Scalable Load Balancing for MapReduce

Wei Yan*, Yuan Xue*, Bradley Malin*

*ISIS/EECS Department, Vanderbilt University, Nashville, TN, USA
*Department of Biomedical Informatics, Vanderbilt University, Nashville, TN, USA

**OBJECTIVE**

The MapReduce framework has proven to be a powerful and cost-efficient for massively parallel data processing. Load imbalance may diminish the benefits realized through parallelization.

The specific goals of the research:
1. To analyze the load imbalance problem within MapReduce framework.
2. To develop a scalable load balancing solution for various MapReduce-based applications.
3. To demonstrate the efficiency and performance of the proposed solutions using several real applications with real and synthetic datasets.

**REDUCE-PHASE SKEW PROBLEM**

A MapReduce program consists of two primitives, map and reduce.

![Image](https://example.com/image1)

Reduce-phase skew problem happens when a varying number of intermediate key-value pairs are assigned to some reducers, thus skewing the load in the reduce phase.

**SKETCH-BASED PROFILING**

What is Sketch?
- A sketch is a compact, yet powerful, data structure, which is capable of summarizing a substantial quantity of data elements.
- We use a sketch structure to summarize the key group size information.

The Sketch structure.
- Model input data as a vector \( x \) of dimension \( m \).
- Create a small summary as an array of \( w \) in size.
- Use \( d \) hash functions to map vector entries to \( [1,...,w] \).

Properties of Sketch-based Profiling.
- Scalable: \( O(dw) \).
- Efficient: \( O(d) \) for update.
- Robust: stateless.
- Mergeable: combine two sketches by entry-wise summation.

**SKETCH-PACKING ALGORITHM [BigData ’13]**

Sketch Packing algorithm.
- Perform bin packing operation for each row in the sketch.
- Select the row with the optimal performance (in terms of reduce-phase imbalance ratio).

A bounded load balancing performance.
- Load balancing performance bound: \( (2+eR/w) \), with probability at least \( (1-1/e^w) \).
- Here \( e \) is the natural logarithm base, \( R \) is the number of reducers, \( w \) and \( d \) are the width and depth of the sketch.

**Experiments**
- PageRank application with twitter data set (40 million person).
- Inverted indexing application with wikipedia data set (14 million documents).

**SKETCH-DIVISION ALGORITHM [IPCCC ’13]**

Sketch-division algorithms work for highly skewed applications (e.g., record linkage), which cannot be handled by the sketch packing algorithm.

**Cell Block Division algorithm (CB).**
- Divide expensive sketch cells.

**Cell Range Division algorithm (CR).**
- Generate a uniform number of pairs for all reduce tasks.

**Experiments**
- Record Linkage application with DBLP data sets (having publication records from 2.5 million to 50.4 million).

**CONCLUSION**

- Scalability is very important, especially in the era of big data.
- Adopt sketch as a scalable data summary.
- Propose several load balancing algorithms that directly work on sketch.
- Experiments with several applications and datasets verify the efficiency.

**Future work.**
- Experiments with more applications and data sets.
- The integration of sketch and Hive/Pig framework to provide load balancing solutions for general SQL operations.

This research is supported in part by the National Institutes of Health (R01LM009989) and the National Science of Foundation (CCF-0424422).