Simulation of Terabit Data Flows for Exascale Applications

Eun-Sung Jung, Rajkumar Kettimuthu, and Venkatram Vishwanath
Argonne National Laboratory
Email: {esjung,kettimut,venkatv}@mcs.anl.gov

***Abstract*—Scientific workflows are increasingly drawing attention as both data and compute resources are getting bigger, heterogeneous, and distributed. Many science workflows are both compute and data intensive and use distributed resources. This situation poses significant challenges in terms of real-time remote analysis and dissemination of massive datasets to scientists across the community. These challenges will be exacerbated in the exascale era.**

A number of data-intensive exascale science workflows will require a terabit/s wide-area network for data movement. For example, next-generation light source experiments are expected to generate data at terabytes per second and will require exascale computing for analysis of this data. Because the operational costs of the exascale system will be high, only a few such systems will be expected and demand for remote data analysis will be high, thus making the wide-area networks and data movement protocols and tools an inherent component of the data-intensive exascale workflows.

However, the modeling and simulation of terabit/s wide-area networks and the associated parallel data flows for exascale science workflows have not received much attention. In this paper, we propose key modeling and simulation functions for wide-area networks and parallel data flows for distributed data-intensive science workflows.

I. INTRODUCTION

Numerous simulation tools have been developed for distributed HPC systems. Such tools include network centric simulators such as OPNET [1], ns2 [2], ns3 [3], and OMNeT++ [4] or higher-level distributed job execution simulators such as CloudSim/GridSim [5]. SST [6], a parallel DES, is being used in several DOE X-Stack projects to help better understand future exascale architectures. SST has an extensible architecture and supports the combined use of OMNeT++ for network models as well as DiskSim [7].

Distributed workflow modeling and simulation tools include basic task models and workload/workflow models describing relationship among tasks. In particular, GridSim provides *Gridlet* objects to define tasks using parameters such as the job length expressed in millions of instructions, disk I/O operations, and the size of input and output files. To the best of our knowledge, however, they all fail to address parallelism in applications and resources, and realistic data transfer simulation factors that are essential for distributed data-intensive workflows and requires taking into account network contention inherent from network topologies. Parallel data transfers are not limited to workflows deployed in wide-area networks. Inside the same data center or computing site, associated tasks such analysis and visualization can exchange data through high-speed interconnection networks. We describe those challenges in more detail in the following sections.

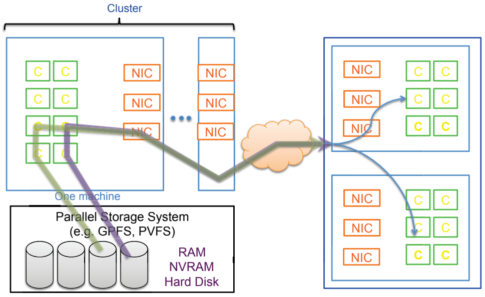

Fig. 1. End-to-end data transfer.

II. CHALLENGES IN DISTRIBUTED, DATA-INTENSIVE WORKFLOW SIMULATION

The performance of distributed data-intensive workflow scheduling depends not only on optimal compute resource allocation but also on efficient data transfers among distributed sites. An understanding of the end-to-end path is critical in order to improve the data transfer performance. The end-to-end path includes both the network and the storage systems, as in Figure 1. We thus need to capture the end-to-end model and be able to integrate such models into current modeling and simulation frameworks.

Network modeling is a challenging task. As the size and capacity of the wide-area networks increase, they become more complex and heterogeneous. Network measurement infrastructure such as perfSONAR [8] has been deployed widely, making it possible to collect metrics such as latency, achievable bandwidth, utilization data on network links, and network topology information. OSCARS [9] service is being enhanced to provide advanced capabilities such as the ability to view current and advanced reservations on the network. These services make it possible to model the networks more accurately. Network models should keep track of availabilities of all the links and provide advance reservation functionalities as OSCARS service currently provides.

Since most data transfers involve disk transfers, it is important to capture the behavior of storage systems as well. Storage is typically a key performance bottleneck in data transfers. File size is an important feature in modeling the file/storage system. The I/O throughput for a dataset consisting of lots of small files varies significantly from the achieved throughput for a dataset consisting of one huge file or a few large files. Additionally, the underlying filesystem, such as Lustre or GPFS, may differ in performance. We need to model file systems as well as disks in order to better simulate data transfer behaviors.

Moreover, data transfers also have diverse characteristics. Parallel flows may be one-to-one, one-to-many, many-to-one, and many-to-many. Many-to-many data transfers usually happen when all the hosts in one cluster send data to the host in the other cluster for the input of next tasks. One flow can be further split into many parallel flows to exploit network parallelism. Time-varying dynamic paths make the simulation more complex and time consuming. Such network path optimization techniques have been proposed but have not been integrated with modeling and simulation frameworks since there have been no proper frameworks considering both networks and workflows.

At the workflow level, the current simulators lack e-Science workload/workflow models. Workflows can have quality-of-service constraints such as latency and jitter. Further, they can have characteristics such as streaming workflow, in which each task in a workflow repeatedly executes for a specified duration of time. Workflow model should be able to capture there characteristics.

Challenges with current simulators include the scale of the system one can capture in the network models and the events, in terms of the number and complexity of the flows, one can simulate. As the number of entities involved in the overall workflow enactment (e.g., the number of disks, network links, and parallel flows) increases, the simulation time increases rapidly. We need a simulation platform that can handle large-scale workflow simulations within a reasonable time.

III. Approach

We will extend the existing parallel simulation platform ROSS/ROSS.Net [10] so that it can simulate both network links and workflow scheduling. This is a good candidate for a parallel simulation platform because it would be able to simulate large-scale networks and many resource entities including disks and CPUs using parallel simulation cores.

First we will formalize a resource model that describes hardware resources and connectivity among them. The types of resources such as storage and CPU will be identified and classified. The network connectivity among those resources can be represented by a graph, and network dynamics will be updated through probing the network links in real time. Each resource type will be further investigated for better modeling and performance metrics. From historical information and live network performance monitoring data, we can build a network topology graph representing connectivity among resources associated with performance metrics. Figure 2 shows that resources and network topology together can be represented by a graph, where a node denotes a resource and an edge denotes a physical network link. We will use our work on an efficient time resource graph model [11] to keep track of past, current, and future resource characteristics. For example, a link is associated with a list of available bandwidths, which contains information on past and current link usage as well as future reservation and forecast.

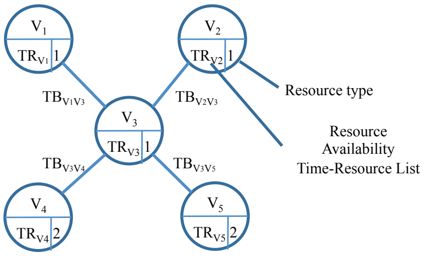

Fig. 2. Resource model.

In addition, we will extend the network graph to include storage to produce an end-to-end resource graph. We will use benchmarks such as IOR [12] and represent the achieved performance metrics as weights for storage. We could successfully develop data transfer optimization algorithms on 100G networks based on the graph model, which describes not only network connectivity but also storage and CPUs. Figure 3 shows that the model-based GridFTP approach could get up to 8 times better performance than does the default GridFTP. This shows the promising future of the end-to-end system modeling for workflow simulation. Our models will be integrated with current simulation frameworks for end-to-end workflow simulation.

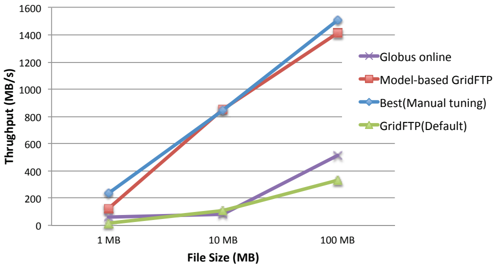

Fig. 3. Performance of model-based parallel data transfer.

References

[1] "Opnet website," http://www.opnet.com/solutions/network_rd/modele

[2] "Ns2 website," http://www.isi.edu/nsnam/ns/.

[3] "Ns3 website," http://www.nsnam.org/.

[4] "Omnet++ website," http://www.omnetpp.org/.

[5] "GridSim: a grid simulation toolkit for resource modelling and application scheduling for parallel and distributed computing." [Online]. Available: http://www.buyya.com/gridsim/

[6] C. L. Janssen, H. Adalsteinsson, and J. P. Kenny, "Using simulation to design extremescale applications and architectures: programming model exploration," *SIGMETRICS Perform. Eval. Rev.*, vol. 38, March 2011.

[7] "Disksim website," http://www.pdl.cmu.edu/DiskSim/.

[8] "perfSONAR network performance monitoring," http://www.perfsonar.net/.

[9] C. Guok, D. Robertson, M. Thompson, J. Lee, B. Tierney, and W. Johnston, "Intra and interdomain circuit provisioning using the OSCARS reservation system," in *3rd International Conference on Broadband Communications, Networks, and Systems*, 2006.

[10] E. Gonsiorowski, C. Carothers, and C. Tropper, "Modeling large scale circuits using massively parallel discrete-event simulation," in *2012 IEEE 20th International Symposium on Modeling, Analysis Simulation of Computer and Telecommunication Systems (MASCOTS)*, 2012, pp. 127–133.

[11] E.-S. Jung, S. Ranka, and S. Sahni, "Workflow scheduling in e-science networks," in *2011 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, Jun. 2011, pp. 432–437.

[12] H. Shan, K. Antypas, and J. Shalf, "Characterizing and predicting the i/o performance of hpc applications using a parameterized synthetic benchmark," in *Proceedings of the 2008 ACM/IEEE conference on Supercomputing*, ser. SC '08. Piscataway, NJ, USA: IEEE Press, 2008, pp. 42:1–42:12.