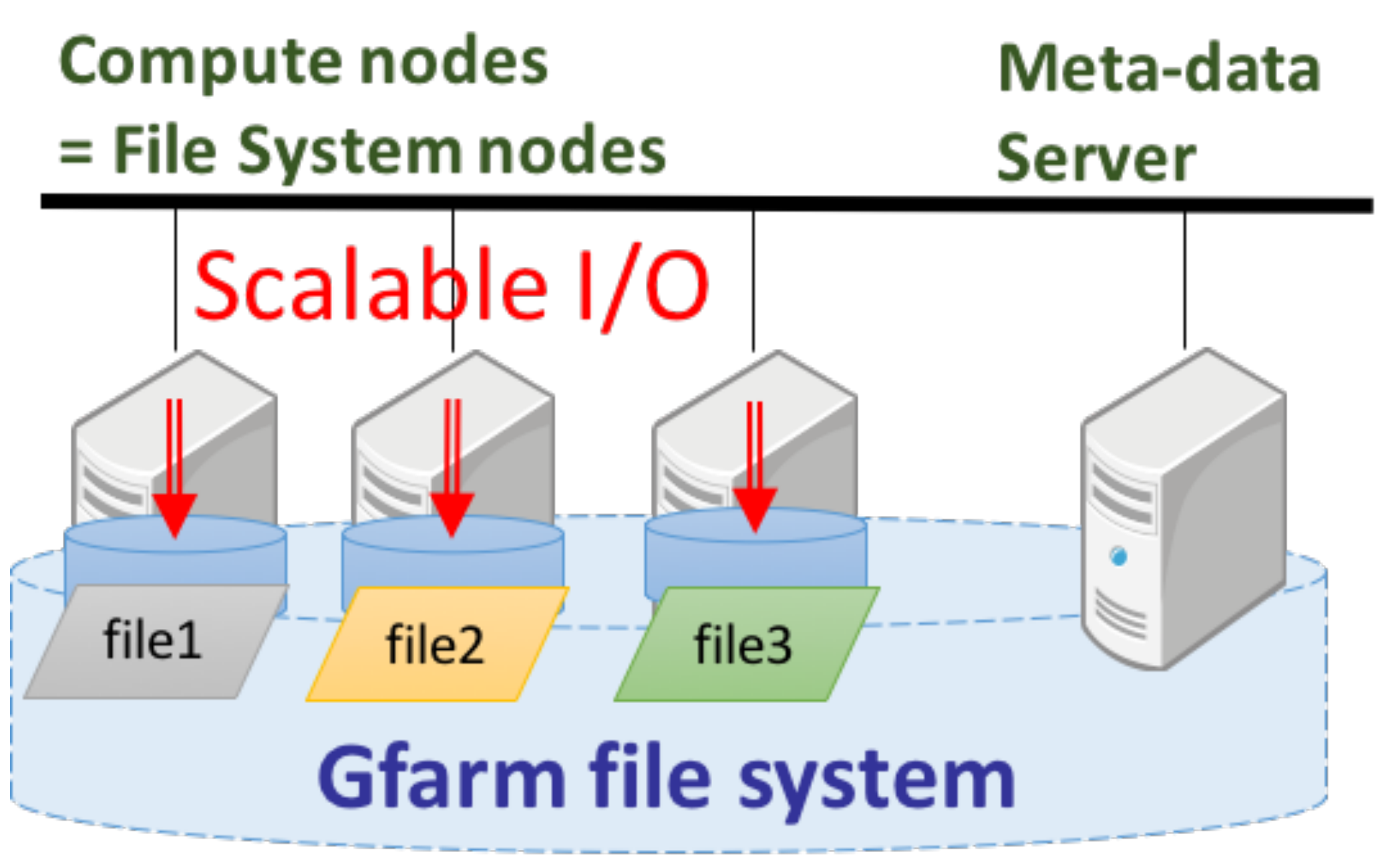
University of Tsukuba | Center for Computational Sciences

Software Researches for Big Data and Extreme-Scale Computing

Gfarm: a High Performance Distributed File System for Supercomputing [1] [2]

Gfarm file system is an open source distributed file system. It is designed for both the cluster environment for high performance data analysis, and the geographically distributed environment for global data sharing and archive. Gfarm provides high performance by exploiting parallel I/O, and high availability by leveraging data replication service.

http://oss-tsukuba.org/en/software/gfarm



Features include

- Exploit local storage, and data locality for scalable I/O performance
- No single point of failure
- MapReduce, MPI-IO, Pwrake workflow system, Batch queuing system with data locality enhancement
- InfiniBand support
- Data integrity is supported for silent data corruption
- 19,000 downloads since March 2007
- Production systems: 8PB JLDG, 22PB HPCI Storage, etc.

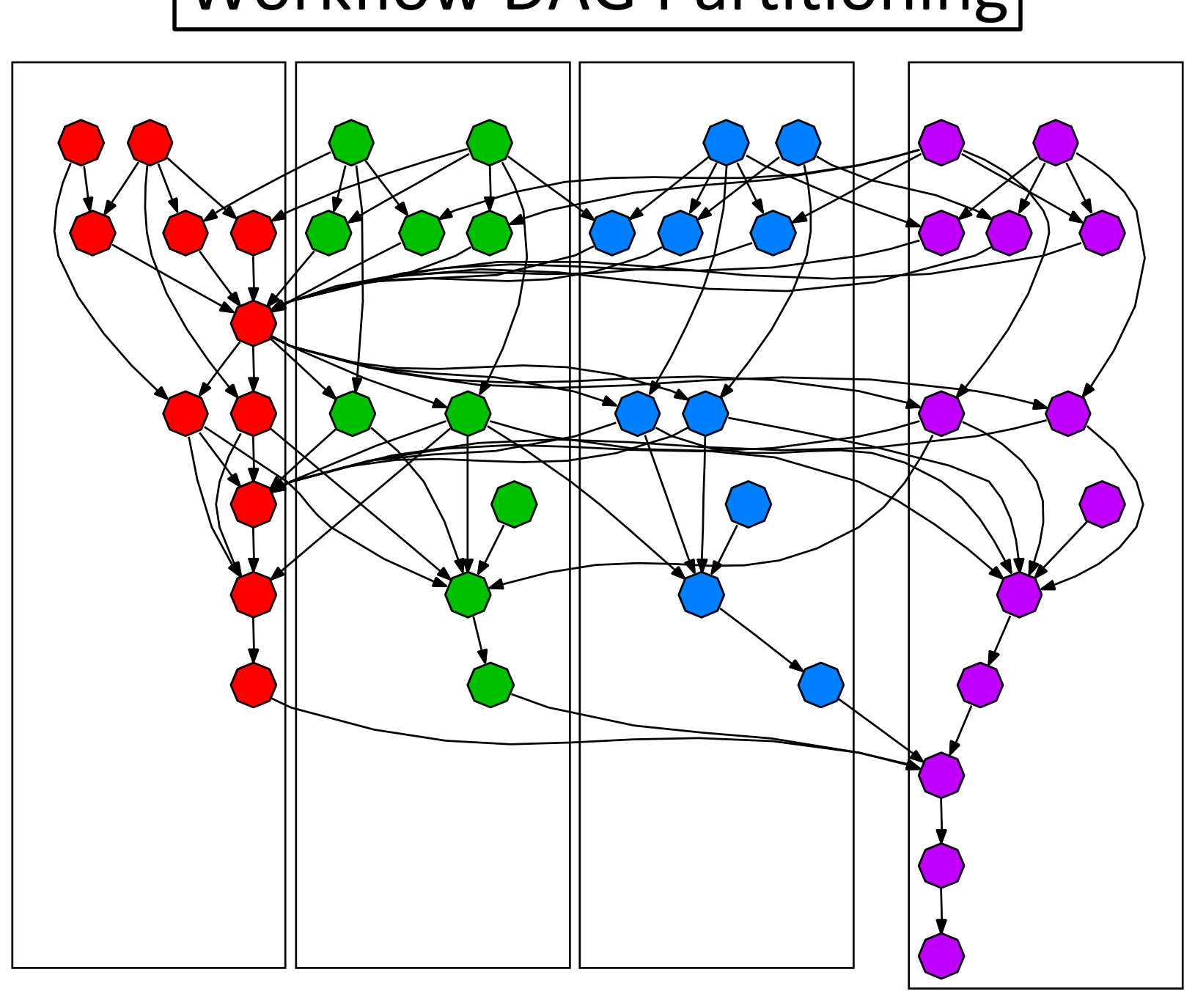
Pwrake workflow system and SMTEF parallel benchmark framework for many-task computing [3] [4]

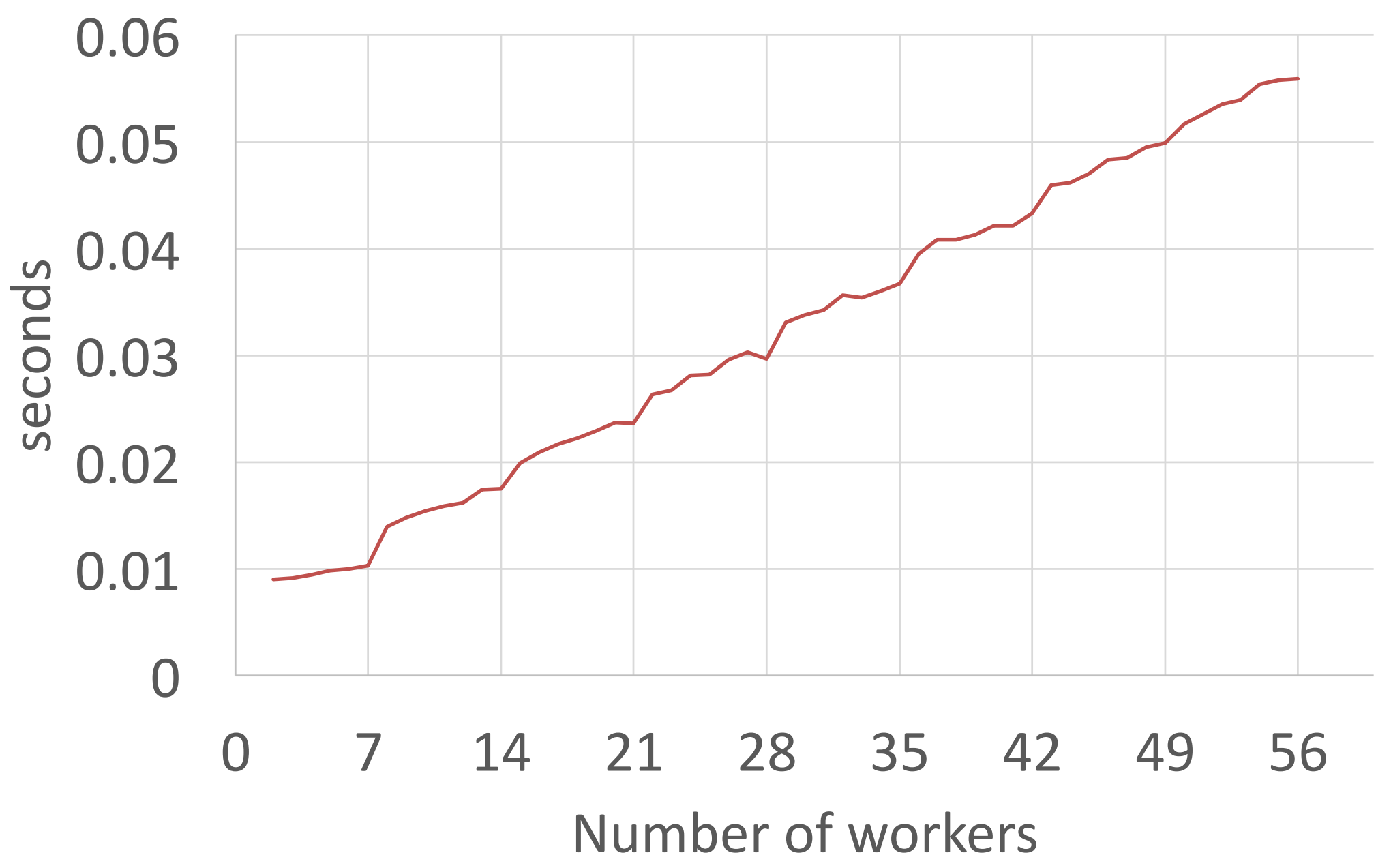
Pwrake is a workflow system for data-intensive science. It provides locality aware scheduling using multi-constraint graph partition to minimize data transfer, and disk cache aware scheduling using LIFO based task queue.

SMTEF is a parallel benchmark framework for many-task computing based on Pwrake. It executes parallel jobs simultaneously to evaluate system maximum throughput

https://github.com/masa16/Pwrake

Workflow DAG Partitioning





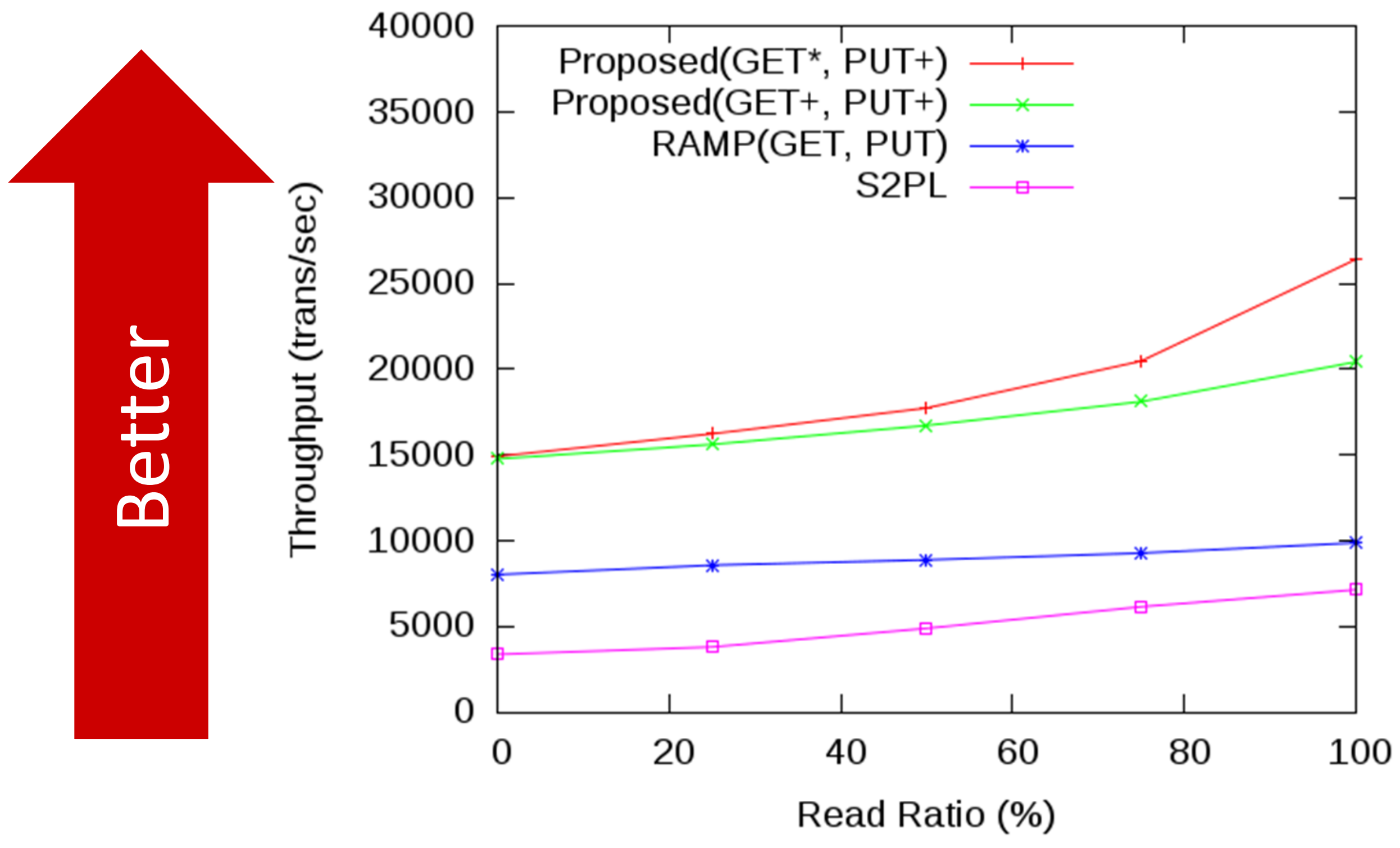
Accelerating Read Atomic Multi-partition Transaction with RDMA [5]

Isolation level
Serializable
Repeatable Read
Read Atomic
Read Committed
Read Uncommitted

1.GET+ operation
-GET with RDMA-Write

2.PUT+ operation
-PUT with RDMA-Write

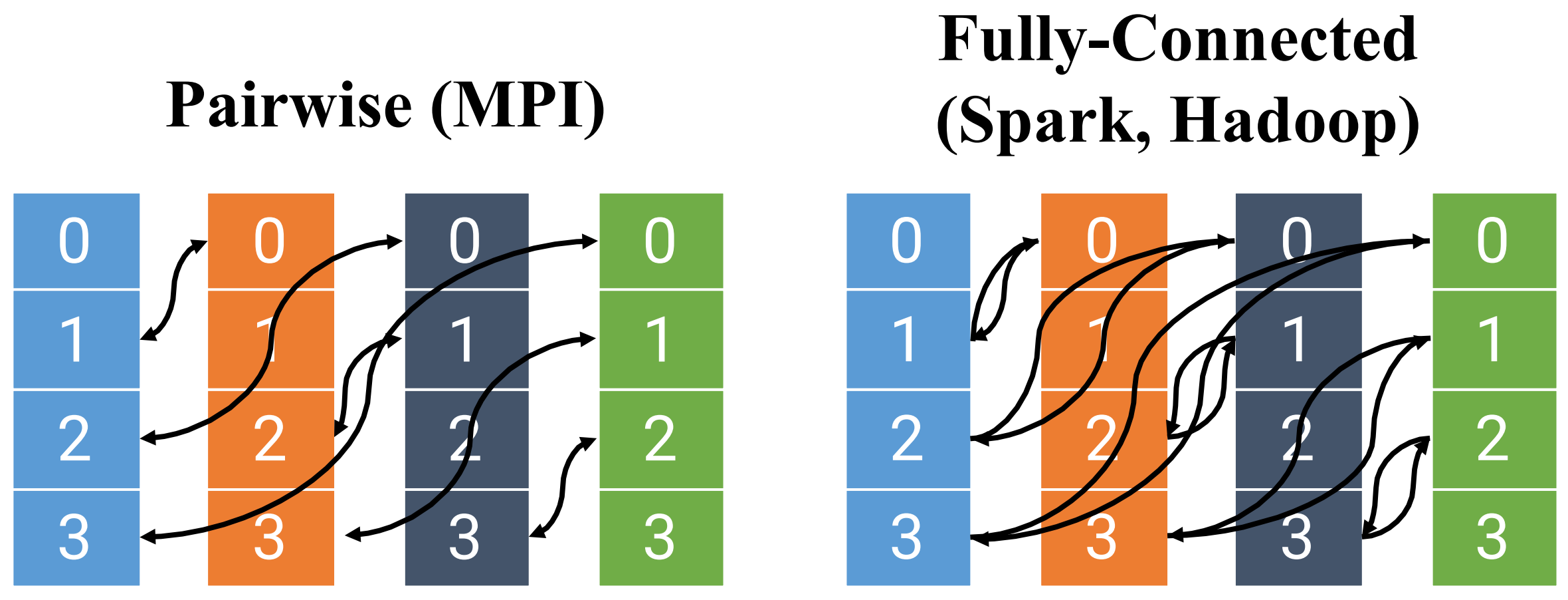
3.GET* operation
-GET+ with RDMA-Read

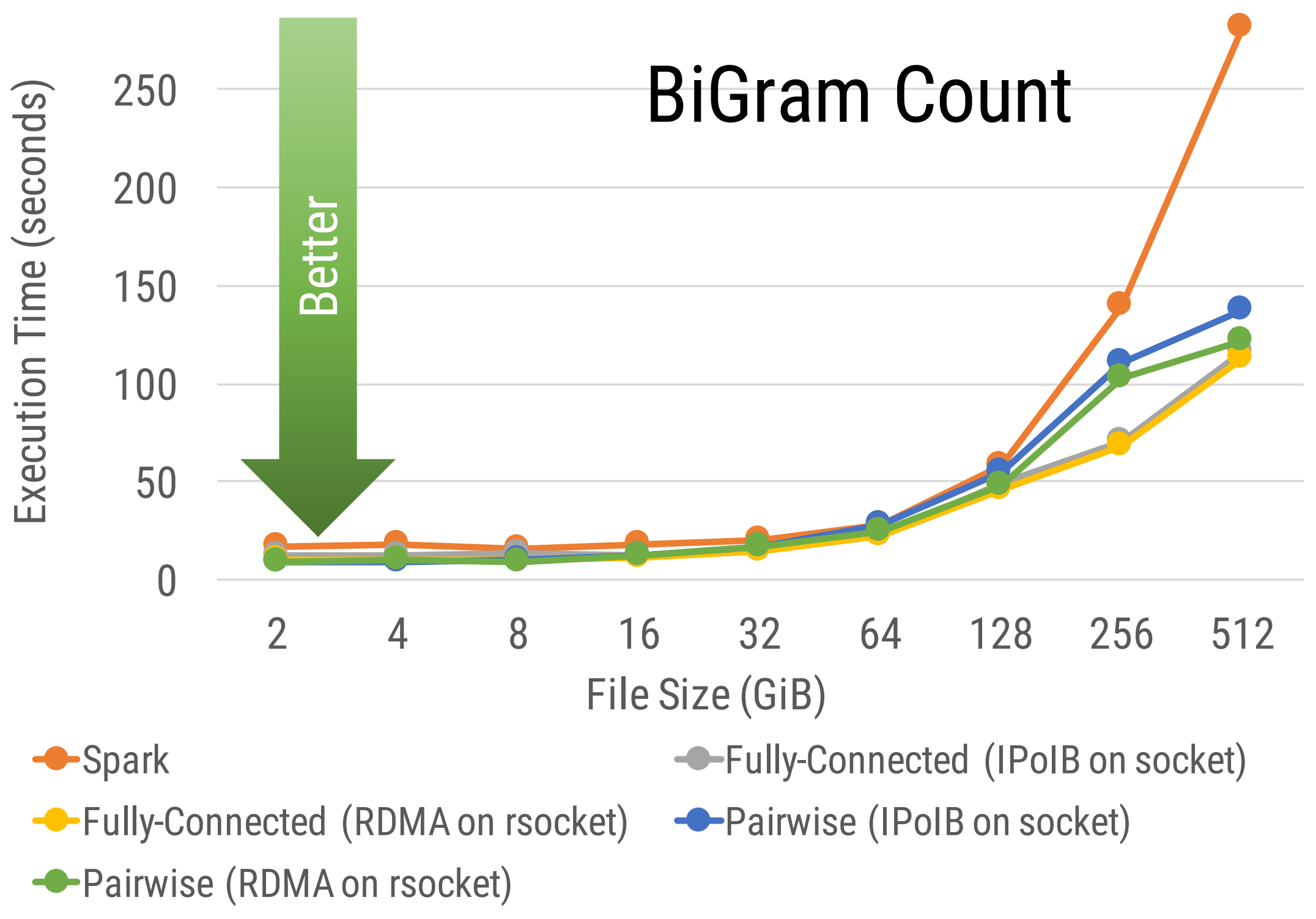


We first present GET+ and PUT+ operations that accelerate the RAMP transaction by exploiting RDMA write operations. We then present the GET* operation, which further accelerates GET+ operations exploiting RDMA read operations. The results of the experiments show that compared with RAMP transactions on IP over InfiniBand, GET* and PUT+ achieve a 2.67x performance improvement on the Yahoo! Cloud Serving Benchmark. All of our code is publicly available.

Acknowledgment
This work is partially supported by JST CREST “System Software for Post Petascale Data Intensive Science”, JST CREST “Extreme Big Data (EBD) Next Generation Big Data Infrastructure Technologies Towards Yottabyte/Year”, JST CREST “Statistical Computational Cosmology with Big Astronomical Imaging Data”, and KAKENHI #16K00150.

On Exploring Efficient Shuffle Design for In-Memory MapReduce [6]





Shuffling, the inter-node data exchange phase of MapReduce, has been reported as the major bottleneck. We compared RDMA shuffling based on rsocket with the one based on IPoIB. We also compared our in-memory system with Apache Spark. Our system demonstrated performance improvement by a factor of 2.64 on BiGram Count as compared to Spark. We conclude that it is necessary to overlap map and shuffle phases to gain performance improvement.

Reference

[1] Osamu Tatebe, Kohei Hiraga, Noriyuki Soda, "Gfarm Grid File System,” New Generation Computing, Ohmsha, Ltd. and Springer, Vol. 28, No. 3, pp.257-275, 2010.
[2] Gfarm File System, http://oss-tsukuba.org/en/software/gfarm
[3] M. Tanaka and O. Tatebe, “Workflow Scheduling to Minimize Data Movement Using Multi-constraint Graph Partitioning,” in 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2012), 2012, pp. 65–72.
[4] M. Tanaka and O. Tatebe, “Disk Cache-Aware Task Scheduling For Data-Intensive and Many-Task Workflow,” in IEEE Cluster 2014, 2014, pp. 167–175.
[5] Naofumi Murata, Hideyuki Kawashima, Osamu Tatebe: Accelerating read atomic multi-partition transaction with remote direct memory access. BigComp 2017: 239-246, Best paper award on big data processing.
[6] Harunobu Daikoku, Hideyuki Kawashima, Osamu Tatebe, “On Exploring Efficient Shuffle Design for In-Memory MapReduce,” BeyondMR workshop, Article 6, 2016.

contact address: pr@ccs.tsukuba.ac.jp

http://www.ccs.tsukuba.ac.jp/