CS531 HW1 Solution

Chain-Wu Lee, lee-d@cs.buffalo.edu

February 24, 1998

1. We know that the lower bound of the sorting problem $= \Omega(n \lg n)$ and for an n input sorting network, we can have at most $n/2$ comparisons, therefore the depth of the sorting network will be at least $\lg n$.

2. Similar as problem 1, we need at least $\Omega(n \lg n)$ comparison to sort arbitrary input, therefore we need at least $\Omega(n \lg n)$ comparators in the sorting network.

3. Suppose a number inputs at position i and outputs at position p_i. Because the transposition network can only exchange data with adjacent lines, therefore the shortest distance from i to j is $abs(i - p_i)$. For all the inputs, the total distance will be

$$\sum_{i=1}^{n} abs(i - p_i) \leq \sum_{i=1}^{n} \frac{(n-1)}{2} = \Omega(n^2)$$

4. $n = 1$, single element and is monotonic
$n = 2$, can be either increasing or decreasing and both of them are monotonic
$n = 3$, can be either increasing or decreasing or increasing then decreasing or decreasing then increasing. The first two are monotonic and the last two are bitonic. $n = 4$, for example, the series [0 2 1 3] is not bitonic, therefore the answer of the question is 4.

5. (a) Let's first take a look at one comparator, one of the output of the comparator is the larger number among the two inputs and the other one is the smaller number. If F is a monotonic increase function, we want to prove the following figure is true.

```
x1 -XXX- max(x1, x2)        F(x1) -XXX- F(max(x1, x2))
    XXX                -->         XXX
x2 -XXX- min(x1, x2)        F(x2) -XXX- F(min(x1, x2))
```

If we have $x_1 \geq x_2$, the output of the comparator after applying F will be $\max(F(x_1), F(x_2))$ and $\min(F(x_1), F(x_2))$. And because F is a monotonic function, $F(x_1) \geq F(x_2)$. Therefore $\max(F(x_1), F(x_2)) = F(x_1)$ and $\min(F(x_1), F(x_2)) = F(x_2)$. Because $x_1 \geq x_2$, $F(\max(x_1, x_2)) = F(x_1)$, $F(\min(x_1, x_2)) = F(x_2)$. The case is proved. Similarly you can prove $x_1 < x_2$.

Now we can prove the network by induction on the depth d of the wires of the network.

If $d = 0$, the input is identical to the output, therefore the case is trivial.

If $d = n$ is true, by the inductive hypothesis, therefore, the input wires to the comparator carry values x_i and x_j when the input sequence X is applied, then they carry $F(x_i)$ and $F(x_j)$ when the input sequence $F(X)$ is applied. By our earlier claim the output wires of this comparator then carry $F(\min x_i, x_j))$ and $F(\max(x_i, x_j))$. Since they carry $\min(x_i, x_j)$ and $\max(x_i, x_j)$ when the input sequence is X, the question is proved.

(b) Suppose contradiction, then we have a network which can sort 0-1s but cannot sort arbitrary network, then we can assume that the sequence which cannot be sorted is $(x_1, x_2, ...x_i, ...x_j, ...a_n)$, $x_i < x_j$ and after feeding the input to the network, the network put x_j before x_i. Then we can define a monotonic increase function F as $F(x) = 0$ if $x \leq x_i$ and $= 1$ if $x > a_i$. But by 5(a), the network will put $F(a_j) = 1$ ahead of $F(a_i) = 0$, which does not sort the 0-1 input, therefore we prove the problem.

6. By 0-1 principle, if the sorting network can sort 0-1 input, then it can sort any arbitrary input. If we use $\#0_i$ as a notation of number of 0s in the list i. then we have

$$\#0_u = \lceil \frac{\#0_x}{2} \rceil + \lceil \frac{\#0_y}{2} \rceil$$

$$\#0_v = \lfloor \frac{\#0_x}{2} \rfloor + \lfloor \frac{\#0_y}{2} \rfloor$$

Then the 0 difference between U and V can be either 0, 1 or 2

(a) 0 : which means the two series is looked like

```
              i
X =  000...000111...111
Y =  000...000111...111
```

X_i compares with Y_{i-1}, which are all 0s. $X_{i+1} = 0$ compare with $Y_i = 1$ and 0 is ahead of 1 by the sorting rules. Therefore the list is sorted.

(b) 1 : which means the two series is looked like

```
              i
X =  000...000111...111
Y =  000...001111...111
```

X_i compares with Y_{i-1}, which are all 0s. X_{i+1} compares with Y_i, which are all 1s. Therefore the list is sorted.

(c) 2 : which means the two series is looked like

```
              i
X =  000...000111...111
Y =  000...011111...111
```

X_{i-1} compares with Y_{i-2}, which are all 0s. $X_i = 0$ compares with $Y_{i-1} = 1$, which put 0 ahead of 1 by the sorting rule and X_{i+1} compares with Y_i, which are all 1s. Therefore the list is sorted.