Simplifying Parallel Graph Processing:
Survey of Existing Platforms

Sam Pollard (spollard@cs.uoregon.edu), University of Oregon

November 23, 2016

This is a survey of existing graph analytics frameworks.

1 Machine Specifications

Below are the preliminary results by running two benchmarks on the research computer Arya.

CPU Model	AMD Phenom(tm) II X4 965 Processor
CPU Sockets	1
CPU Cores	4
CPU Clock	3400MHz
RAM Size	7.79619Gib
RAM Freq	1600MHz
GPU Model	GF106 [GeForce GTS 450]

Table 1: Machine specifications.

Transport
Network Topology
Local Scheduling
Runtime Feedback
Approach
Algorithmic Considerations

Table 2: Middleware specifications

Performance in millions of traversed edges per second (MTEPS)

	PowerGraph	OpenG
BFS	87.4	341
SSSP	1.09	3.08

Table 3: Performance Results

2 Graph Processing Taxonomy

This is in the spirit of [1]. Here, "|" means "or" and "+" means "and." FOSS means Free and Open Source Software. The quotes around "yes" for HPC mean that the product claims to be amenable to high performance computing. Whether these actually achieve their goal is one of the purposes of this project.

[1]The current version is a closed-source product by Turi though PowerGraph v2.2 is on Github.

Name	Type	HPC	Parallelism	Target	FOSS	Source	Notes
PowerGraph	Framework	"yes"	both	CPU	yes	[2]	[1]
GraphBIG	Benchmark	"yes"	shared	CPU|GPU	yes	[3]	

Table 4: Tools used for graph processing

3 Conclusion

We have presented an updated survey of parallel graph processing frameworks supplementary to [1]. From this, we have selected a representative subset of frameworks on which performance is analyzed and have stored these results in a database. To facilitate parallel graph processing, hardware information and performance results are automatically populated (as were all the tables in this paper). These performance results are then used to provide simple recommendations of the optimally-performing framework given a particular algorithm and problem size.

References

[1] Doekemeijer, N., and Varbanescu, A. L. A survey of parallel graph processing frameworks. Tech. rep., Delft University of Technology, 2014.

[2] Gonzalez, J. E., Low, Y., Gu, H., Bickson, D., and Guestrin, C. Powergraph: Distributed graph-parallel computation on natural graphs. In *Presented as part of the 10th USENIX Symposium on Operating Systems Design and Implementation (OSDI 12)* (Hollywood, CA, 2012), USENIX, pp. 17–30.

[3] Nai, L., Xia, Y., Tanase, I. G., Kim, H., and Lin, C.-Y. GraphBIG: Understanding graph computing in the context of industrial solutions. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (New York, NY, USA, 2015), SC '15, ACM, pp. 69:1–69:12.