A Study of Graph Analytics for Massive Datasets on Distributed Multi-GPUs

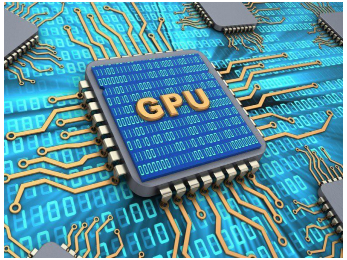
Vishwesh Jatala[1,2], Roshan Dathathri[1,3], Gurbinder Gill[1,3], Loc Hoang[1,3],
V. Krishna Nandivada[4,5], and Keshav Pingali[1,3]

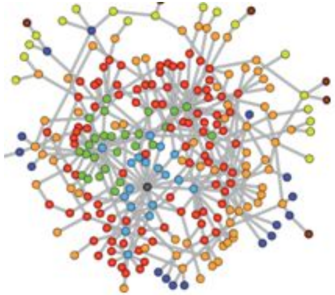
[1]The University of Texas at Austin, USA
[2]vishwesh.jatala@austin.utexas.edu
[3]{loc,roshan,gill,pingali}@cs.utexas.edu

[4]Indian Institute of Technology Madras, India
[5]nvk@iitm.ac.in

IEEE International Parallel and Distributed Processing Symposium (IPDPS) 2020

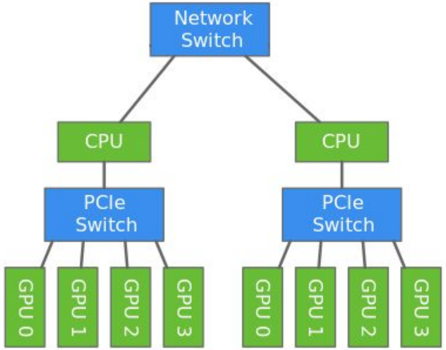
Motivation





Graph Analytics

- Data Growth!
  - Clueweb is ~ 1 TB
    42.5 B edges
- Limited GPU Memory
  - NVIDIA P100 has
    16 GB memory



Distributed Multi-GPUs

Images Source: Internet

Study of Graph Analytics on Distributed GPUs

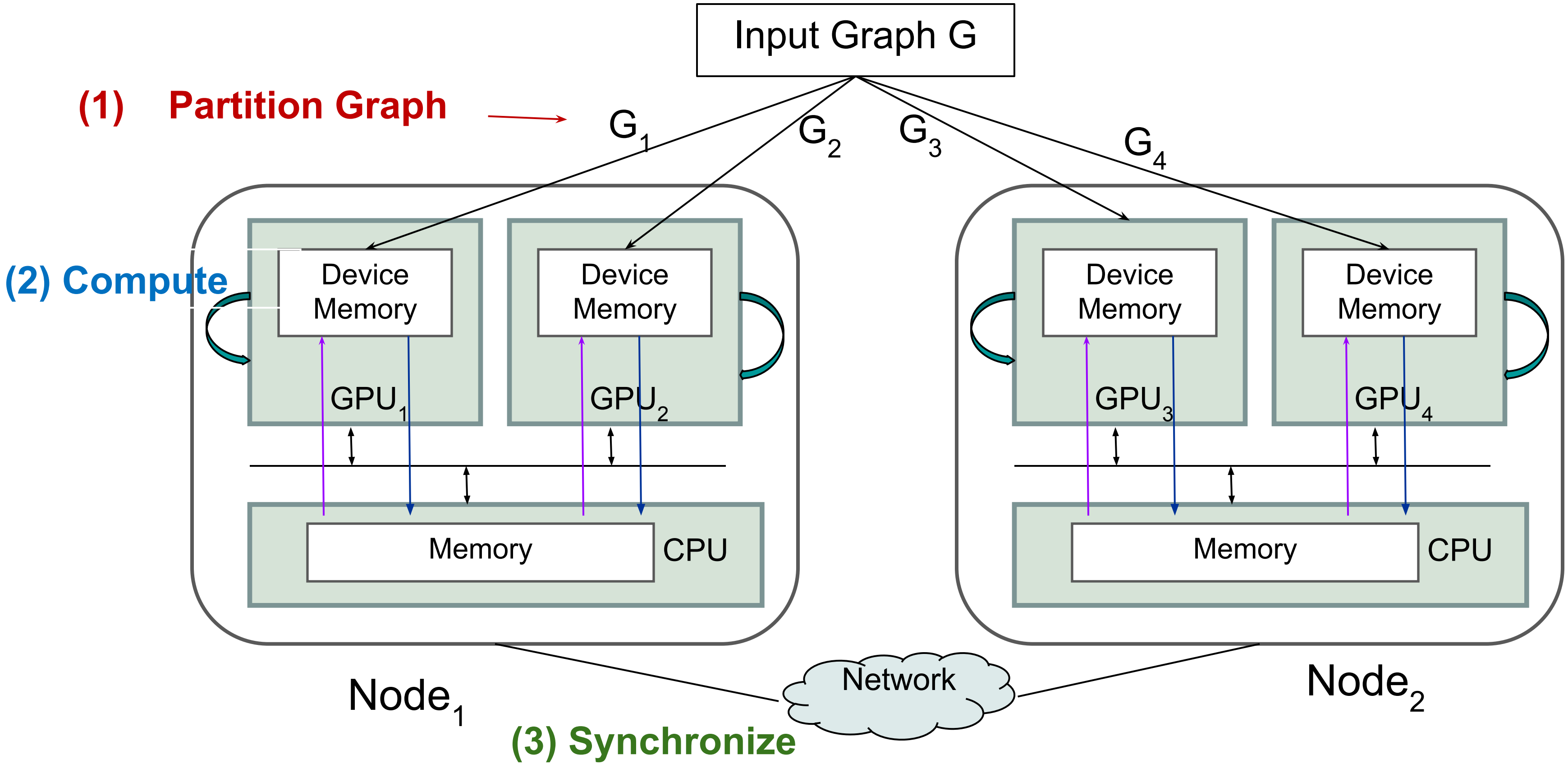
Limitations of Prior Studies

- Customized for few applications
  Scalable BFS [Pan et al. IPDPS'18]
- Focused only for CPUs
  Partitioning study [Gill et al. PVLDB'18]
- Restricted for single GPUs
  Graph survey [Shi et al.Comput.'18]
- Not exhaustive
  [Gluon PLDI'18, Lux PVLDB'17]

Contributions of Our Study

- Shows impact of partitioning, computation, and communication
- Analyzes massive graphs using state-of-the-art D-IrGL
- Provides key suggestions for designers
- Identifies scope for improvements

Distributed Graph Analytics



Evaluated Techniques in The Study

Partitioning

CuSP [Hoang et al. IPDPS'19]

- Incoming Edge Cut[1,2] (IEC)
- Outgoing Edge Cut[2,3] (OEC)
- Cartesian Vertex Cut[2,4] (CVC)
- Hybrid Vertex Cut[5] (HVC)

Computation

- Thread/Warp/CTA Distribution[3,6,7] (TWC)
- Adaptive Load Balancer (ALB)[8]

Synchronization

- Execution Model
  - Bulk-Synchronous Parallel[1,2,9] (BSP)
  - Bulk-Asynchronous Parallel[10] (BASP)
- Communication
  - Update-Only[2]
  - All-Shared[1]

[1]Lux PVLDB'17, [2]Gluon PLDI'18, [3]Gunrock IPDPS'17, [4]Boman et al. SC'13, [5]PowerLyra EuroSys' 15, [6]IrGL OOPSLA'16, [7]Merill et al. PPOPP'12, [8]Jatala et al. Arxiv'19, [9]Valiant CACM'90, [10]Gluon-Async PACT'19,

Experimental Setup

Hardware

Bridges Supercomputer

32 (machines) * 2 (NVIDIA P100 GPUs)

Benchmarks

bfs, sssp, **cc**, **pagerank**, and kcore

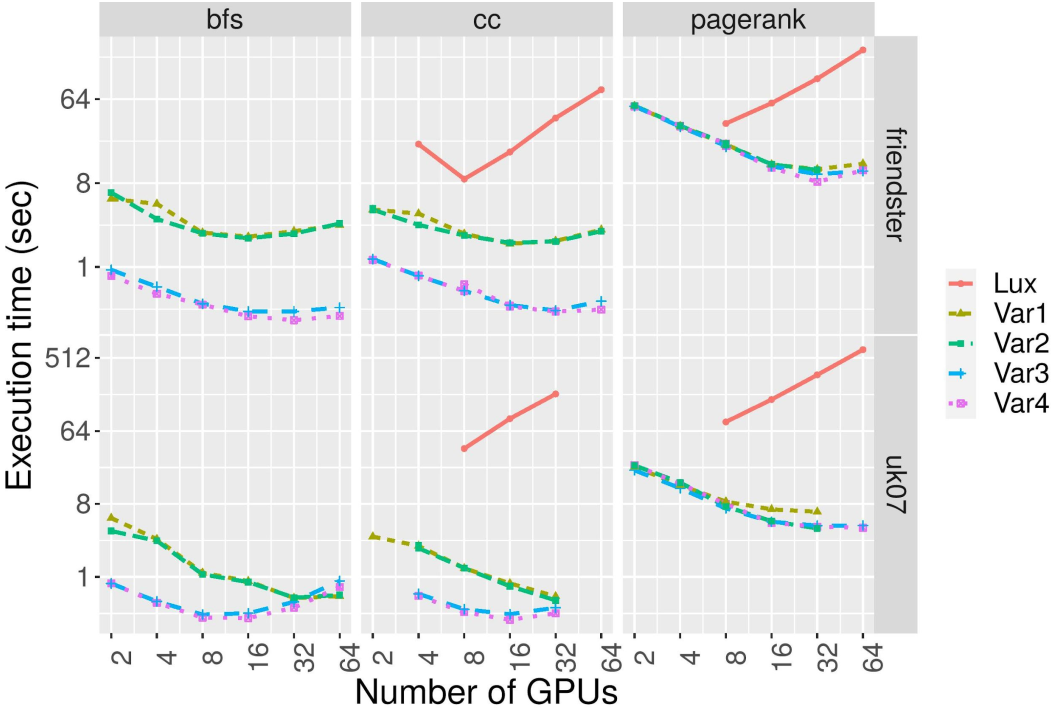
Frameworks

D-IrGL and Lux (Distributed Multi-GPU Frameworks)

Inputs

Inputs (Medium)	|V|	|E|	Input (Large)	|V|	|E|
twitter50	51 M	1,963 M	**clueweb12**	978 M	42.5 B
friendster	66 M	1,806 M	**uk14**	788 M	47.6 B
uk07	106 M	3,739 M	wdc14	1725 M	64.4 B

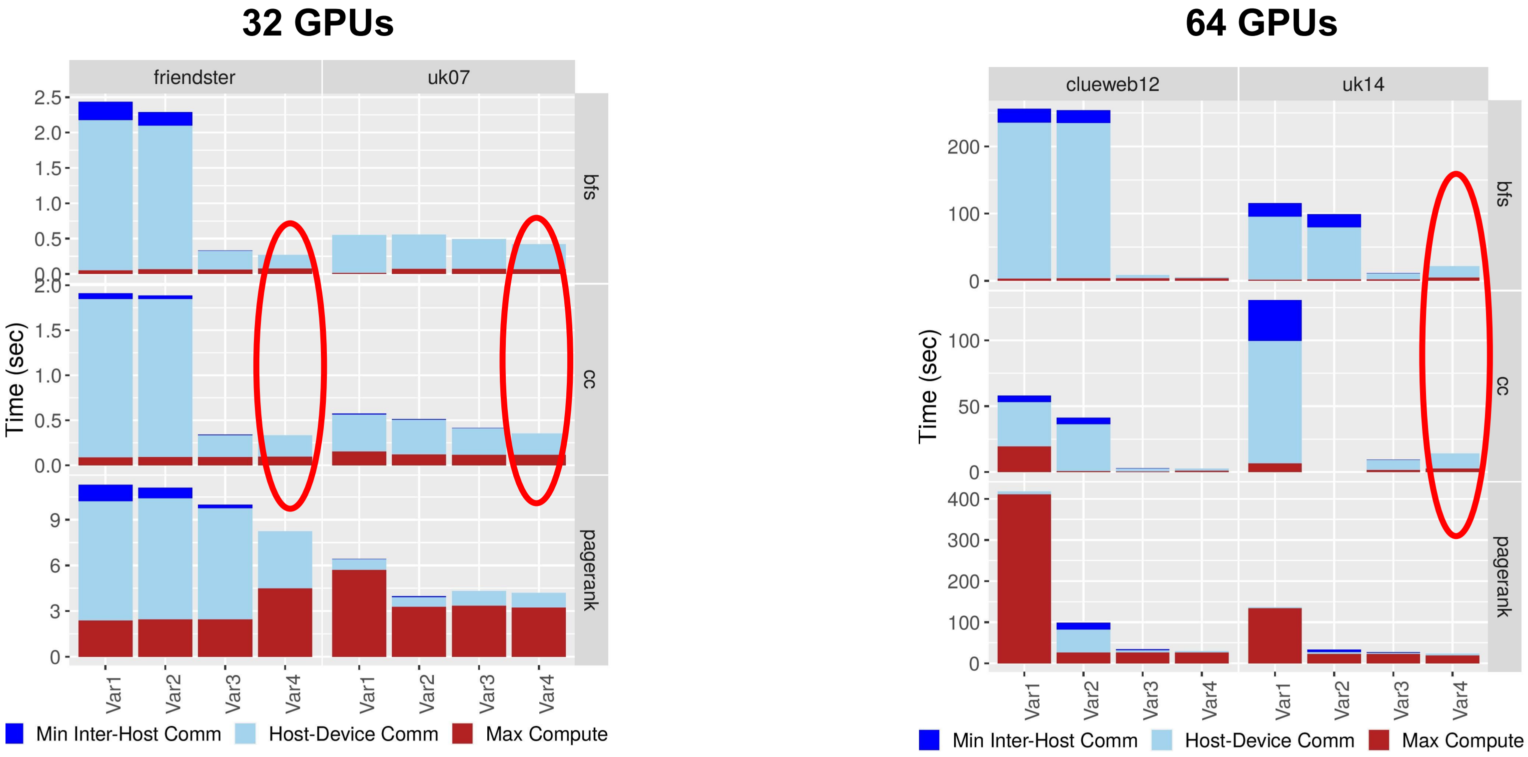
Results: Computation and Synchronization



Variant	Optimizations
Var1	TWC + All Shared + Sync
Var2	ALB + All Shared + Sync
Var3	ALB + Update Only + Sync
Var4 (default)	ALB + Update Only + Async

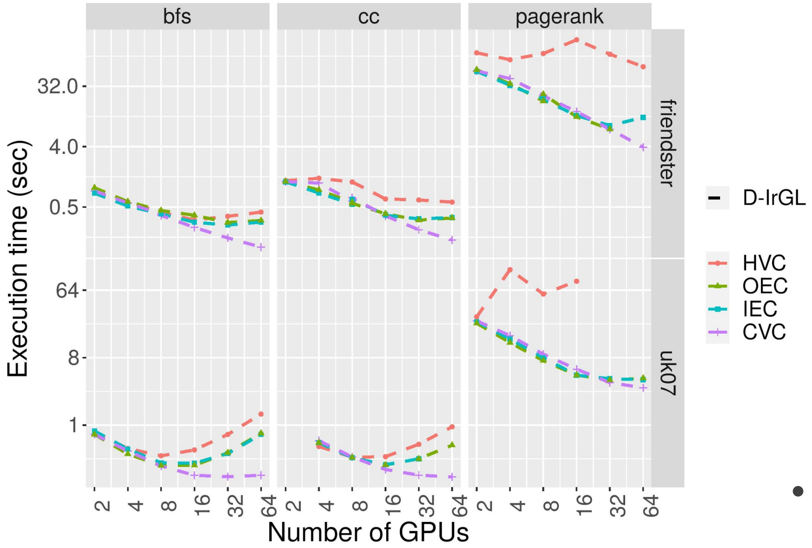
All variants (even Lux) use same partitioning policy (IEC).

Analyzing Computation and Synchronization



Host-device time is significant **Asynchronous behavior should be throttled**

Analyzing Partitioning Schemes



Benchmark	Partition	uk07 on 32 GPUs			uk14 on 64 GPUs		
		Static	Dynamic	Memory	Static	Dynamic	Memory
bfs	CVC	1.15	1.17	1.15	1.15	1.11	1.14
	HVC	1.10	1.20	1.08	1.40	1.38	1.38
	IEC	1.00	1.14	1.04	1.00	1.31	1.08
	OEC	1.00	1.20	1.02	1.00	1.24	1.03
cc	CVC	1.03	1.18	1.05	1.12	1.10	1.13
	HVC	1.09	1.30	1.08	1.11	1.34	1.11
	IEC	1.00	1.27	1.02	1.00	1.24	1.04
	OEC	1.00	1.29	1.02	1.00	1.22	1.04
pagerank	CVC	1.16	1.04	1.15	1.15	1.02	1.14
	IEC	1.00	1.09	1.04	1.00	1.09	1.08
	OEC	1.00	1.10	1.03	1.00	1.08	1.04

- CVC outperforms other schemes on 16 or more GPUs
  - fewer communication partners

- Static load imbalance not correlated to dynamic load imbalance
- Static load imbalance is correlated to memory usage
  - Critical for GPUs due to limited memory

Conclusion

- Detailed analysis of distributed multi-GPU graph analytics
- Lessons:
  - CVC is crucial for scaling
  - Static load balance is important for GPUs
- Scope for Improvements:
  - Reduce host-device communication time through GPUDirect
  - Control asynchrony in Bulk-Asynchronous execution

Please contact authors for any questions. Thank you!

http://iss.oden.utexas.edu/?p=projects/galois Galois