1. Problem

- In a connected, edge-weighted undirected graph, a minimum spanning tree (MST) is a subset of edges that connects all nodes together with the smallest total weights.

- Prim's algorithm is hard to parallelize because each step depends on the sub-graph built previously.

2. Kruskal’s Algorithm

Kruskal's algorithm
- Sort: sort all edges by weights
- Merge: add the smallest edge that does not create a cycle
- Time complexity: $O(e \log e)$ in sorting, and $O(e)$ in merging

3. Serial Algorithm and Profiling

Conclusion:
In large-scale dataset, sorting takes the majority of the time.

4. Proposed Parallel Algorithms

We use OpenMP framework to parallelize sorting via different methods.
- Enumeration Sort (not evaluated)
  - For each value, count the number that is smaller than it
  - Parallel in nature but has $O(n^2)$ complexity
- Parallel Quick Sort
  - Partition the list, and recursively sort two individual partition
  - Parallelize recursive calls via OpenMP tasks
- Sample Sort [3]
  - Select $k$ pivots to partition the dataset, and then sort each partition individually via OpenMP loop parallelism
  - Use oversampling to balance the workload

5. Dataset

The minimal spanning tree of a input graph is decided by the order of edge weights and the graph topology.
- Number of vertices: 40000.
- Weight: generate a random edge permutation by uniformly sampling the weight between 1.0 and 10.0.
- Topology:
  - Random graph: For any two distinguished vertices $u$ and $v$, edge $(u, v)$ is in the graph with probability $p \in [0.01, 0.05, 0.1]$.
  - Sparse graph: For any vertex $u$, $d \in (10, 50, 100)$ distinguished vertices $v$ are connected to $u$. After removing duplicated edges, the degree of each vertex is bounded between $d$ and $2d$.
  - Power-law graph: The vertex degree $d$ satisfies the power-law distribution $p(d) \sim d^{-\gamma}$, $\gamma \in [1.5, 2.3, 3.5]$. Each vertex $u$ is connected with $d$ distinguished vertices $v$ and duplicated edges are removed.

6. Experiments & Results

Experiments are conducted on GHC machine 41 with a 8-cores Intel(R) Xeon(R) E5-1660 v4 @ 3.20GHz processor and 32GB RAM.

### Sorting Algorithm Speedup:

### Parallel Minimum Spanning Tree:

### Speedup for Different Graphs:

### Speedup for Different Graphs (Cont'):

---