Graph500 Benchmark with Automatic Performance Tuning

[†]Masahiro Nakao, [‡]Koji Ueno, [*]Katsuki Fujisawa
[†]Yuetsu Kodama, [†]Mitsuhisa Sato
[†]RIKEN R-CCS, [‡]Fixstars Corporation, [*]Kyushu University

Background

Large-scale graph processing and Graph500

- There is a growing demand for converting large-scale data in the real world into a graph structure and processing it at high speed
- The number of vertices in such a graph can exceed 1 trillion, and the number of edges can be several tens of times the number of vertices

- There is a performance ranking Graph500 list (https://graph500.org) that measures large-scale graph processing performance
- Graph500 benchmark measures the performance of BFS on graphs with scale-free property (some vertices are connected to many other vertices, and many other vertices are connected to only a few vertices)

Our past research

- We developed Graph500 benchmark that can process large-scale graphs at high speed on supercomputers [1-2]
- As a result of using the maximum 158,976 nodes of the supercomputer Fugaku, we achieved the 1st place of Graph500 list (eight consecutive terms from 2020 to 2023)

Results of this poster

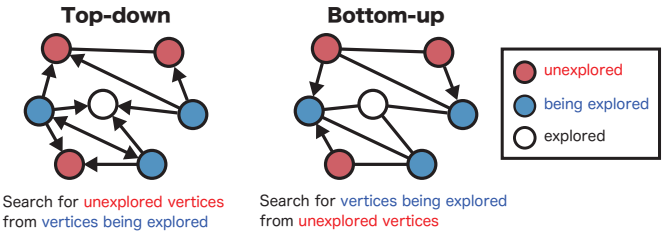
- Development of automatic performance tuning function for parameters of Graph500 benchmark
- Achieved 27% performance improvement as a result of using Fugaku

[1] Koji Ueno et al. "Efficient Breadth-First Search on Massively Parallel and Distributed Memory Machines", Data Science and Engineering, Springer, March 2017, Volume 2, Issue 1, pp 22-35, 2017
[2] Masahiro Nakao et al. "Performance of the Supercomputer Fugaku for Breadth-First Search in Graph500 Benchmark". ISC High Performance, pp. 372-390, June 2021

Graph500 Benchmark

Hybrid-BFS[3]

- Use Top-down in the beginning and end, and Bottom-up in the middle

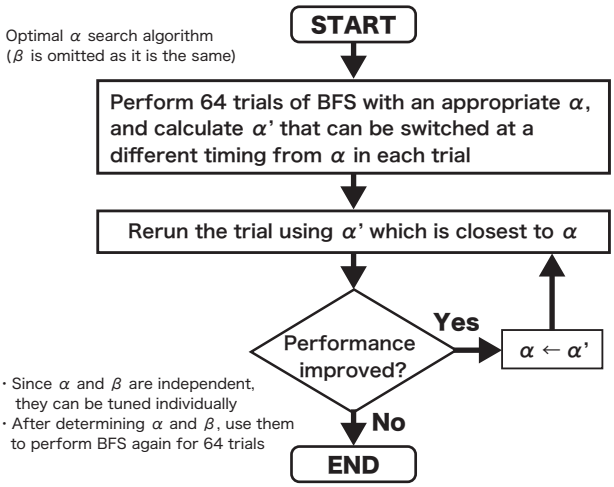


- In the middle stage, the number of vertices being explored increases explosively, so there are fewer attempts to determine vertices (arrow in the diagram above) in Bottom-up than in Top-down
- Timing of switching between Top-down and Bottom-up is as follows:
  - **Top-down → Bottom-up : #edges of the vertex being explored > #edges in graph / α**
  - **Bottom-up → Top-down : #vertices being explored < #vertices in graph / 32β**

[3] Scott Beamer et al. "Direction-optimizing breadth-first search", Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, Salt Lake City, UT, USA, pp. 1-10, 2012

Automatic Top-down and Bottom-up switching

- The α and β are parameters that can be set arbitrarily by the user
- We develop a function to automatically determine α and β

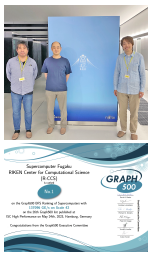


- Since α and β are independent, they can be tuned individually
- After determining α and β, use them to perform BFS again for 64 trials

Evaluation on Fugaku

Results

#nodes	without function	with function	Rate
1	1,903 MTEPS	2,161 MTEPS	14%
12	23.9 GTEPS	28.5 GTEPS	19%
768	1,752 GTEPS	2,110 GTEPS	20%
3,072	5,974 GTEPS	6,851 GTEPS	15%
6,144	8,780 GTEPS	11,211 GTEPS	28%
12,288	13,951 GTEPS	17,221 GTEPS	23%
152,064	113,146 GTEPS	143,487 GTEPS	27%



- Performance is significantly improved regardless of #nodes
- We achieved the 1st place in the world on Graph500 lists in 2023
- Wuhan Supercomputer ranked 2nd with 115,357 GTEPS

Automatic tuning function with 152,064 nodes