Research Article No. 6 Parallel And Distributed Computing

ISTS IN BUBBLE SORT NETWORK

Hamza Omer ◦ Ali Naveed ◦ Ariyan Chaudhary

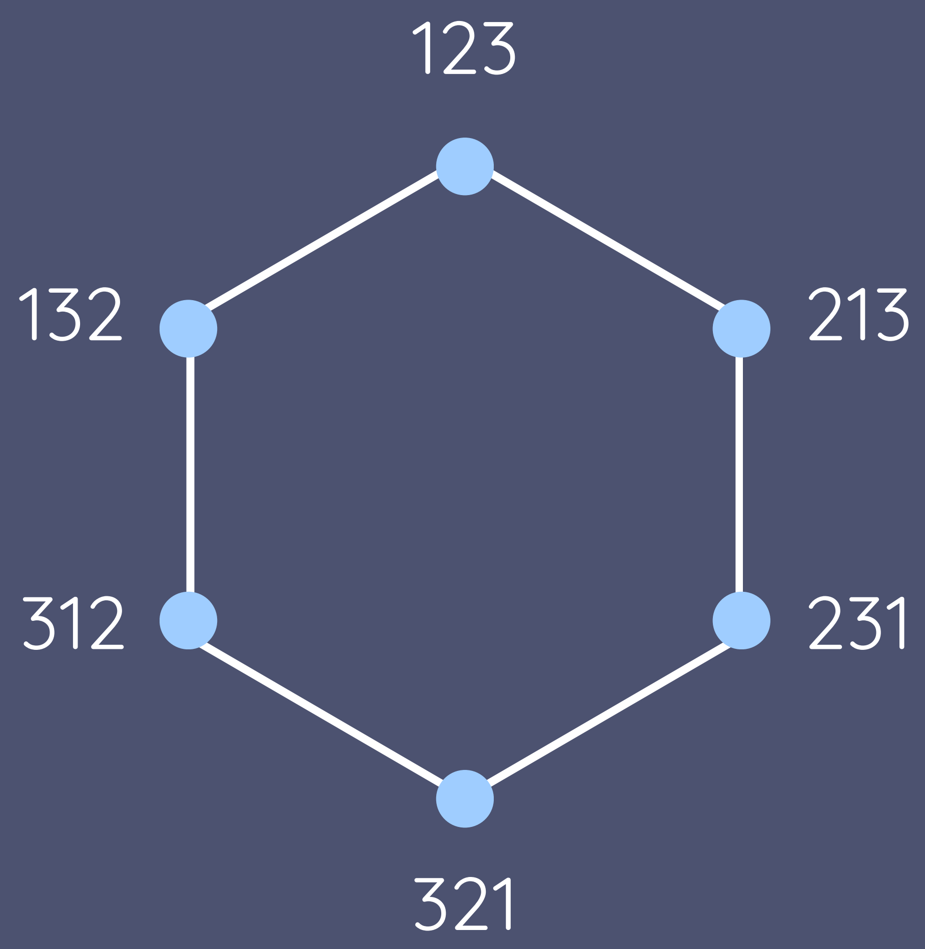
22i-1218 22i-0778 22i-1035

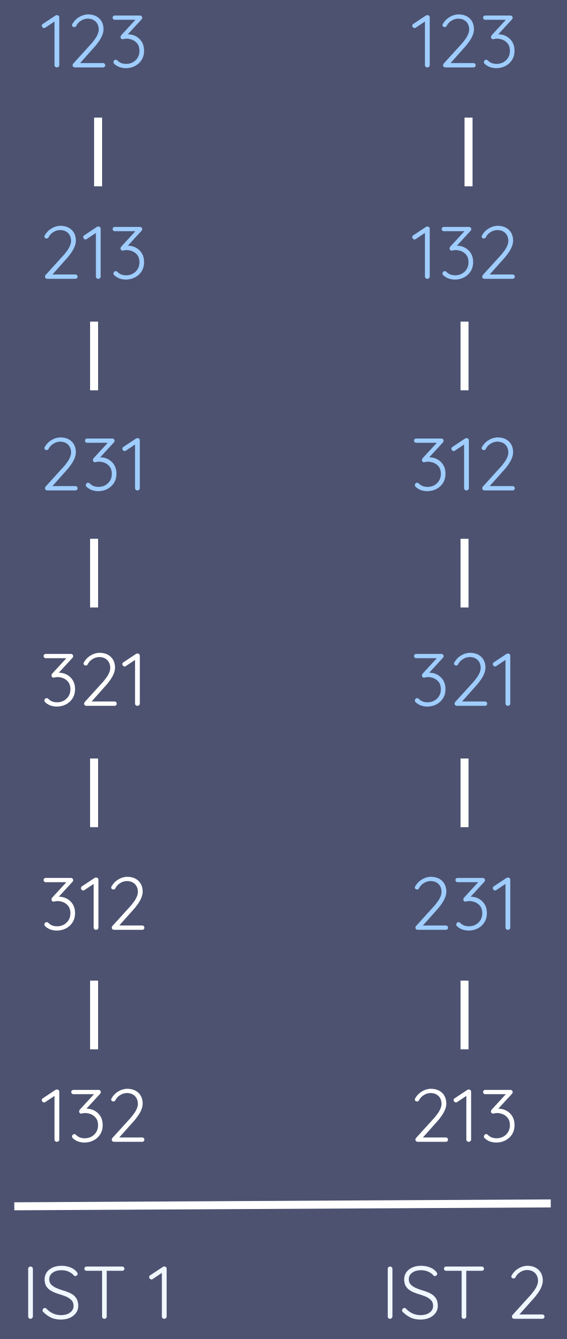
INTRODUCTION OF THE PROBLEM

Bubble Sort Networks

Independent Spanning Trees

Fault Tolerance





Previous work has been done in Kao et al. 2019 - Recursive approach to the algorithm. Cannot be easily parallelized.

All bubble sort networks are vertex transitive.

Why do we need fault tolerance and different independent spanning trees?

Benefits:

- No single point of failure
- Message distributed
- All nodes receive one pkt except destination node
- Secure Message Dist.

Iterative algorithm derived from Kao et al. 2019. Time complexity: $O(n \cdot n!)$ - Asymptotically optimal.

There are two main parts of the algorithm:

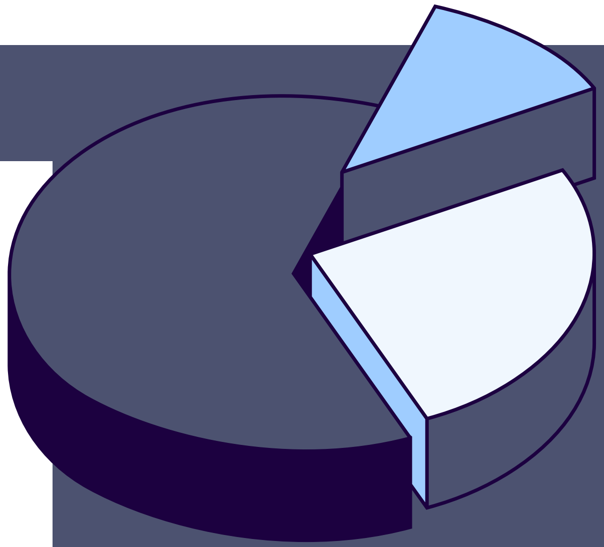
1. Inverse Permutations
2. Swap Elements

Inverse Permutations

This function calculates the parent of a particular node by reversing the permutations performed on it.

Swap Elements

Once the positions of the elements which are in the wrong index is known, they are swapped.



Background

Algorithm

Parallel

Conclusion

OpenMPI will be used to distribute the nodes to individual processors.

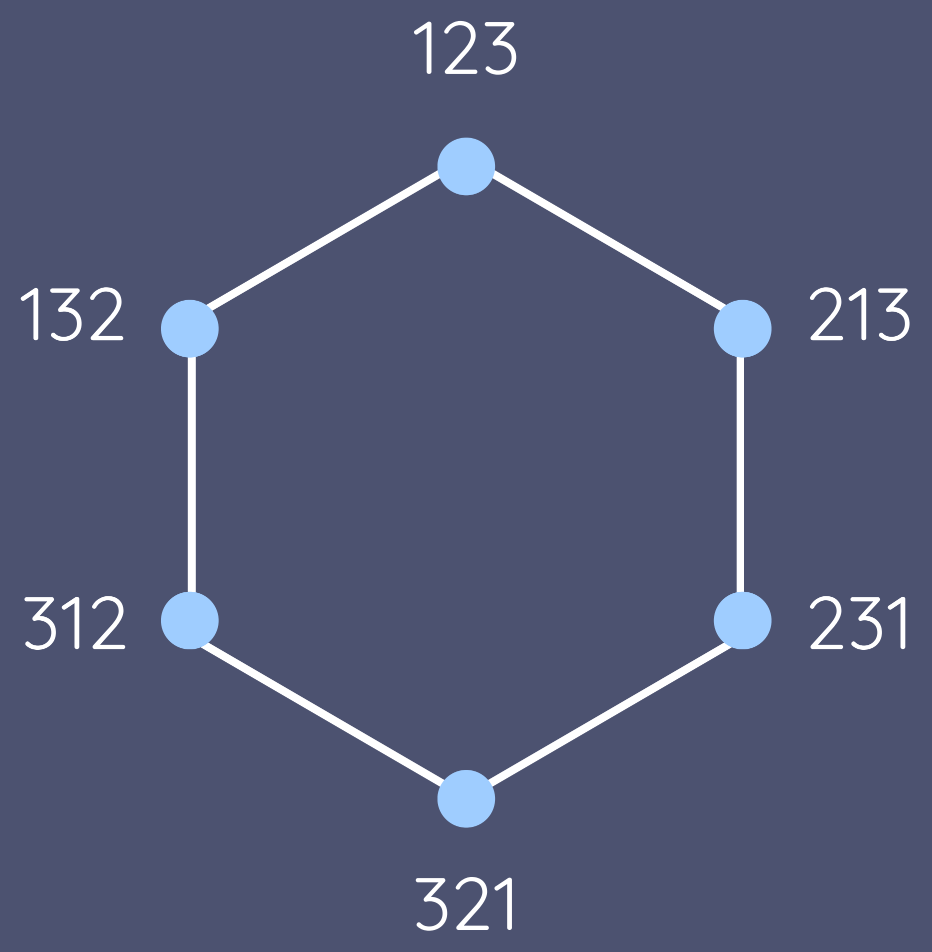
OpenMP will be used to run the workload across those multiple processors and perform the steps in the algorithm.

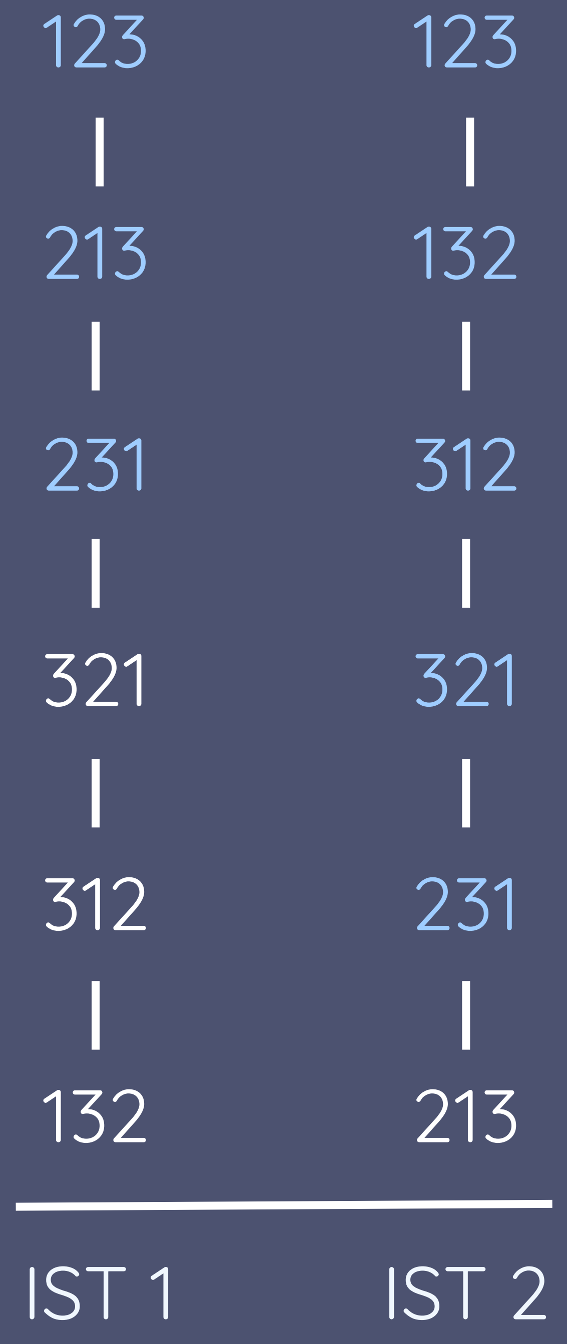
METIS will be used to partition the graph before the distribution of nodes to processors.

Results:

From a provided Bubble Sort Network the program should derive all possible Independent Spanning Trees (ISTs).

Computationally intensive workloads managed by each individual node in the cluster effectively communicating and collaborating with each other.





Research Article No. 6 Parallel And Distributed Computing

THANK YOU FOR LISTENING

Hamza Omer ◦ Ali Naveed ◦ Ariyan Chaudhary

22i-1218 22i-0778 22i-1035