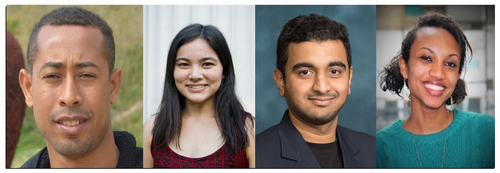


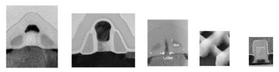
Efficient Execution of MapReduce Applications on Irregular NoC Topology

Abraham Addisie, Meghan Cowan, Milind Furia, Helen Hagos

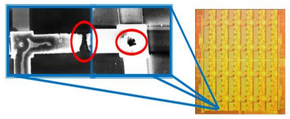
UNIVERSITY OF MICHIGAN

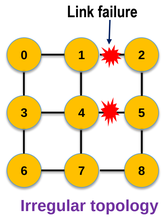
EECS 578 Correct Operation for Processors and Embedded Systems — **Fall'15**

Irregular Network Topology



- Shrinking transistor size with time
- Increase in susceptibility of NoC to permanent faults



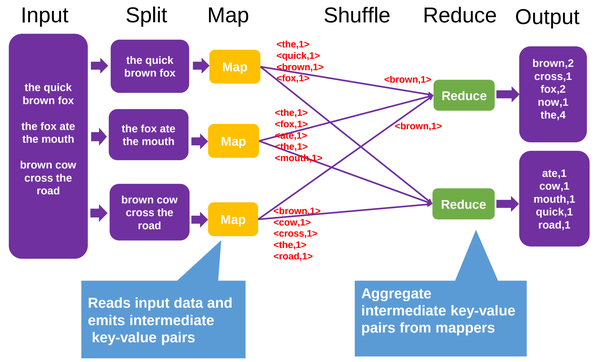


Irregular topology

Irregular topology limits communication and increases execution time.

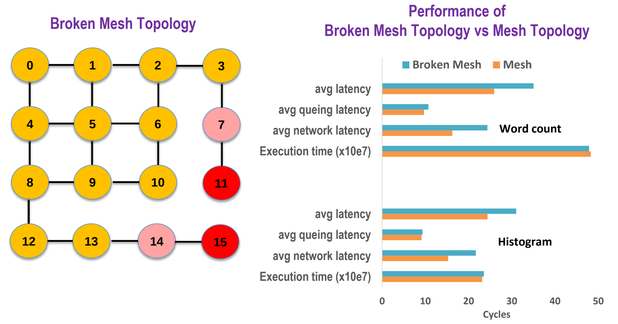
1

MapReduce Execution - WordCount



2

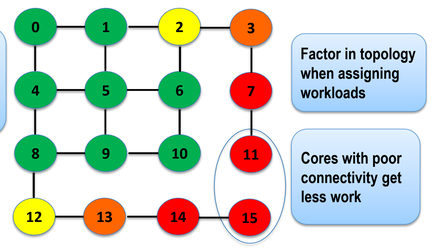
Motivation



Map Reduce applications are sensitive to network topology. Adapt applications by load balancing to avoid bottlenecks

3

Load Balancing Algorithm



Core X's workload ~ Average hop count between all nodes / Average hop count for Core X

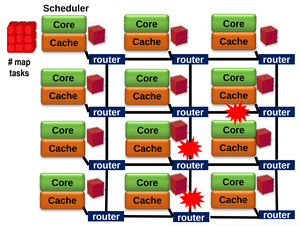
Factor in topology when assigning workloads

Cores with poor connectivity get less work

Average hop count for node 0 = ((hop count between 0 and 1) + (hop count between 0 and 2) + …) / 9

4

Map Phase Inefficiencies



Map phase characteristics
- # map tasks >> # workers
- Distributed data across cores
- Congested links can slow down read requests from remote cores

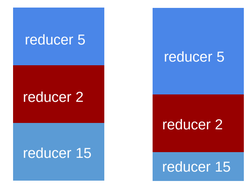
Solution: Load balance according to topology
- Assigns each core one map task
- Task size determined by the core's connectivity
- Reduces read requests from nodes with higher average hop count

5

Reduce Phase Inefficiencies

Large number of unique keys → High all to all communication during reduce phase

Connectivity: reducer 5 > reducer 2 > reducer 15

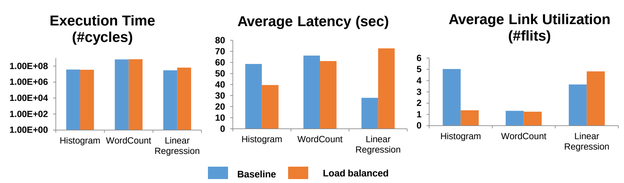


- Mappers produce key-value pairs in a hash table resident to their local caches.
- Reducers collect equal partition of key-value pairs from a hash table.
- Less connected reducer nodes take more time to reduce

Baseline hash table | Load balanced hash table | Hash table partitioned based on reducer nodes connectivity

6

Map Phase Analysis



For compute bound map tasks load balancing hurts performance by unevenly distributing computation.

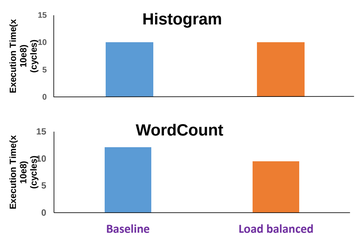
For communication bound map tasks load balancing improves performance by reducing memory read latency.

Map Task Computation vs. Communication Ratio

Linear Regression — Word Count — Histogram

7

Reduce Phase analysis



Histogram's execution time remain the same

Word Count achieved over 25% execution time improvement

- Word Count benefits from reduce phase load balancing due to its high number of unique key-value pairs
- Histogram doesn't benefit as much due to its low number of unique key-value pairs.

8

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 our load balancing algorithm for more workloads.

9