Performance Improvement of MapReduce Applications using Flame-MR

Jorge Veiga, Roberto R. Expósito, Guillermo L. Taboada, Juan Touriño
{jorge.veiga, rreye, taboada, juan}@udc.es

2nd NESUS Winter School & PhD Symposium 2017
Vibo Valentia, Italy
February 22th, 2017
Contents

1 Introduction

2 Flame-MR Design

3 Performance Results

4 Conclusions & Future Work
Introduction

- Big Data has been adopted by many organizations
- Hadoop is one of the most used frameworks
- Limited performance
  - Redundant memory copies
  - Disk overhead
- Existing alternatives must rewrite applications
Big Data has been adopted by many organizations
Hadoop is one of the most used frameworks
Limited performance
  - Redundant memory copies
  - Disk overhead
Existing alternatives must rewrite applications
Introduction

Our proposal: Flame-MR

- Completely new event-driven architecture
- Transparent performance improvement of Hadoop applications
- In-memory computing
- Overlapping of data movement and computation
Flame-MR Design

Main features

- Event-driven architecture
- Efficient memory management
- In-memory sort and merge algorithms
- Support for iterative workloads
- Full compatibility with Hadoop
Flame-MR Design
Flame-MR Design

Worker

DataPool

ThreadPool

DataStructure 2

DataStructure 3

O2
Flame-MR Design
Flame-MR Design

Main features

- Event-driven architecture
- Efficient memory management
- In-memory sort and merge algorithms
- Support for iterative workloads
- Full compatibility with Hadoop
Flame-MR Design

Main features

- Event-driven architecture
- Efficient memory management
- In-memory sort and merge algorithms
- Support for iterative workloads
- Full compatibility with Hadoop
Flame-MR Design

Main features

- Event-driven architecture
- Efficient memory management
- In-memory sort and merge algorithms
- Support for iterative workloads
- Full compatibility with Hadoop
Flame-MR Design

Main features

- Event-driven architecture
- Efficient memory management
- In-memory sort and merge algorithms
- Support for iterative workloads
- Full compatibility with Hadoop
Contents

1 Introduction

2 Flame-MR Design

3 Performance Results

4 Conclusions & Future Work
Performance Results

Testbed configuration

- Evaluations conducted on Amazon EC2
  - 33 i2.4xlarge instances
  - Interconnected via GbE
  - Instance characteristics
    - 2 × 8-core Intel Xeon E5-2670 v2 2.6 GHz
    - 122 GB RAM
    - 4 × 800 GB SSD

- Experiments automated by the Big Data Evaluator tool (BDEv)
  - Configuration of the frameworks
  - Generation of input datasets
  - Collection of results
  - Available at http://bdev.des.udc.es
## Performance Results

### Frameworks
- Hadoop 2.7.2
- Flame-MR 1.0

### Benchmarks

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Characterization</th>
<th>Input data size</th>
</tr>
</thead>
<tbody>
<tr>
<td>WordCount</td>
<td>CPU bound</td>
<td>500 GB</td>
</tr>
<tr>
<td>Sort</td>
<td>I/O bound</td>
<td>500 GB</td>
</tr>
<tr>
<td>Connected Components</td>
<td>Iterative (5 iter.)</td>
<td>40 GB</td>
</tr>
<tr>
<td>PageRank</td>
<td>Iterative (5 iter.)</td>
<td>40 GB</td>
</tr>
</tbody>
</table>
Performance Results

The graph compares the performance of Hadoop and Flame-MR for various tasks:

- **WordCount**
- **Sort**
- **PageRank**
- **ConComp**

The x-axis represents the tasks, and the y-axis represents time in seconds. The chart shows that Flame-MR generally outperforms Hadoop, especially for PageRank and ConComp tasks.
Performance Results

The chart above shows the performance comparison between Hadoop and Flame-MR for different tasks: WordCount, Sort, PageRank, and ConComp. Flame-MR outperforms Hadoop by 65% in PageRank, as indicated by the marked improvement in the chart.
Conclusions & Future Work

Conclusions

- Flame-MR improves transparently the performance of Hadoop
- Results show high performance improvements
  - Up to 65%
- Publicly available at http://flamemr.des.udc.es

Future work

- Development of new features
  - Automatic load balancing
- Evaluation of Flame-MR using real-world use cases
Conclusions & Future Work

Conclusions

- Flame-MR improves transparently the performance of Hadoop
- Results show high performance improvements
  - Up to 65%
- Publicly available at http://flamemr.des.udc.es

Future work

- Development of new features
  - Automatic load balancing
- Evaluation of Flame-MR using real-world use cases
Acknowledgments
Performance Improvement of MapReduce Applications using Flame-MR

Jorge Veiga, Roberto R. Expósito, Guillermo L. Taboada, Juan Touriño

{jorge.veiga, rreye, taboada, juan}@udc.es

2nd NESUS Winter School & PhD Symposium 2017
Vibo Valentia, Italy
February 22th, 2017