = 2 + 3 = 5 \\ B(\omega_{1,4}) &= B_E(\omega_{2,4}) + \omega_{1,4}B_O(\omega_{2,4}) = B_E(\omega_{1,2}) + i \cdot B_O(\omega_{1,2}) = 2 + 3i \\ B(\omega_{2,4}) &= B_E(\omega_{0,4}) - \omega_{0,4}B_O(\omega_{4,4}) = B_E(\omega_{0,2}) - 1 \cdot B_O(\omega_{0,2}) = 2 - 3 = -1 \\ B(\omega_{3,4}) &= B_E(\omega_{2,4}) - \omega_{1,4}B_O(\omega_{2,4}) = B_E(\omega_{1,2}) - i \cdot B_O(\omega_{1,2}) = 2 - 3i \end{aligned}$$

### 4.3 Interpolation

Let  $C(x) = A(x)B(x)$ . At the roots of unity,  $C(x)$  is:

$$\begin{aligned} C(\omega_{0,4}) &= 4 \cdot 5 = 20 \\ C(\omega_{1,4}) &= (-1 + i)(2 + 3i) = -5 - i \\ C(\omega_{2,4}) &= 2 \cdot -1 = -2 \\ C(\omega_{3,4}) &= (-1 - i)(2 - 3i) = -5 + i \end{aligned}$$

Let  $D(x) = \sum_{j=0}^{n-1} C(\omega_{j,n})x^j$ . Thus,

$$D(x) = 20 - (5 + i)x - 2x^2 + (-5 + i)x^3$$

Now, to interpolate, we use the following lemma:

**Lemma 1.**  $c_{n-j} = D(\omega_{j,n})/n$ , where  $c_{n-j}$  are coefficients of the polynomial  $C(x)$ .

Therefore, we need to first evaluate  $D$  at the 4th roots of unity. Again, we can express  $D(x)$  as the following:

$$D(x) = D_E(x^2) + xD_O(x^2)$$

where  $D_E(x) = 20 - 2x$  and  $D_O(x) = -(5 + i) + (-5 + i)x$ .

Recursively, we can also re-express  $D_E(x), D_O(x)$ :

$$D_E(x) = (D_E)_E(x^2) + x(D_E)_O(x^2)$$

where  $(D_E)_E(x) = 20$  and  $(D_E)_O(x) = -2$ .

$$D_O(x) = (D_O)_E(x^2) + x(D_O)_O(x^2)$$

where  $(D_O)_E(x) = -(5 + i)$  and  $(D_O)_O(x) = -5 + i$ .

#### 4.3.1 Evaluate $D_E$ at 2nd roots of unity

$$\begin{aligned} D_E(\omega_{0,2}) &= (D_E)_E(\omega_{0,2}) + \omega_{0,2}(D_E)_O(\omega_{0,2}) = 20 + 1 \cdot (-2) = 18 \\ D_E(\omega_{1,2}) &= (D_E)_E(\omega_{0,2}) - \omega_{0,2}(D_E)_O(\omega_{0,2}) = 20 - 1 \cdot (-2) = 22 \end{aligned}$$

#### 4.3.2 Evaluate $D_O$ at 2nd roots of unity

$$\begin{aligned} D_O(\omega_{0,2}) &= (D_O)_E(\omega_{0,2}) + \omega_{0,2}(D_O)_O(\omega_{0,2}) = -5 - i + 1 \cdot (-5 + i) = -10 \\ D_O(\omega_{1,2}) &= (D_O)_E(\omega_{0,2}) - \omega_{0,2}(D_O)_O(\omega_{0,2}) = -5 - i - 1 \cdot (-5 + i) = -2i \end{aligned}$$

#### 4.3.3 Evaluate $D$ at 4th roots of unity

$$\begin{aligned} D(\omega_{0,4}) &= D_E(\omega_{0,4}) + \omega_{0,4}D_O(\omega_{0,4}) = D_E(\omega_{0,2}) + 1 \cdot D_O(\omega_{0,2}) = 18 - 10 = 8 \\ D(\omega_{1,4}) &= D_E(\omega_{2,4}) + \omega_{1,4}D_O(\omega_{2,4}) = D_E(\omega_{1,2}) + i \cdot D_O(\omega_{1,2}) = 22 + i \cdot (-2i) = 24 \\ D(\omega_{2,4}) &= D_E(\omega_{0,4}) - \omega_{0,4}D_O(\omega_{4,4}) = D_E(\omega_{0,2}) - 1 \cdot D_O(\omega_{0,2}) = 18 + 10 = 28 \\ D(\omega_{3,4}) &= D_E(\omega_{2,4}) - \omega_{1,4}D_O(\omega_{2,4}) = D_E(\omega_{1,2}) - i \cdot D_O(\omega_{1,2}) = 22 - i \cdot (-2i) = 20 \end{aligned}$$

#### 4.3.4 Determine coefficients of $C(x)$

Therefore,

$$c_0 = D(\omega_{4,4})/4 = D(\omega_0, 4)/4 = 8/4 = 2$$

$$c_1 = D(\omega_{3,4})/4 = 20/4 = 5$$

$$c_2 = D(\omega_{2,4})/4 = 28/4 = 7$$

$$c_3 = D(\omega_{1,4})/4 = 24/4 = 6$$

So, we have:

$$C(x) = 2 + 5x + 7x^2 + 6x^3$$

## Problem 5

For each number in the set  $X$ , write it as the power of each term in the polynomial. For example, if  $X = \{2, 4, 7\}$ , then write  $X(t) = t^2 + t^4 + t^7$ . Do the same for each number in the set  $Y$ , and denote the corresponding polynomial as  $Y(t)$ . From here, we simply need to multiply  $X(t)$  with  $Y(t)$  by first using FFT evaluation to express in point representation then using interpolation to re-express in coefficient representation. The set  $Z$  is just the values of the exponent in the polynomial  $X(t)Y(t)$ .

*Proof of correctness.* When multiplying polynomials, all possible combinations of exponents are added, which is exactly what we want in finding the set  $Z$ . Since the FFT evaluation and interpolation method of multiplying polynomials is correct, the algorithm correctly computes the set  $Z$ .  $\square$

*Proof of runtime.* Since each value in a set is a unique natural number, and all values  $x \in X$  and  $y \in Y$  are less than  $M$ , there are at most  $M$  values in each set. Writing the polynomials  $X(t)$  and  $Y(t)$  thus takes  $O(M)$  time. From lecture, we know that the polynomial multiplication step in our algorithm takes  $O(M \log M)$  time. The polynomial  $X(t)Y(t)$  has at most  $2M$  terms, so returning the exponents of the polynomial will take  $O(M)$  time. Therefore, the algorithm runs in  $O(M \log M)$  time.  $\square$