Novel Optimization of Task Scheduling within MapReduce/Hadoop

Daniel Vicory
Allan Hancock College, Computer Science

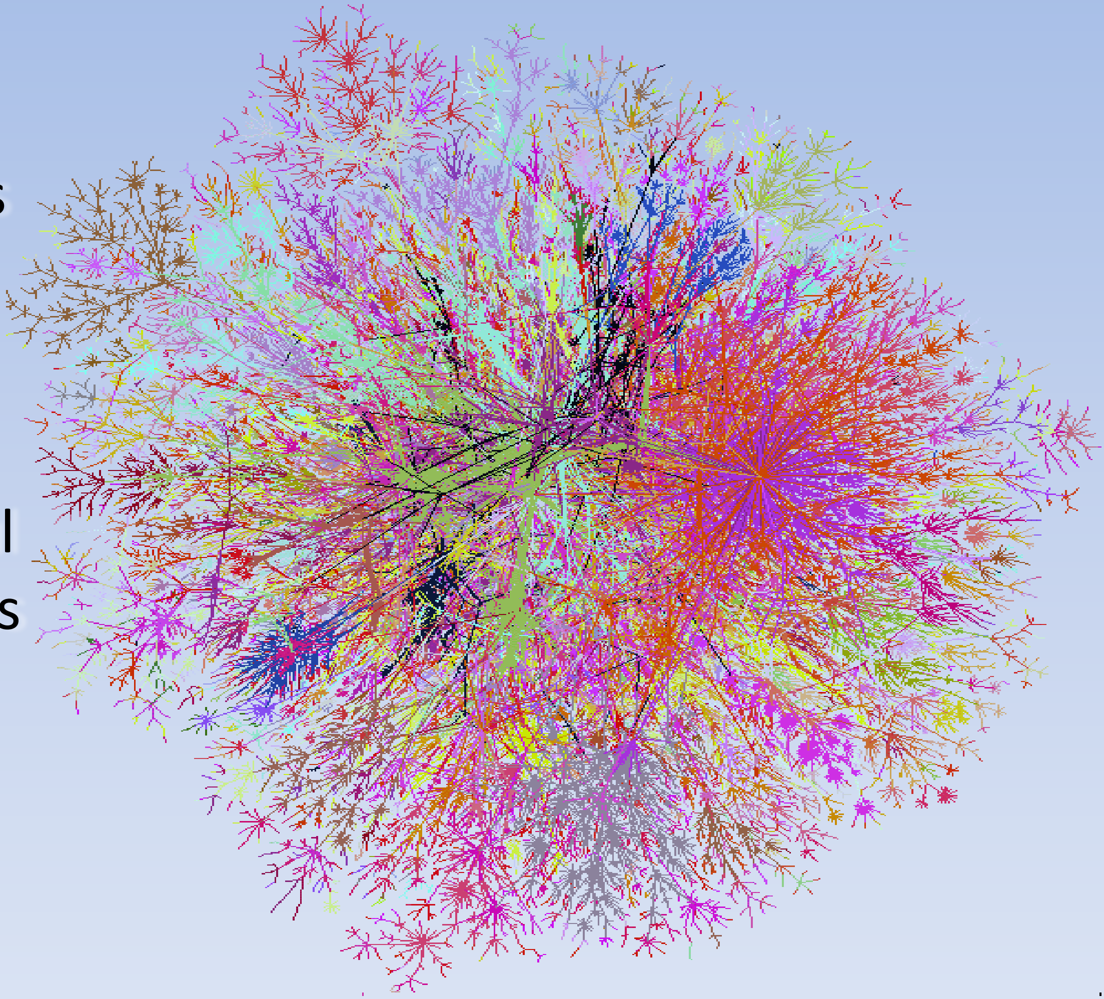
Mentor: Nan Li
Faculty advisor: Prof. Xifeng Yan; University of California, Santa Barbara

Data Mining: Big Picture

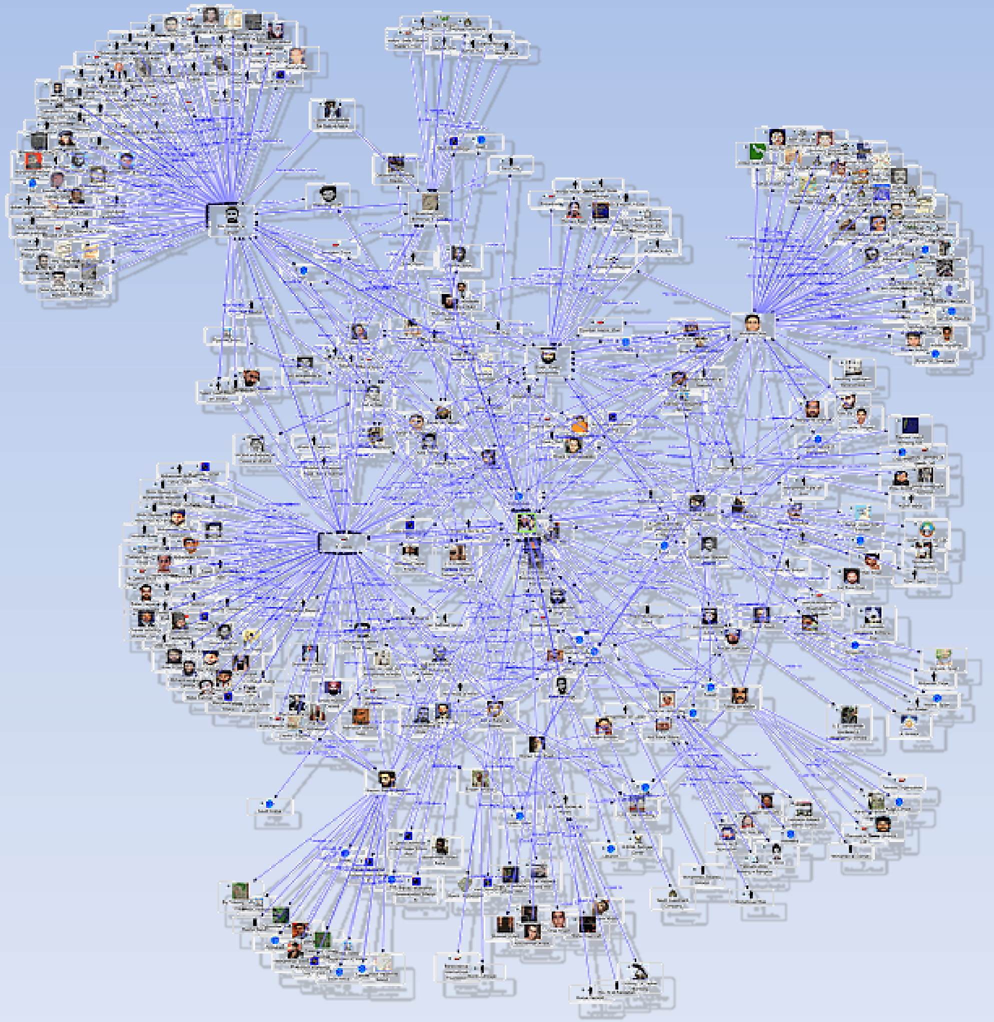
- *Big data* is rampant in fields, data mining helps solve that

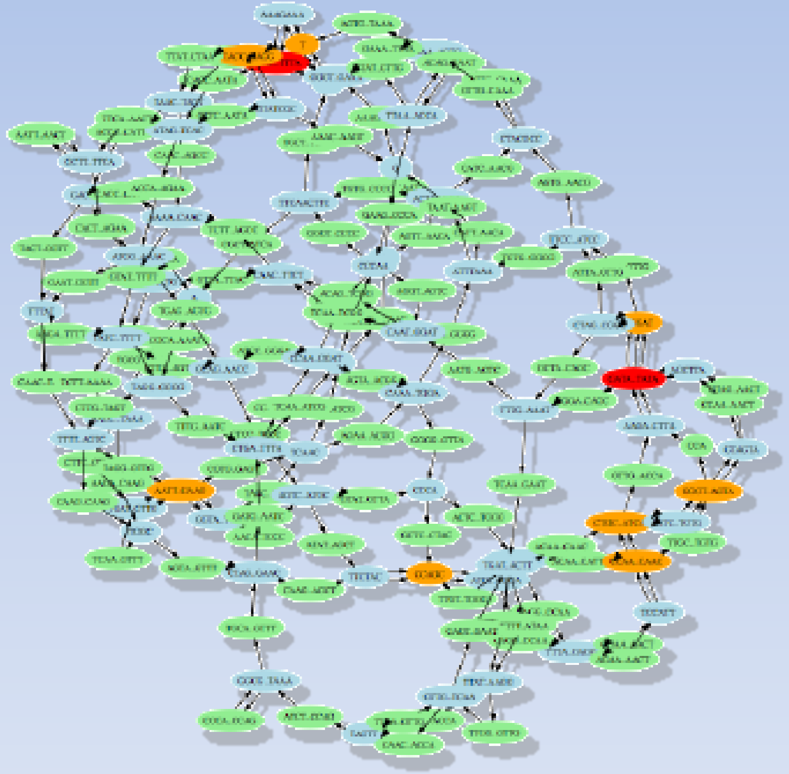
- Process of extracting patterns and meaningful data from large data sets

- Useful for business, research, medicine, etc.



Data Mining Applications



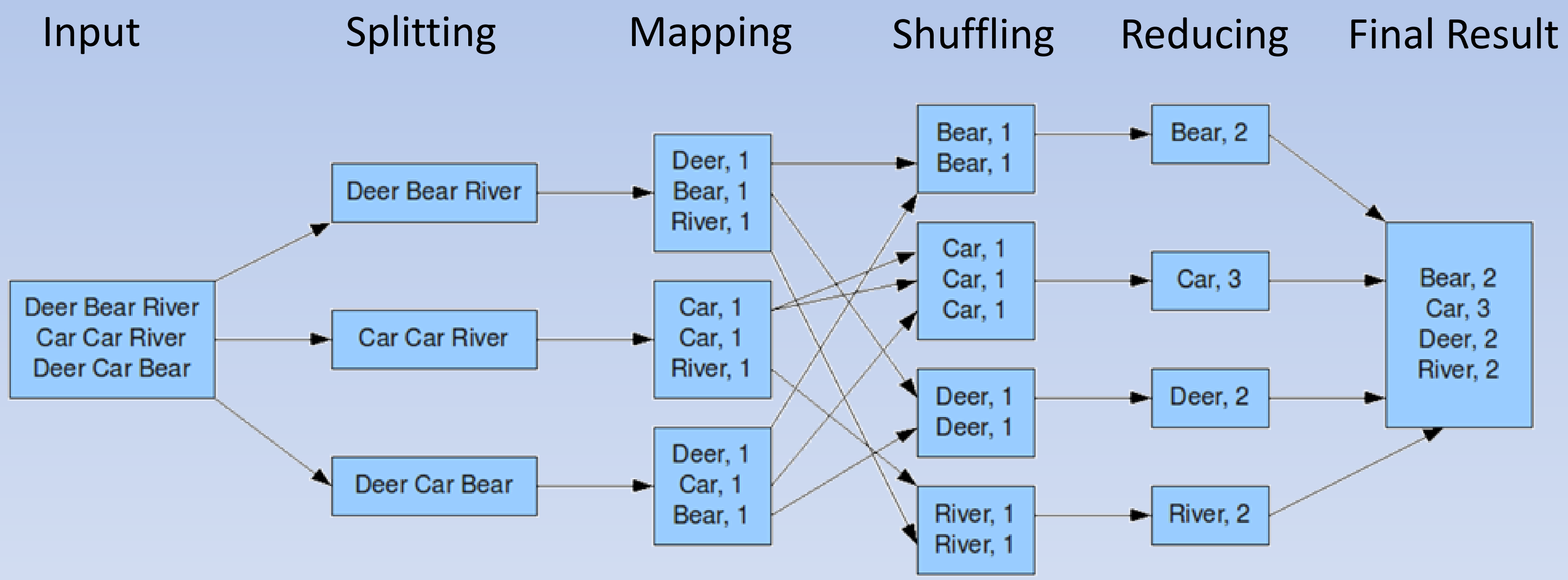


What is MapReduce and Hadoop?

- MapReduce was invented by Google and used to index the web
- Hadoop is software that implements MapReduce
- Map and Reduce refer to the two main steps in algorithm
- MapReduce steps and final results are key-value pairs

1. **map (k1,v1) → list(k2,v2)**
2. **reduce (k2,list(v2)) → list(k3,v3)**

Word Count in MapReduce

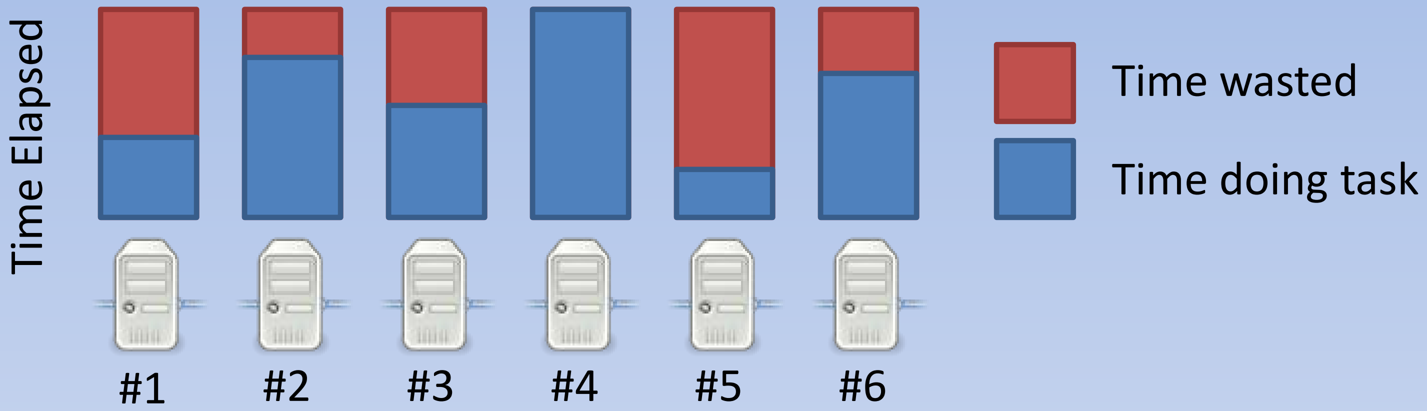


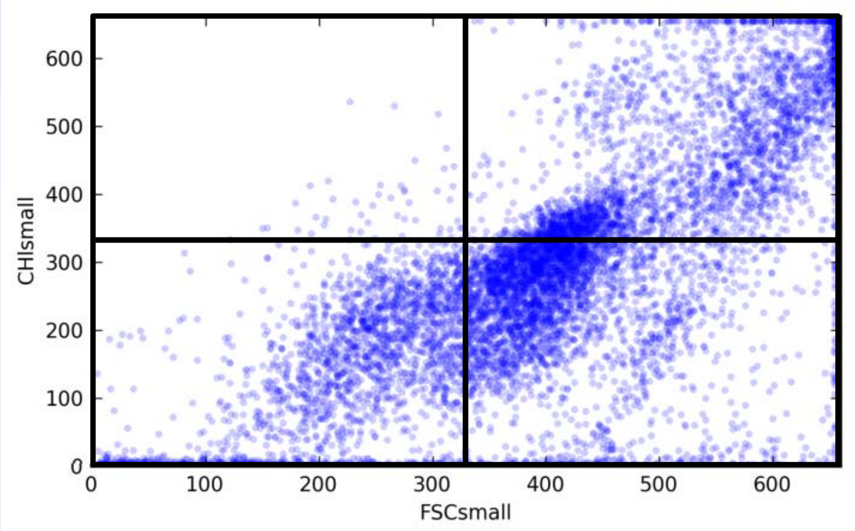
Courtesy of JTeam/Martijn van Groningen
<http://blog.jteam.nl/2009/08/04/introduction-to-hadoop/>

The Problem of Skew

- MapReduce is a sequential algorithm
- Heterogeneous computing environments and non-random datasets can cause each task, or partition of data, to complete at varying times, also known as *skew*
- Skew can mean that a cluster will not be utilized efficiently
- SkewReduce, a framework developed by Washington researchers, solves the issue of skew

Skew Illustrated





Courtesy of Skew-Resistant Parallel Processing […]
YongChul Kwon, Magdalena Balazinksa, Bill Howe, and Jerome Rolia

SkewReduce

- SkewReduce is a framework built on top of Hadoop
- Has an API, which is tied to processing specific types of data
- Also has an optimizer
  - Makes use of cost analysis functions
  - Cost is used to partition data so that each computer finishes its task at about the same time as the rest
- Cost functions require sample data and more programming, not out of the box

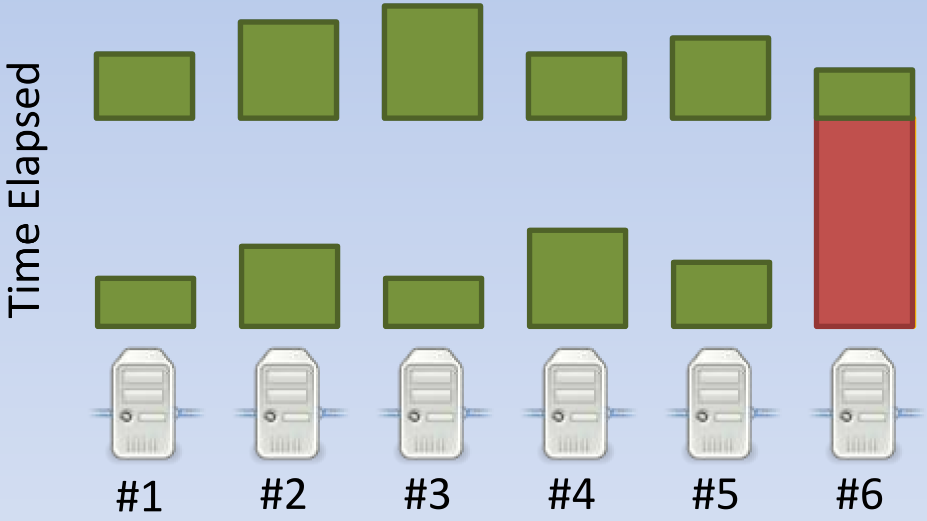
Project Goals

- Setup Hadoop cluster, run SkewReduce
- Work off of SkewReduce as a base
- Leave API alone, remove optimizer
- Implement a task scheduler that does not make use of cost functions or respective sample data
- Compare performance with default “dumb” Hadoop task scheduler and SkewReduce’s optimizer

Our Optimized Task Scheduler

- Novel and clever way of “fast-tracking” tasks to completion
- Does not care about underlying data or algorithm
- Tasks which are deemed to take *too long* in comparison to all other equally-sized tasks on a computer are stopped and split up for rest of the cluster
- Removes the need for cost functions or sample data

Task Scheduler Visualized



🟥 Killed tasks

🟨 Incomplete, running too long, task

🟦 Redistributed task chunks

🟩 Complete task

Hadoop Cluster Performance Tuning

- Test cluster with 8665 books from Gutenberg project, or ~3.2 GB, using word count
- Seven node cluster, Core 2 Duo 2.8GHz, 3GB RAM, and 160GB HD each

Run #	Configuration (each run inherits last configuration)	Runtime
1	8665 separate files, replication 2	1 hrs, 3 mins, 12 sec
2	Compiled single file	3 mins, 30 sec
3	Increase file buffer size	3 mins, 25 sec
4	Turn off speculative execution	3 mins, 20 sec
5	Increase MapReduce memory to 512MB from 200MB	3 mins, 30 sec
6	Increase block size to 128MB from 64MB	3 mins, 21 sec

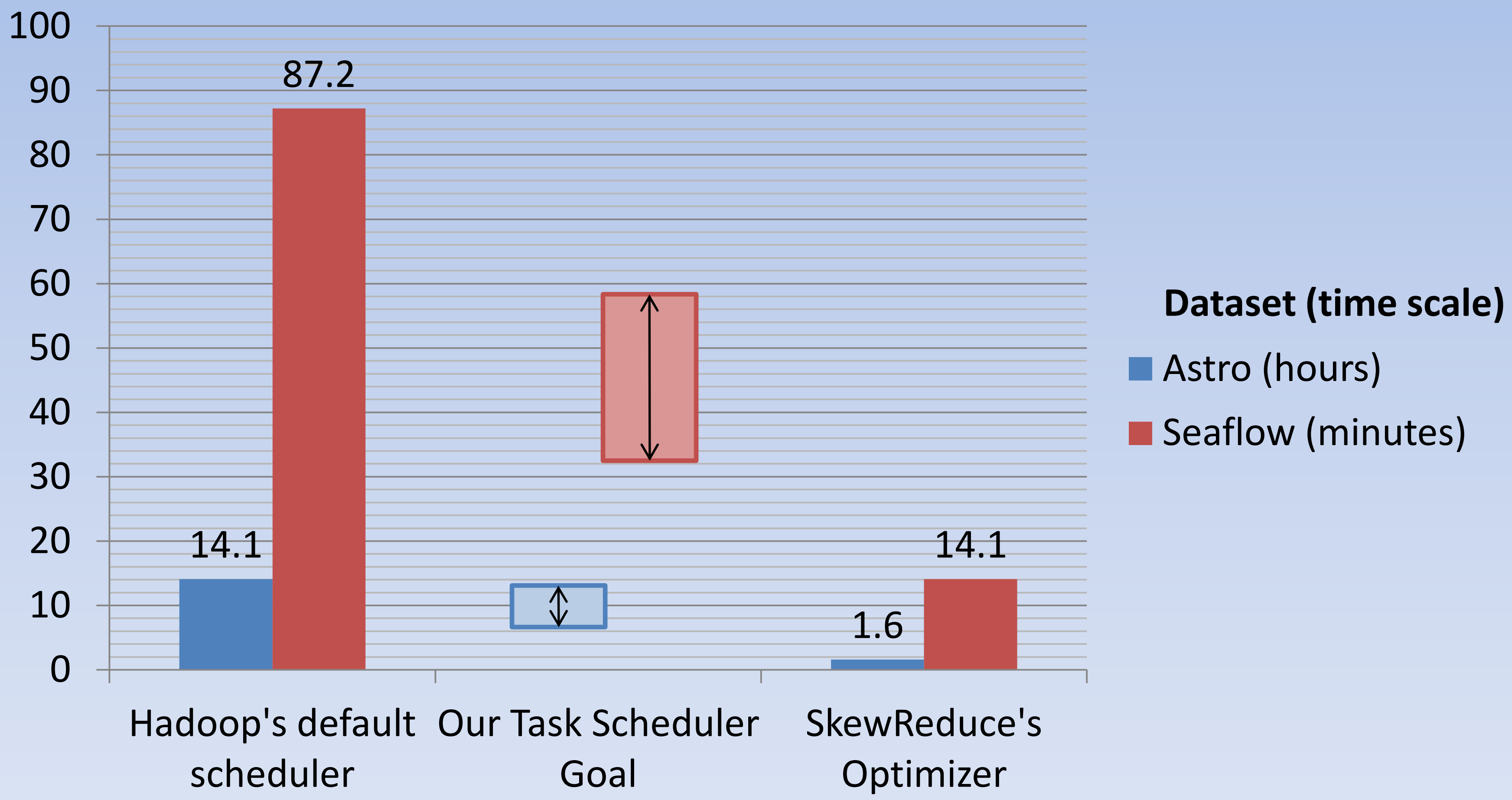
Experimental Methods

- Use SkewReduce's following datasets:

Dataset	Size	# Items	Description
Astro	18 GB	900 M	Cosmology simulation
Seaflow	1.9 GB	59 M	Flow cytometry

- Use SkewReduce's included MapReduce algorithms which identifies clusters of particles
- Benchmark SkewReduce emulating Hadoop behavior, SkewReduce's Optimizer, and our task scheduler with both datasets

Expected Runtime Results



Challenges and Future Work

- Large learning curve for MapReduce, Hadoop, and SkewReduce
- Finish task scheduler
- Ensure task scheduler requires no changes to algorithm or dataset
- Experiment with small variations to task scheduler algorithm to improve upon
- Compare to Hadoop’s scheduler and SkewReduce’s optimizer

Acknowledgements

University of California, Santa Barbara

Mentor: Nan Li

Faculty advisor: Prof. Xifeng Yan

Graduate students: Shengqi Yang

University of Washington

Skew-Resistant Parallel Processing of Feature-Extracting Scientific User-Defined Functions

YongChul Kwon, Magdalena Balazinska