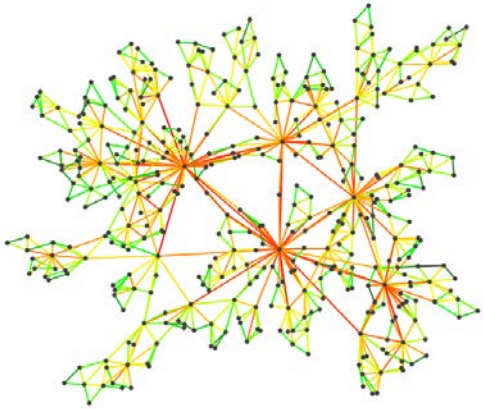
清華大學
Tsinghua University

FPGP: Graph Processing Framework on FPGA

Guohao DAI, Yuze CHI, Yu WANG, Huazhong YANG
E.E. Dept., TNLIST, Tsinghua University
dgh14@mails.tsinghua.edu.cn

Big graph is widely used

- Big graph is widely used in many domains
- Involved with **billions** of edges and **Gbytes ~ Tbytes** storage (On-chip memory/DRAM not applicable, needs disk to store!)
  - WeChat: 0.65 billions active users (2015)
  - Facebook: 1.55 billions active users (2015Q3)
  - Twitter2010: 1.5 billions edges, 13GB
  - Yahoo-web: 6.6 billions edges, 51GB
  - Page: 129 billions edges, 1.1TB
- Different graph algorithms
  - Generality requirement



Social network analysis

facebook

amazon.com

User behavior analysis

Bio-sequence analysis

ENCODE

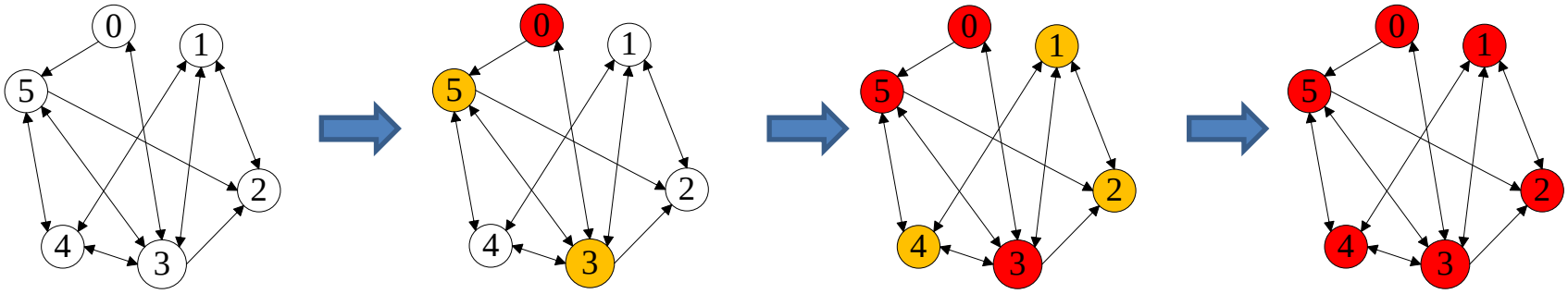
NETFLIX

User preference recommendation

G. Dror, N. Koenigstein, Y. Koren, and M. Weimer. The yahoo! music dataset and kdd-cup'11
H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media?

Generality requirement

- High-level abstraction model
  - Read-based/Queue-based Model for BFS/APSP **[PACT10]** ×
  - Vertex-Centric Model **[SIGMOD10]** √
    - A vertex updated → Neighbor vertices to be updated
    - Different graph algorithms → Different updating functions



- Vertex-Centric Model is memory-bounded
  - Random memory access pattern
  - Poor locality

Low memory access bandwidth

[PACT10] Hong S, Oguntebi T, Olukotun K. Efficient parallel graph exploration on multi-core CPU and GPU
[SIGMOD10] Malewicz G, Austern M H, Bik A J C, et al. Pregel: a system for large-scale graph processing

Graph partition

- Graph partition to solve the memory-bounded problem
  - Locality
  - Sequential memory access

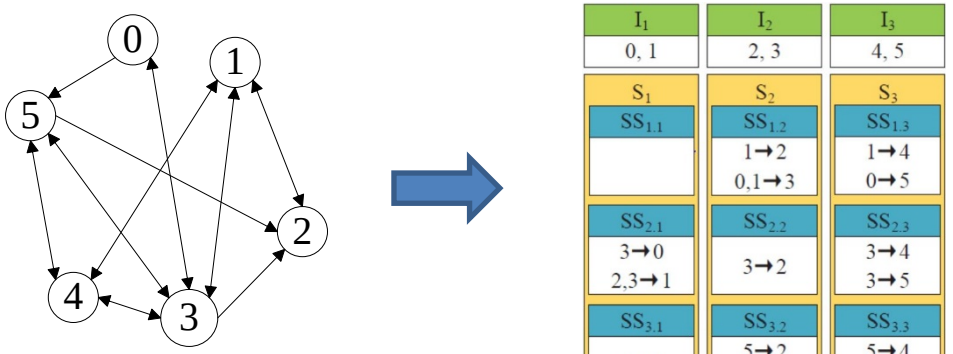
  Higher bandwidth, friendly to disks & SSDs
  - Less data transfer
  - Higher degree of parallelism

  Larger graph size

System	VENUS[ICDE15]	GridGraph[ATC15]	X-stream[SOSP13]	Our method [ICDE16]*
Execute time(s)	95.48	24.11	81.70	**12.55**

1 iteration of PageRank on Twitter2010 graph, HDD

- Partition method
  - Vertices: **Intervals**, Edges: **Sub-Shards**



*[ICDE16] Y. Chi, G. Dai, Y. Wang, G. Sun, G. Li, and H. Yang. Nxgraph: An efficient graph processing system on a single machine.

Related work

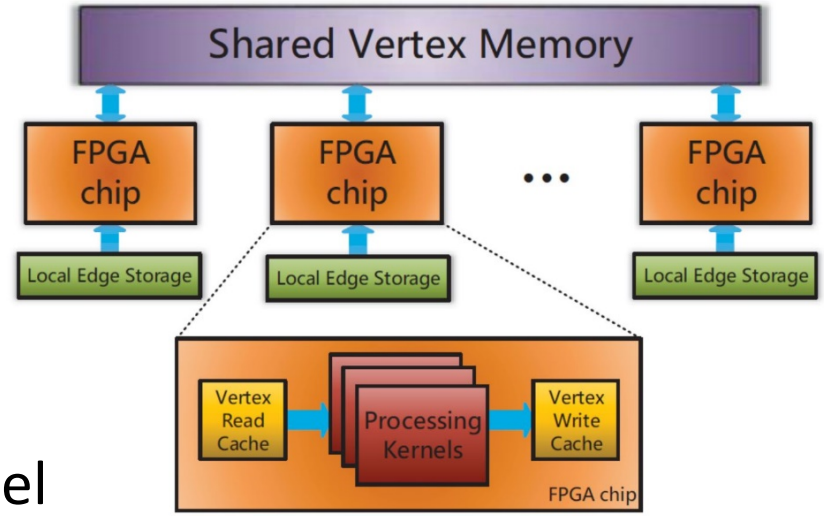
Work	Graph size	Platform	Generality	Limitation
Brahim et al. [FPT11, FPL12]	Millions of edges	Convey, Virtex-5 LX330 FPGA	APSP, Graphlet counting	Dedicated algorithms
Brahim et al. [ASAP12]	1 billion edges	Convey, Virtex-5 LX330 FPGA	BFS	Dedicated algorithms
Eriko et al. [FCCM14] GraphGen	Millions of edges	ML 605 / DE4	Several graph algorithms	The size of CoRAM
Kyrola et al. [OSDI12] GraphChi	Billions of edges	AMD Opteron CPU	Several graph algorithms	Power efficiency Partition method
Our work [ICDE16] Nxgraph	Billions of edges	Intel i7 CPU	Several graph algorithms	Power efficiency

- We want to propose a solution that can handle graphs with **billions of edges** on FPGAs and **apply to several graph algorithms**

[FPT11] Betkaoui B, Thomas D B, et al. A framework for FPGA acceleration of large graph problems: graphlet counting case study
[ASAP12] Betkaoui B, Wang Y, et al. A reconfigurable computing approach for efficient and scalable parallel graph exploration
[FPL12] Betkaoui B, Wang Y, et al. Parallel FPGA-based all pairs shortest paths for sparse networks: A human brain connectome case study
[FCCM14] Nurvitadhi E, Weisz G, Wang Y, et al. Graphgen: An fpga framework for vertex-centric graph computation
[ICDE16] Chi Y, Dai G, Wang Y, et al. NXgraph: An Efficient Graph Processing System on a Single Machine
[OSDI12] Kyrola A, Blelloch G E, Guestrin C. GraphChi: Large-Scale Graph Computation on Just a PC

FPGP Framework

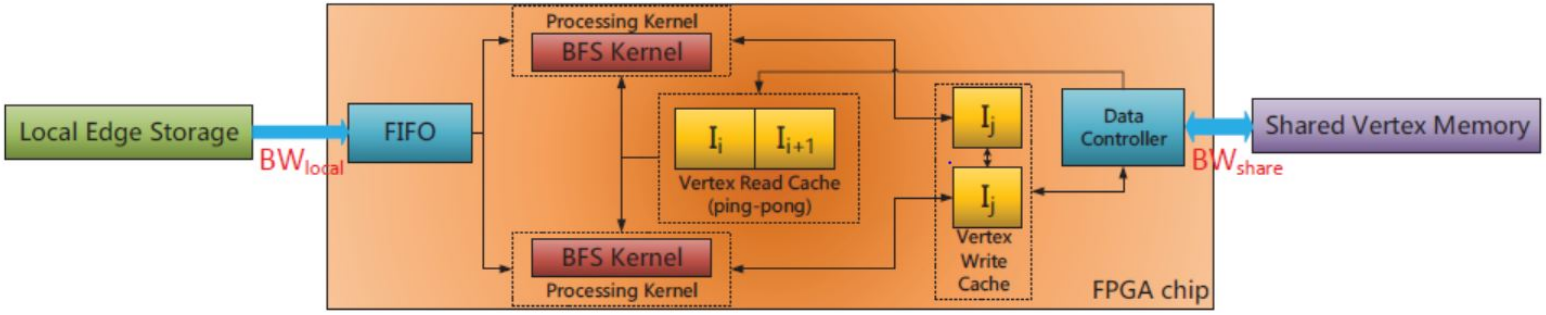
- Map the interval-shard based graph structure to FPGA
  - Improve the memory access efficiency



- Processing Kernel
  - Configured with different updating functions (Generality)
  - Update destination interval using source interval
- Storage can be extended to ~Gbytes (Graph size)
  - Multiple FPGA attached with Local Edge Storage (potentially bandwidth improvement)

Our FPGA implementation

- On-chip logic: Xilinx Virtex-7 FPGA VC707, one board
- Simulate the bandwidth



- Performance (BFS)

Graph	GraphChi[OSDI12]	TurboGraph[SIGKDD13]	FPGP
Twitter2010	148.6	76.1	**121.9**
Yahoo-web	2451.6	-	**635.4**

- Graph size
  - Sequential edge access pattern (Local Edge Storage can be SSD!)

System	GraphGen[FCCM14]	Brahim's work[ASAP12]	FPGP
Maximum graph size*	Millions of edges	1 billion edges	**~100 billions edges**

* Inferred from paper

Limitation

- Graph problems are memory-bounded
  - Resources utilization unbalanced
    - The size of BRAM becomes the bottleneck

Resource	Utilization	Available	Utilization
FF	610	607200	0.1%
LUT	4399	303600	1.5%
BRAM	928	1030	90%
BUFG	1	32	3%

  - Limited on-chip memory leads to frequent interval replacement (Swapping on-chip intervals with in-memory intervals)
    - May cause scalability problems (graphs with **billions of vertices**)
    - BRAM: ~Mbytes, so graphs with **billions of vertices** have heavy replacement overhead

Conclusion & Future work

- We proposed an FPGA graph processing framework, FPGP
  - Handle graphs with billions of edges
  - Apply to several graph algorithms
  - Sequential edge access pattern, friendly to disks/SSDs
  - Power efficiency
- Future work
  - Multi-FPGA platform demo
  - Larger on-chip memory technique
    - 3D stacked memory
    - In memory computing

Reference

1. Hong S, Oguntebi T, Olukotun K. Efficient parallel graph exploration on multi-core CPU and GPU[C]//Parallel Architectures and Compilation Techniques (PACT), 2011 International Conference on. IEEE, 2011: 78-88.
2. Boccaletti S, Ivanchenko M, Latora V, et al. Detecting complex network modularity by dynamical clustering[J]. Physical Review E, 2007, 75(4): 045102.
3. Chi Y, Dai G, Wang Y, et al. NXgraph: An Efficient Graph Processing System on a Single Machine[J]. arXiv preprint arXiv:1510.06916, 2015.
4. Low Y, Bickson D, Gonzalez J, et al. Distributed GraphLab: a framework for machine learning and data mining in the cloud[J]. Proceedings of the VLDB Endowment, 2012, 5(8): 716-727.
5. Kyrola A, Blelloch G E, Guestrin C. GraphChi: Large-Scale Graph Computation on Just a PC[C]//OSDI. 2012, 12: 31-46.
6. Nurvitadhi E, Weisz G, Wang Y, et al. GraphGen: An FPGA Framework for Vertex-Centric Graph Computation[C]//Field-Programmable Custom Computing Machines (FCCM), 2014 IEEE 22nd Annual International Symposium on. IEEE, 2014: 25-28.
7. Betkaoui B, Thomas D B, Luk W, et al. A framework for FPGA acceleration of large graph problems: graphlet counting case study[C]//Field-Programmable Technology (FPT), 2011 International Conference on. IEEE, 2011: 1-8.
8. Betkaoui B, Wang Y, Thomas D B, et al. A reconfigurable computing approach for efficient and scalable parallel graph exploration[C]//Application-Specific Systems, Architectures and Processors (ASAP), 2012 IEEE 23rd International Conference on. IEEE, 2012: 8-15.

Reference

9. Betkaoui B, Wang Y, Thomas D B, et al. Parallel FPGA-based all pairs shortest paths for sparse networks: A human brain connectome case study[C]//Field Programmable Logic and Applications (FPL), 2012 22nd International Conference on. IEEE, 2012: 99-104.
10. Roy A, Mihailovic I, Zwaenepoel W. X-stream: Edge-centric graph processing using streaming partitions[C]//Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles. ACM, 2013: 472-488.
11. Han W S, Lee S, Park K, et al. TurboGraph: a fast parallel graph engine handling billion-scale graphs in a single PC[C]//Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013: 77-85.
12. Cheng J, Liu Q, Li Z, et al. VENUS: Vertex-Centric Streamlined Graph Computation on a Single PC[C]//ICDE. 2015.
13. Zhu X, Han W, Chen W. GridGraph: Large-scale graph processing on a single machine using 2-level hierarchical partitioning[C]//Proceedings of the Usenix Annual Technical Conference. 2015: 375-386.
14. Malewicz G, Austern M H, Bik A J C, et al. Pregel: a system for large-scale graph processing[C]//Proceedings of the 2010 ACM SIGMOD International Conference on Management of data. ACM, 2010: 135-146.
15. Nurvitadhi E, Weisz G, Wang Y, et al. Graphgen: An fpga framework for vertex-centric graph computation[C]//Field-Programmable Custom Computing Machines (FCCM), 2014 IEEE 22nd Annual International Symposium on. IEEE, 2014: 25-28.

Thank you

Q & A