Efficient Execution of MapReduce Applications on Irregular NoC Topology
Abraham Addisie, Meghan Cowan, Milind Furia, Helen Hagos

Irregular Network Topology
- Shrinking transistor size with time
- Increase in susceptibility of NoC to permanent faults
- Irregular topology limits communication and increases execution time.

Load Balancing Algorithm
- Core X's workload = Average hop count between all nodes / Average hop count for Core X
- Factor in topology when assigning workloads
- Cores with poor connectivity get less work

Map Reduce Execution - WordCount
- Histogram doesn't benefit as much due to its low number of router ratios
- Reduce WordCount benefits from reduce phase load balancing due to its high number of unique key-value pairs

Map Phase Analysis
- For compute bound mappers, load balancing improves performance by evenly distributing computation
- Linear Regression

Map Phase Inefficiencies
- Map phase characteristics:
  - # map tasks >> # workers
  - Distributed data across cores
  - Congested links can slow down read requests from remote cores
- Solution: Load balance according to connectivity
  - Assigns each core one map task
  - Task size determined by the core's connectivity
  - Reduces read requests from nodes with higher average hop count
- Histogram's execution time remain the same

Reduce Phase Analysis
- Word Count achieved over 25% execution time improvement
- Word Count benefits from reduce phase load balancing due to its low number of unique key-value pairs

Motivation
- Performance of Broken Mesh Topology vs Mesh Topology
- Broken Mesh vs Mesh
- Execution time (x10e8)
- # cycles
- # unique key-value pairs
- Histogram

Reduce Phase Inefficiencies
- Large number of unique keys
- High all-to-all communication during reduce phase
- Mapper's produce key-value pairs in a hash table resident to their local caches
- Reducers collect equal partition of key-value pairs from a hash table
- Less connected reducer nodes take more time to reduce
- Hash table partitioned based on reducer nodes connectivity

Conclusions and Future Work
- Conclusions
  - Irregular network topologies can cause inefficient execution of data-intensive applications.
  - Load balancing based on a node's connectivity can improve performance in a broken mesh.
- Future Work
  - Evaluate load balancing algorithm for more workloads.