

Abstraction of simulation of neuroimaging processing systems

Dzung Do

May 2019

The amount of data in many fields, especially neuroimaging, is rapidly growing in size and distribution areas. However, the performance of data movements in big data processing engines seems to be a bottleneck in data processing workflows due to the limits of disk and network bandwidths.

To improve the efficiency of storage elements, experiments have been conducted real systems, emulators and simulators. While experimenting on real systems is not reproducible and possible to impact to the operations of those systems, emulators are not suitable for researching behaviors of systems, simulators are preferable due to reproducibility, scalability and configurability. However, many of them are domain specific and, some are packet or block level with high accuracy of simulation but long simulation time. SimGrid is a simulation have been developed to be a versatile simulation with a balance between accuracy and performance, modular design allowing researchers to add custom simulation models. This is a good reason to adopt SimGrid to execute experiments on the performance of storage elements.

In the paper about CCGrid 2015, a simulation model for storage elements has been proposed and developed, which have been added to SimGrid in later versions. The assumption then was checked by experimenting on three clusters with different types of hard drives with different bandwidths. In these experiments, the actions are solely reading and writing from and to hard disk. It seems to be simple and biased when in real big data processing workflows, the data processing processes comprise transformations and actions on data before writing to and after reading from disk, as well as the bandwidth depends on several factors. In addition, the read/write mechanisms using memory and cache on different operating systems may impact the performance. But this assumption can be treated as a baseline for further experiments or can be used for experiments not focusing on performance of storage elements.

In order to evaluate the impact of Big Data processing strategies on neuroimaging processing, Spark was taken in comparison with Nipype. The results show that in-memory computing and data locality obviously improve the storage performance. However, although in-memory is the strategy that boost the performance the most, there is a limitation when apply it to fMRI datasets, or files with small size in general. On the other hand, this performance can be obtained by leveraging page cache, which is engine-independent.

If we can leverage the page cache mechanism, the performance of big data processing will not depend on strategies provided by engines (Spark in this paper). In comparison between big data processing systems namely Dask, Spark, SciDB, Myria, Tensorflow (Comparative Evaluation of BigData Systems on Scientific Image Analytics Workloads), the performance of Dask is better than that of other engines in particular steps. Moreover, Dask is a Python library for parallel computing with the API familiar to basic Python. It is interesting to expect if we can optimize Dask with in-memory computing, data locality, lazy evaluation and page cache.