Next →← Prev

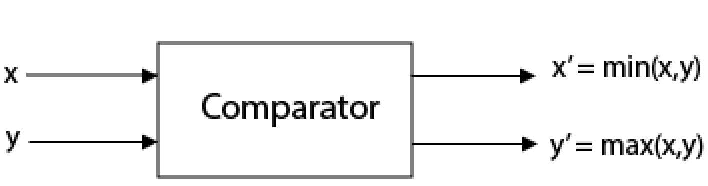
Comparison Networks

A comparison network is made of wires and comparators. A **comparator** is a device with two inputs, x and y, and two outputs, x' and y,' where

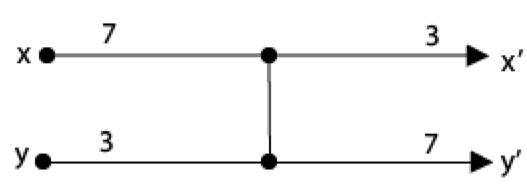
x' = min (x, y)
y' = max (x, y)

In Comparison Networks input appear on the left and outputs on the right, with the smallest input value appearing on the top output and the largest input value appearing on the bottom output. Each comparator operates in O (1) time. In other words, we consider that the time between the appearance of the input values x and y and the production of the output values x' and y' is a constant.

A wire transmits a value from place to place. A comparison network contains n input wires $a_1, a_2, \ldots\ldots a_n$ through which the benefits to be sorted enter the network, and n output wires $b_1, b_2, \ldots\ldots b_n$ which produce the results computed by the network.



(a)



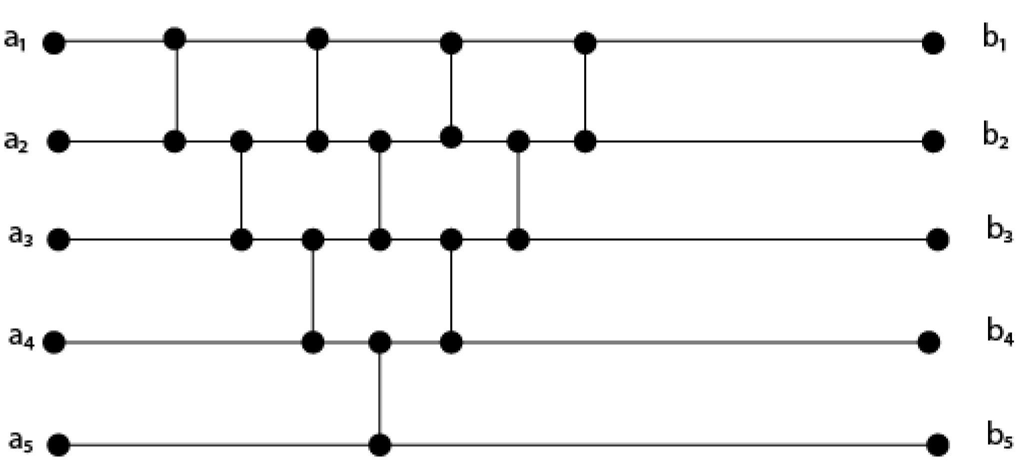
(b)

Comparison Network is a set of comparators interconnected by wires. Running time of comparator can define regarding **depth**.

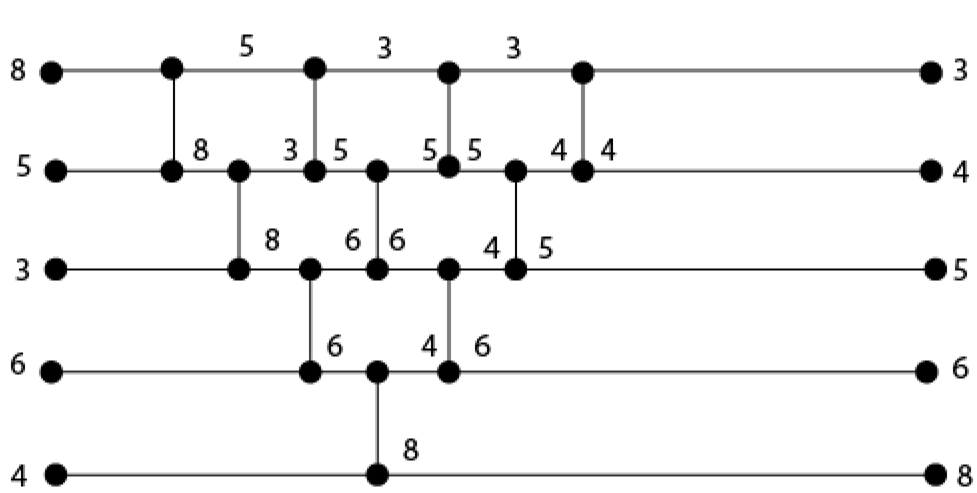
Depth of a Wire: An input wire of a comparison network has depth 0. Now, if a comparator has two input wires with depths d_x and d_y' then its output wires have depth max $(d_x, d_y) + 1$.

A sorting network is a comparison network for which the output sequence is monotonically increasing (that is $b_1 \leq b_2 \leq \ldots . b_n$) for every input sequence.

Fig: A Sorting network based on Insertion Sort



(a)



(b)

Bitonic Sorting Network

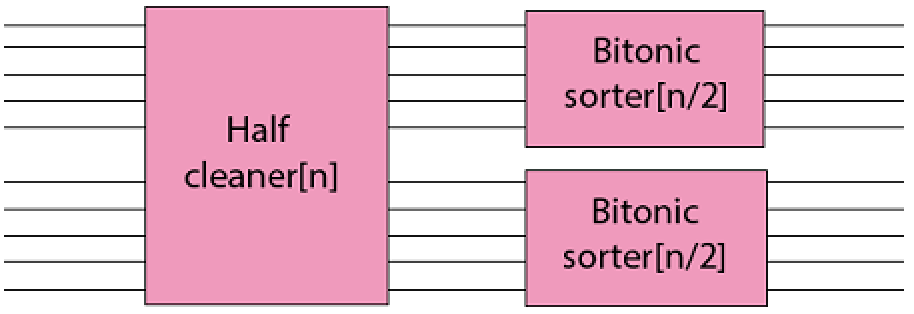
A sequence that monotonically increases and then monotonically decreases, or else monotonically decreases and then monotonically increases is called a bitonic sequence. For example: the sequence (2, 5, 6, 9, 3, 1) and (8, 7, 5, 2, 4, 6) are both bitonic. The bitonic sorter is a comparison network that sorts bitonic sequence of 0's and 1's.

Half-Cleaner: A bitonic sorter is containing several stages, each of which is called a half-cleaner. Each half-cleaner is a comparison network of depth 1 in which input line i is compared with line 1+ $\frac{n}{2}$ for i = 1, 2..... $\frac{n}{2}$.

When a bitonic sequence of 0's and 1's is practiced as input to a half-cleaner, the half-cleaner produces an output sequence in which smaller values are in the top half, larger values are in the bottom half, and both halves are bitonic, and at least one of the halves is clean.

Bitonic Sorter: By recursively connecting half-cleaners, we can build a bitonic sorter, which is a network that sorts bitonic sequences. The first stage of BITONIC-SORTER [n] consists of HALF-CLEANER [n], which produces two bitonic sequences of half the size such that every element in the top half is at least as small as each element in the bottom half. Thus, we can complete the sort by utilizing two copies of BITONIC-SORTER [n/2] to sort the two halves recursively.

Fig: The depth D (n) of BITONIC-SORTER [n] is given by recurrence whose solution is D (n) = log n.



Merging Network

Merging Network is the network that can join two sorted input sequences into one sorted output sequence. We adapt BITONIC-SORTER [n] to create the merging network MERGER [n].

The merging network is based on the following assumption:

Given two sorted sequences, if we reverse the order of the second sequence and then connect the two sequences, the resulting sequence is bitonic.

For Example: Given two sorted zero-one sequences X = 00000111 and Y =00001111, we reverse Y to get Y^R = 11110000. Concatenating X and Y^R yield 0000011111110000, which is bitonic.

The sorting network SORTER [n] need the merging network to implement a parallel version of merge sort. The first stage of SORTER [n] consists of n/2 copies of MERGER [2] that work in parallel to merge pairs of a 1-element sequence to produce a sorted sequence of length 2. The second stage subsists of n/4 copies of MERGER [4] that merge pairs of these 2-element sorted sequences to generate sorted sequences of length 4. In general, for k = 1, 2..... log n, stage k consists of $n/2^k$ copies of MERGER [2^k] that merge pairs of the 2^k-1 element sorted sequence to produce a sorted sequence of length2^k. At the last stage, one sorted sequence consisting of all the input values is produced. This sorting network can be shown by induction to sort zero-one sequences, and therefore by the zero-one principle, it can sort arbitrary values.

The recurrence given the depth of SORTER [n]

$$D(n) = \begin{cases} 0 & \text{if} \quad n = 1 \\ D\left(\frac{n}{2}\right) + \log n & \text{if } n = 2^k \quad \text{and } k \geq 1 \end{cases}$$

Whose solution is D (n) = θ ($\log^2 n$). Thus, we can sort n numbers in parallel in ₒ ($\log^2 n$) time.