2021 OFA Virtual Workshop

RDMA SPARK MEETS OSU INAM: PERFORMANCE ENGINEERING BIG DATA APPLICATIONS ON HPC CLUSTERS

Mansa Kedia, Pouya Kousha, Hari Subramoni, Aamir Shafi, and Dhabaleswar K. (DK) Panda
The Ohio State University
E-mail: shafi.16@osu.edu
https://people.engineering.osu.edu/people/shafi.16
OUTLINE

- Motivation
- Introduction to RDMA Spark and OSU INAM
- Analyzing RDMA Spark Shuffle Stage using OSU INAM
- Usage Scenario:
  - MPI
  - RDMA Spark
- Conclusions and Future Work
BACKGROUND

• Big Data computing frameworks, including Spark and Dask, enable high-performance data science on HPC clusters built with InfiniBand/RoCE interconnects:
  • Recent trend to support accelerators
  • Adding support for low-latency and high-throughput networks like InfiniBand and RoCE

• The programming model for these frameworks enables programmer productivity
  • Hiding low-level communication and I/O details

• It is challenging to analyze the performance of their applications at finer granularity:
  • Such knowledge, in the context of communication, is critical to maximize utilization of underlying network and the application
MOTIVATION

- One popular Big Data computing framework is the Spark framework
  - Java, Scala, and Python APIs to transform and act upon Resilient Distributed Datasets (RDDs)
  - Components include Spark SQL, MLlib, Spark Steaming, GraphX, and others

- Spark web user interface enables monitoring the resource consumption of the Spark cluster. This, however, lacks:
  - Network topology details and in-depth network communication information

- The core idea of this work is to utilize the OSU INAM* framework for monitoring and analyzing network resource usage of the Spark framework on HPC systems

OUTLINE

- Motivation
- Introduction to RDMA Spark and OSU INAM
- Analyzing RDMA Spark Shuffle Stage using OSU INAM
- Usage Scenario:
  - MPI
  - RDMA Spark
- Conclusions and Future Work
THE HIGH-PERFORMANCE BIG DATA (HIBD) PROJECT

- Since 2013
- **RDMA for Apache Spark**
  - MPI4Dask 0.1 (MVAPICH2-GDR based Dask Distributed Library)
  - RDMA for Apache Hadoop 3.x (RDMA-Hadoop-3.x)
  - RDMA for Apache Hadoop 2.x (RDMA-Hadoop-2.x)
    - Plugins for Apache, Hortonworks (HDP) and Cloudera (CDH) Hadoop distributions
- RDMA for Apache Kafka
- RDMA for Apache HBase
- RDMA for Memcached (RDMA-Memcached)
- RDMA for Apache Hadoop 1.x (RDMA-Hadoop)
- OSU HiBD-Benchmarks (OHB)
- [http://hibd.cse.ohio-state.edu](http://hibd.cse.ohio-state.edu)
- Users Base: 340 organizations from 37 countries
- More than 39,250 downloads from the project site
High-Performance Design of Spark over RDMA-enabled Interconnects

- High performance RDMA-enhanced design with native InfiniBand and RoCE support at the verbs-level for Spark
- RDMA-based data shuffle and SEDA-based shuffle architecture
- Non-blocking and chunk-based data transfer
- Off-JVM-heap buffer management
- Support for OpenPOWER
- Easily configurable for different protocols (native InfiniBand, RoCE, and IPoIB)

Current release: 0.9.5

- Based on Apache Spark 2.1.0
- Tested with
  - Mellanox InfiniBand adapters (DDR, QDR, FDR, and EDR)
  - RoCE support with Mellanox adapters
  - Various multi-core platforms (x86, POWER)
  - RAM disks, SSDs, and HDD

http://hibd.cse.ohio-state.edu
Design Features

- Enables high performance RDMA communication, while supporting traditional socket interface
- JNI Layer bridges Scala based Spark with communication library written in native code
- RDMA based shuffle plugin
- SEDA-based architecture
- Dynamic connection management and sharing
- Non-blocking data transfer
- Off-JVM-heap buffer management
- InfiniBand/RoCE support

INAM - InfiniBand Network Analysis and Monitoring tool that can analyze traffic on the InfiniBand network with inputs from the MPI runtime.

- Remotely monitors IB clusters in real time by querying various subnet management entities and gathering input from the MPI runtimes.

- Capability to analyze and profile node-level, job-level and process-level activities for MPI communication.

- Visualize the data transfer happening in a “live” or “historical” fashion for entire network, job or set of nodes.

OSU INAM has been downloaded more than 4,400 times directly from the OSU site (http://mvapich.cse.ohio-state.edu/tools/osu-inam/)
OUTLINE

- Motivation
- Introduction to RDMA Spark and OSU INAM
- Analyzing RDMA Spark Shuffle Stage using OSU INAM

Usage Scenario:
  - MPI
  - RDMA Spark

Conclusions and Future Work
OSU INAM is enhanced to allow monitoring and analyzing RDMA Spark jobs on HPC systems.

The vanilla Spark WebUI provides basic communication information like the volume of data sent and received by Spark workers:
- The UI has no knowledge of the native C code for RDMA communication.

RDMA Spark in conjunction with OSU INAM can provide the following information:
- Summary of all jobs and stages
- Overview of jobs
- Detailed Shuffle information:
  - Amount of data exchange between each pair of executors
  - Time taken for each exchange
- Task-level information
PROPOSED ARCHITECTURE

- Components modified/added
- Components unmodified
OUTLINE

- Motivation
- Introduction to RDMA Spark and OSU INAM
- Analyzing RDMA Spark Shuffle Stage using OSU INAM
- Usage Scenario:
  - MPI
  - RDMA Spark
- Conclusions and Future Work
VIEWING TOPOLOGY USING OSU INAM

- Network view showing 3 heterogeneous clusters at OSC
- Clusters are connected to the same InfiniBand Fabric:
  - 114 switches
  - 1,428 compute nodes
  - 3,402 links

<table>
<thead>
<tr>
<th>View</th>
<th>Average</th>
<th>Min</th>
<th>Max</th>
<th>STDEV_p</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network View</td>
<td>196.15 ms</td>
<td>187 ms</td>
<td>206.09 ms</td>
<td>5.75 ms</td>
</tr>
<tr>
<td>Live Jobs View</td>
<td>18.17 ms</td>
<td>16 ms</td>
<td>20 ms</td>
<td>1 ms</td>
</tr>
</tbody>
</table>
MONITORING MPI JOBS USING OSU INAM

Link utilization by MPI processes

Topology within a node

Communication grid for message exchange between processes

Overview of MPI Usages
Monitoring Jobs Based on Various Metrics

<table>
<thead>
<tr>
<th>Job ID</th>
<th>CPU User Usage</th>
<th>Virtual Memory Size</th>
<th>Total Communication</th>
<th>Total Inter Node</th>
<th>Total Intra Node</th>
<th>Total Collective</th>
<th>RMA Sent</th>
</tr>
</thead>
<tbody>
<tr>
<td>270747</td>
<td>99</td>
<td>8.19 Mb</td>
<td>92.35 Gb</td>
<td>36.69 Gb</td>
<td>55.66 Gb</td>
<td>64.48 Gb</td>
<td>0.00 bytes</td>
</tr>
<tr>
<td>270748</td>
<td>99</td>
<td>15.12 Mb</td>
<td>149.98 Gb</td>
<td>58.23 Gb</td>
<td>91.76 Gb</td>
<td>102.78 Gb</td>
<td>0.00 bytes</td>
</tr>
<tr>
<td>270749</td>
<td>99</td>
<td>30.39 Mb</td>
<td>151.23 Gb</td>
<td>58.35 Gb</td>
<td>92.68 Gb</td>
<td>100.34 Gb</td>
<td>0.00 bytes</td>
</tr>
<tr>
<td>270759</td>
<td>99</td>
<td>17.99 Mb</td>
<td>58.71 Gb</td>
<td>37.29 Gb</td>
<td>21.43 Gb</td>
<td>303.73 Kb</td>
<td>0.00 bytes</td>
</tr>
<tr>
<td>270765</td>
<td>99</td>
<td>9.42 Mb</td>
<td>32.52 Gb</td>
<td>23.19 Gb</td>
<td>9.33 Gb</td>
<td>0.00 bytes</td>
<td>0.00 bytes</td>
</tr>
</tbody>
</table>

Overview of MPI Usages

Profiling and Reporting Performance Metrics at Different Granularities
OUTLINE

- Motivation
- Introduction to RDMA Spark and OSU INAM
- Analyzing RDMA Spark Shuffle Stage using OSU INAM
- Usage Scenario:
  - MPI
  - RDMA Spark
- Conclusions and Future Work
## Jobs Level Summary for RDMA Spark as provided by OSU INAM

<table>
<thead>
<tr>
<th>Job Id</th>
<th>Job Name</th>
<th>Submission Time</th>
<th>Completion Time</th>
<th>Status</th>
<th>Number of tasks</th>
<th>Number of active tasks</th>
<th>Number of completed tasks</th>
<th>Number of failed tasks</th>
<th>Number of active stages</th>
<th>Number of completed stages</th>
<th>Number of failed stages</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>count at SortbyTest.scala:47</td>
<td>2020-11-12T15:56:56.973GMT</td>
<td>2020-11-12T15:57:00.594GMT</td>
<td>SUCCEEDED</td>
<td>48</td>
<td>0</td>
<td>48</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>sortByKey at SortbyTest.scala:14</td>
<td>2020-11-12T15:56:55.318GMT</td>
<td>2020-11-12T15:56:56.949GMT</td>
<td>SUCCEEDED</td>
<td>16</td>
<td>0</td>
<td>16</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>count at SortbyTest.scala:45</td>
<td>2020-11-12T15:56:52.320GMT</td>
<td>2020-11-12T15:56:55.291GMT</td>
<td>SUCCEEDED</td>
<td>16</td>
<td>0</td>
<td>16</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Showing 1 to 3 of 3 rows

## Stages Details

## Stages Level Summary for RDMA Spark Jobs as provided by OSU INAM

<table>
<thead>
<tr>
<th>Stage Id</th>
<th>Stage Name</th>
<th>Submission Time</th>
<th>Completion Time</th>
<th>Status</th>
<th>Input Bytes</th>
<th>Input Records</th>
<th>Output Bytes</th>
<th>Output Records</th>
<th>Shuffle Read Bytes</th>
<th>Shuffle Read Records</th>
<th>Shuffle Write Bytes</th>
<th>Shuffle Write Records</th>
<th>Memory Bytes Spilled</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>count at SortbyTest.scala:47</td>
<td>12 Nov 2020 15:56:59.98 GMT</td>
<td>12 Nov 2020 15:57:00.799 GMT</td>
<td>COMPLETE</td>
<td>0 B</td>
<td>0</td>
<td>0 B</td>
<td>0</td>
<td>4.3 GB</td>
<td>1048576</td>
<td>0 B</td>
<td>0</td>
<td>0 B</td>
</tr>
<tr>
<td>2</td>
<td>filterMap at SortbyTest.scala:34</td>
<td>12 Nov 2020 15:56:56.91 GMT</td>
<td>12 Nov 2020 15:56:56.935 GMT</td>
<td>COMPLETE</td>
<td>2.18 GB</td>
<td>524288</td>
<td>0 B</td>
<td>0</td>
<td>0</td>
<td>4.3 GB</td>
<td>1048576</td>
<td>0 B</td>
<td>0 B</td>
</tr>
<tr>
<td>1</td>
<td>sortByKey at SortbyTest.scala:14</td>
<td>12 Nov 2020 15:56:56.91 GMT</td>
<td>12 Nov 2020 15:56:56.915 GMT</td>
<td>COMPLETE</td>
<td>0 B</td>
<td>0</td>
<td>0 B</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0 B</td>
<td>0</td>
<td>0 B</td>
</tr>
<tr>
<td>0</td>
<td>count at SortbyTest.scala:45</td>
<td>12 Nov 2020 15:56:52.320 GMT</td>
<td>12 Nov 2020 15:56:55.87 GMT</td>
<td>COMPLETE</td>
<td>0 B</td>
<td>0</td>
<td>0 B</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0 B</td>
<td>0</td>
<td>0 B</td>
</tr>
</tbody>
</table>

Showing 1 to 4 of 4 rows
Overview of a single RDMA Spark Job as provided by OSU INAM
RDMA SPARK MEETS INAM: NETWORK USAGE FOR SHUFFLE STAGE

Per Worker Statistics for the Shuffle Stage
Task-level information:
- the launch time
- executor run time
- executor deserialize time
- executor deserialize time
- executor deserialize CPU time
- result serialization time
- GC time
- executor CPU time

Task-level Information for Spark Jobs
OUTLINE

- Motivation
- Introduction to RDMA Spark and OSU INAM
- Analyzing RDMA Spark Shuffle Stage using OSU INAM
- Usage Scenario:
  - MPI
  - RDMA Spark
- Conclusions and Future Work
CONCLUSIONS AND FUTURE WORK

■ Enhanced RDMA Spark and OSU INAM to monitor and analyze the communication traffic on the InfiniBand and RoCE networks

■ The main contribution is to provide a detailed view of the network traffic incurred by the shuffle operation of Spark jobs

■ Users include Big Data applications users, developers, and HPC system administrators

■ Future enhancements:
  • Collect and visualize data transfer during read/write from the underlying HDFS filesystem
  • Allow users to set notifications for any overlap of spark data transfer with other I/O or MPI traffic

■ We plan to publicly release RDMA Spark and OSU INAM enhancements