Introduction to Data Analytics Using Twister2

Kyanie Waters

Introduction

Overtime the amount of data being collected and processed has tripled, which has made it more difficult to manage and analyze. Today streaming and batch data processing are the most important forms of big data analytics and huge systems like Spark [1] and Hadoop [2] were created to process the extreme amounts of data. Although, Spark and Hadoop were developed to handle big data each system has its restrictions as single projects. They are restricted to certain aspects such as data management coordinated together, task management, and communication. Therefore, in real time, there is a need for a big data toolkit that will include all of the many aspects that may be distributed over different data driven applications. Different designs integrated into one major toolkit that can present components to handle volume, velocity, veracity, and variety all in one data driven application. Today's research domain is facing difficult challenges dealing with different data analyzing applications and frameworks. The reason being that certain components in each application or framework may not be in all, so some may lack main pieces such as cost efficiency, energy consumption, and the least amount of time being taken to finish.

Research Questions

The following displays some research questions Twister2 [3] is trying to answer:

1. How efficiently can we analyze data sets?
2. Can a better framework be created that includes both batch and streaming analytics?
3. How to provide HPC capabilities to big data analytics systems?

Methodology

Below explains the tasks given during this research:
1. Used Twister2 to run experiments working with different MPI operations such as "reduce".

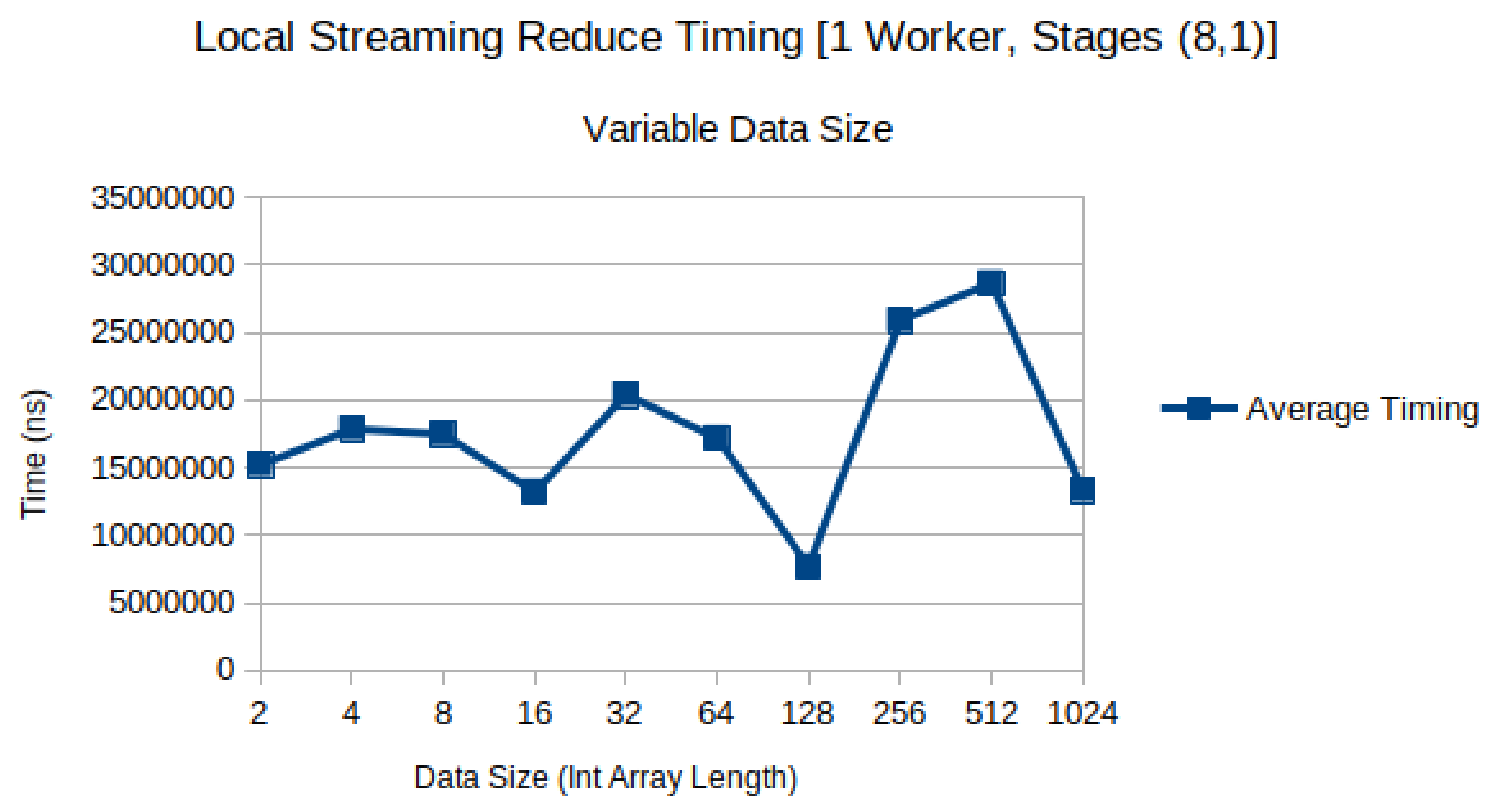
2. Created Charts to show the results from the experiments ran in twister2.

3. Generated a class that consisted of three subjects to then find the max score of subject1, the average score of subjects 1-3, and who was at the top of the class using MPI and Twister2.

Results

The following highlights the results provided by the First and Third task given:
https://github.com/KyanieWaters/Twister2-Project

The following highlights the results provided by the second task given below. It shows data from running experiments that work with one processor, the "reduce" MPI operator, and a data range of 2-1024.



Conclusion

In conclusion, Twister2 can have a huge impact on data analytics in the world today. From the experiment created, Twister2 provides just the right data analytic options in order to get the job done and provide some of the best result. Consequently, it has an absence of user APIs that most enjoy from many other data analytic frameworks. Twister2 can become a framework that works well with both batch and streaming analytics, and with an enormous amount of data it can efficiently analyze data sets better than other data analytic frameworks. Although, it is yet to attract the Dev community generously, it also provides a well developed version of how to combine HPC capabilities with big data [4] analytic systems.

References

[1] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, I. Stoica, "Spark: Cluster computing with working sets" in Proceedings of the 2Nd USENIX Conference on Hot Topics in C loud Computing ser. HotCloud'10, Berkeley, CA, USA:USENIX Association, pp. 10-10, 2010.
[2] C. Lam, Hadoop in action, Manning Publications Co., 2010.
[3] Supun Kamburugamuve, Kannan Govindarajan, Pulasthi Wickramasinghe, Vibhatha Abeykoon, Geoffrey Fox, "Twister2: Design of a Big Data Toolkit" in EXAMPI 2017 workshop November 12 2017 at SC17 conference, Denver CO 2017, Published in Concurrency and Computation Practice and Experience, 06 March 2019, https://doi.org/10.1002/cpe.5189
[4] M. D. Assuno, R. N. Calheiros, S. Bianchi, M. A. Netto, R. Buyya, "Big data computing and clouds: Trends and future directions", Journal of Parallel and Distributed Computing, vol. 79-80, no. Supplement C, pp. 3-15, 2015.

Acknowlegements

I would like to Acknowledge and extend my gratitude to the following:
NSF CIF21 IBBS 1443054 and NSF nanoBIO 1720625
Vibhatha Abeykoon, IU
Niranda Perera, IU
Kannan Govindajan, IU
Geoffrey Fox, IU

INDIANA UNIVERSITY
SCHOOL OF INFORMATICS, COMPUTING, AND ENGINEERING