High-performance Software Stacks for Extremely Large-scale Graph Analysis System

Katsuki Fujisawa
Chuo University
& JST CREST
Tokyo, Japan
fujisawa@indsys.chuo-u.ac.jp

Toyotaro Suzumura
IBM Research
& University College Dublin
& JST CREST
Dublin, Ireland
suzumurat@gmail.com

Hitoshi Sato
Tokyo Institute of Technology
& JST CREST
Tokyo, Japan
hitoshi.sato@gsic.titech.ac.jp

Toshio Endo
Tokyo Institute of Technology
& JST CREST
Tokyo, Japan
endo@is.titech.ac.jp

I. Introduction

The objective of many ongoing research projects in high performance computing (HPC) areas is to develop an advanced computing and optimization infrastructure for extremely large-scale graphs on the peta-scale supercomputers. The extremely large-scale graphs that have recently emerged in various application fields, such as transportation, social networks, cyber-security, and bioinformatics, require fast and scalable analysis (Fig. 1). The number of vertices in the graph networks has grown from billions to trillions and that of the edges from hundreds of billions to tens of trillions (Fig. 2). For example, a graph that represents the interconnections of all the neurons of the human brain has over 89 billion vertices and over 100 trillion edges. To analyze these extremely large-scale graphs, we require a new generation exascale supercomputer, which will not appear until the 2020s, and therefore, we propose a new framework of software stacks for extremely large-scale graph analysis systems, such as parallel graph analysis and optimization libraries on multiple CPUs and GPUs, hierarchal graph stores using non-volatile memory (NVM) devices, and graph processing and visualization systems.

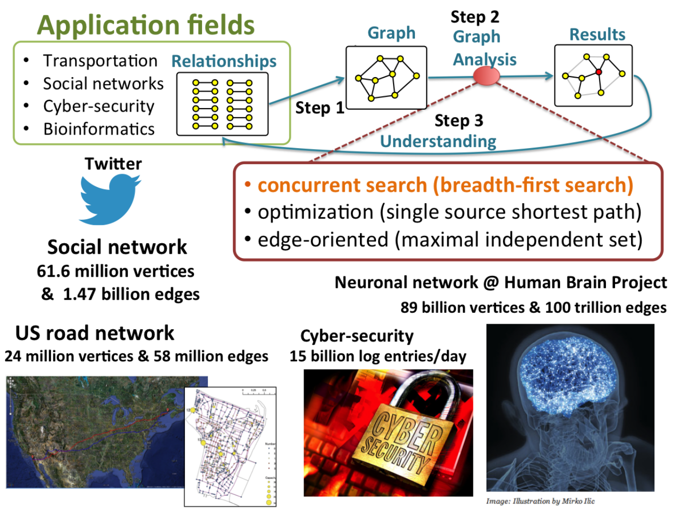

Application fields
• Transportation
• Social networks
• Cyber-security
• Bioinformatics
Relationships
Step 1
Graph
Step 2
Graph
Analysis
Results
Step 3
Understanding
• concurrent search (breadth-first search)
• optimization (single source shortest path)
• edge-oriented (maximal independent set)
Twitter
Social network
61.6 million vertices
& 1.47 billion edges
Neuronal network @ Human Brain Project
89 billion vertices & 100 trillion edges
US road network
24 million vertices & 58 million edges
Cyber-security
15 billion log entries/day
CYBER SECURITY
Image: Illustration by Mirko Ilic

Fig. 1. Graph analysis and its application fields

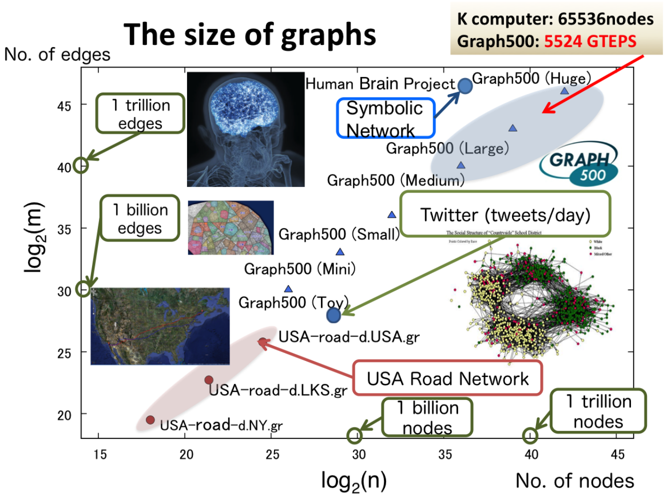
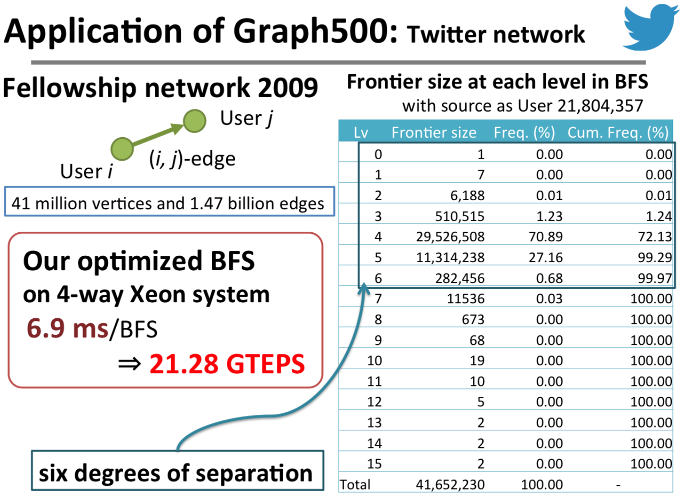


Fig. 2. Size of graphs in various application fields and Graph500 benchmark

II. Graph500 and Green Graph500 Benchmarks

The Graph500 (http://www.graph500.org) and Green Graph 500 (http://green.graph500.org) benchmarks are designed to measure the performance of a computer system for applications that require irregular memory and network access patterns. Following its announcement in June 2010, the Graph500 list was released in November 2010, since when it has been updated semiannually. The Graph500 benchmark measures the performance of any supercomputer performing a breadth-first search (BFS) in terms of traversed edges per second (TEPS). We implemented the world's first GPU-based BFS on the TSUBAME 2.0 supercomputer at the Tokyo Institute of Technology and gained forth place in the fourth Graph500 list in 2012. The rapidly increasing number of these large-scale graphs and their applications has attracted significant attention in recent Graph500 lists (Fig. 2). In 2013, our project team gained first place in both the big and small data categories in the second Green Graph 500 benchmarks. The Green Graph 500 list collects TEPS-per-watt metrics [1]. Our other implementation, which uses both DRAM and NVM devices and whose objective is to analyze extremely large-scale graphs that exceed the DRAM capacity of the nodes, which gained forth place in the big data category in the second Green Graph500 list.



Application of Graph500: Twitter network

Fellowship network 2009

User *j*
User *i*
(*i*, *j*)-edge

41 million vertices and 1.47 billion edges

Our optimized BFS on 4-way Xeon system
6.9 ms/BFS
⇒ 21.28 GTEPS

six degrees of separation

Frontier size at each level in BFS
with source as User 21,804,357

Lv	Frontier size	Freq. (%)	Cum. Freq. (%)
0	1	0.00	0.00
1	7	0.00	0.00
2	6,188	0.01	0.01
3	510,515	1.23	1.24
4	29,526,508	70.89	72.13
5	11,314,238	27.16	99.29
6	282,456	0.68	99.97
7	11536	0.03	100.00
8	673	0.00	100.00
9	68	0.00	100.00
10	19	0.00	100.00
11	10	0.00	100.00
12	5	0.00	100.00
13	2	0.00	100.00
14	2	0.00	100.00
15	2	0.00	100.00
Total	41,652,230	100.00	-

Fig. 3. Application of Graph500 benchmarks

Fig. 3 shows an application of the Graph500 benchmark. We slightly modified the source code for the Graph500 benchmark, which was applied to making a BFS tree of the Twitter Fellowship Network 2009. It takes only about 7 ms to make a BFS tree from a root node, although this graph has 41 million vertices and 1.47 billion edges.

III. High-Performance Computing for Mathematical Optimization Problems

We also present our parallel implementation for large-scale mathematical optimization problems. In the last decade, mathematical optimization programming (MOP) problems have been intensively studied in both their theoretical and practical aspect in a wide range of fields, such as combinatorial optimization, structural optimization, control theory, economics, quantum chemistry, sensor network location, data mining, and machine learning. The semidefinite programming (SDP) problem is a predominant problem in mathematical optimization. We have developed a new version of the semidefinite programming algorithm parallel version (SDPARA), which is a parallel implementation on multiple CPUs and GPUs for solving extremely large-scale SDP problems that have over a million constraints [2], [3]. SDPARA can also perform parallel Cholesky factorization using thousands of GPUs and techniques to overlap computation and communication if an SDP problem has over two million constraints and Cholesky factorization constitutes a bottleneck. We demonstrated that SDPARA is a high-performance general solver for SDPs in various application fields through numerical experiments at the TSUBAME 2.5 supercomputer, and we solved the largest SDP problem (which has over 2.33 million constraints), thereby creating a new world record. Our implementation also achieved 1.713 PFlops in double precision for large-scale Cholesky factorization using 2,720 CPUs and 4,080 GPUs [3].

IV. Software Stacks for Extremely Large-Scale Graph Analysis System

In this paper, we finally propose new software stacks for an extremely large-scale graph analysis system (Fig. 4), which are based on our current ongoing research studies above.

1) Hierarchal Graph Store: Utilizing emerging NVM devices as extended semi-external memory volumes for processing extremely large-scale graphs that exceed the DRAM capacity of the compute nodes, we design highly efficient and scalable data offloading techniques, PGAS-based I/O abstraction schemes, and optimized I/O interfaces to NVMs.
2) Graph Analysis and Optimization Library: Large-scale graph data are divided between multiple nodes, and then, we perform graph analysis and search algorithms, such as the BFS kernel for Graph500, on multiple CPUs and GPUs. Implementations, including communication-avoiding algorithms and techniques for overlapping computation and communication, are needed for these libraries. Finally, we can make a BFS tree from an arbitrary node and find a shortest path between two arbitrary nodes on extremely large-scale graphs with tens of trillions of nodes and hundreds of trillions of edges.
3) Graph Processing and Visualization: We aim to perform an interactive operation for large-scale graphs with hundreds of million of nodes and tens of billion of edges.

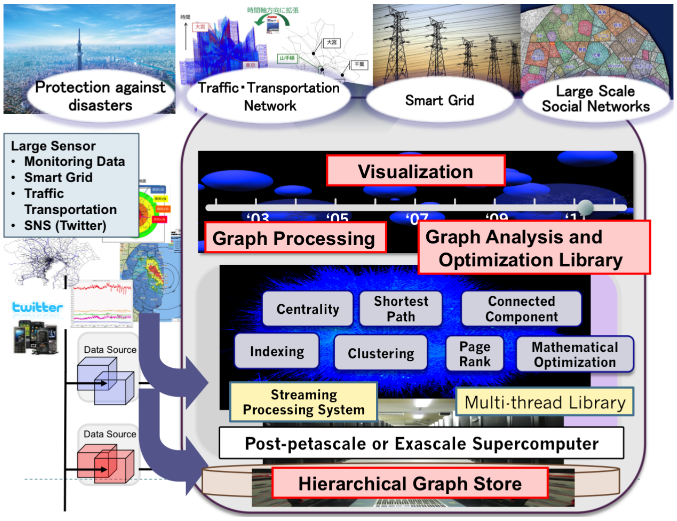


Fig. 4. Software stacks for extremely large-scale graph analysis system

References

[1] Y. Yasui, K. Fujisawa and K. Goto: NUMA-optimized parallel breadth-first search on multicore single-node system, Proceedings of the IEEE 2013 Conference on Big Data (BigData 2013) (2013)

[2] K. Fujisawa, T. Endo, H. Sato, M. Yamashita, S. Matsuoka and M. Nakata: High-performance general solver for extremely large-scale semidefinite programming problems, Proceedings of the 2012 ACM/IEEE Conference on Supercomputing, SC'12, (2012)

[3] K. Fujisawa, T. Endo, Y. Yasui, H. Sato, N. Matsuzawa, S. Matsuoka and H. Waki: Peta-scale general solver for semidefinite programming problems with over two million constraints, The 28th IEEE International Parallel & Distributed Processing Symposium (IPDPS 2014), (2014)