Master Thesis

Javier González García

February 2021

1 Introduction

A sorting network is a formal representation of a sorting algorithm that for any inputs generates monotically increasing outputs. A sorting network is formed by n channels each of them carrying one input, which are connected pairwise by comparators. A comparator compares the inputs from it's 2 channels and outputs them sorted to the same 2 channels. A comparator network is a sorting network if for any input sequence the output is always the sorted sequence. What makes sorting networks special is their high parallelization capacity, we can create parallel layers of comparators as long as none of them is part of the same input channel at once.

The search of optimal size sorting networks (that means with the lowest number of comparators) involves to test all comparator networks of a giving size. For example for proving the optimality of the sorting network with 11 inputs and 35 comparators we should consider 55 = (11 x 10)/2 possibilities to place each comparator in 2 out of 11 channels. Therefore the search space is of $55^{35} \approx 9 * 10^{60}$ comparator networks.

This problem can only be addressed by using symmetry breaking rules to trim the search space. The method is the same than the one used in [1]with a modification that reduces the search space in a factor of 650 for the 9 inputs problem allowing to solve the 11 and 13 inputs problem in a modest setup. The idea is to incrementally generate comparator Networks by adding a comparator in all possible positions. In the second phase the redundant networks are removed by a prune algorithm.

The method expressed before together with heuristic functions has dealt promising results finding networks of the same size than the state of the art smallest networks in the interval 3-16. In the following chapters I will state with further details the work performed in this master thesis.

2 Representation of comparator networks

A comparator network with n inputs is a sequence of comparators, each comparator is formed by a tuple of 2 channels. We name size k to the number of

comparators the network has. A comparator network is a sorting network if for any n inputs the outputs are the ascending ordered sequence.

To test that a comparator network is a sorting network we should test all the sequences 2^n of 0, 1 this is enough due to the zero-one principle that states that a comparator network orders all sequences in 0, 1 if and only if it sorts all sequences in any ordered set such as the integers set. This way we can test if a comparator network is a sorting network without having to test the n! combinations of sequences.

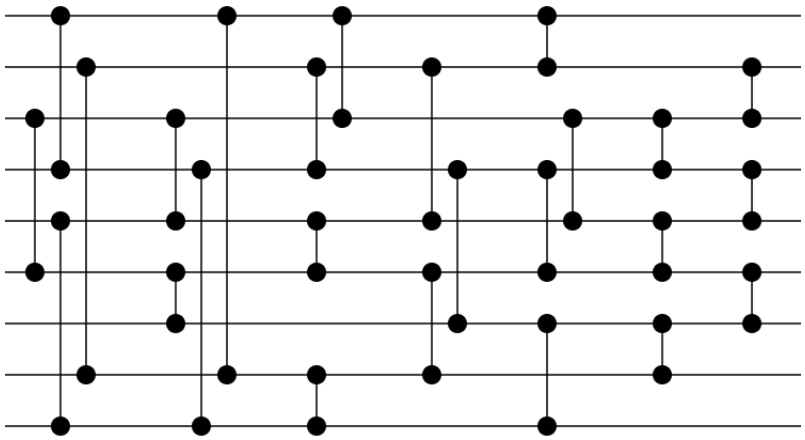

Figure 1: Size 8 sorting network

3 Generate and Prune

4 Generate and Prune Implementation

Algorithm 1 Generate

```
result ← ∅
N ← networks
C ← comparators
for n in N do
  for c in C do
    n′ ← n ⋃ c
    if n′ is not redundant then
      result ← result ⋃ n′
    end if
  end for
  n′ ←
end for
return result
```

Algorithm 2 Prune

```
R ← ∅
N ← networks
for n in N do
  for r in R do
    if r subsumes n then
      subsumed ← true
      break
    end if
    if n subsumes r then
      R ← R \ r
    end if
  end for
  if subsumed is false then
    R ← R ⋃ n
  end if
end for
return R
```

Algorithm 3 Parallel Prune

$N \leftarrow networks$
$C \leftarrow Divide(N)$ {Divide N in as many Clusters as processor}
Each processor performs:
PRUNE(C_i)
for c in C **do**
  Remove($c, C \setminus c$)
end for
return N

Algorithm 4 Remove

$result \leftarrow \emptyset$
$N_i \leftarrow networks$
$N_j \leftarrow networks$
for n_i in N_i **do**
  for n_j in N_j **do**
    if n_i subsumes n_j **then**
      $N_j \leftarrow N_j \setminus n_j$
    end if
  end for
end for
return N_j

5 Subsume Implementations

5.1 Permutations Enumeration

5.2 Bigraph Perfect Matchings

6 Heuristics

7 Conclusion

References

[1] Michael Codish; Luís Cruz-Filipe; Michael Frank; Peter Schneider-Kamp. Sorting nine inputs requires twenty-five comparisons. *Journal of Computer and System Sciences*, 2016.