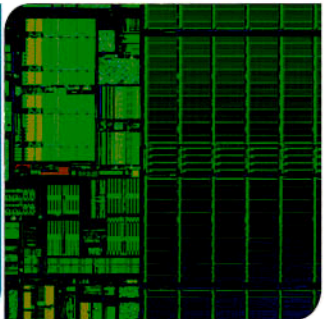
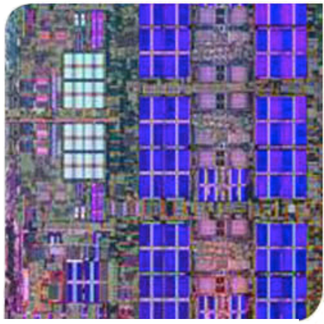
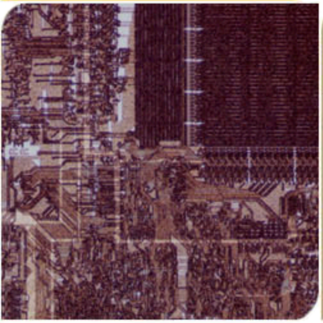




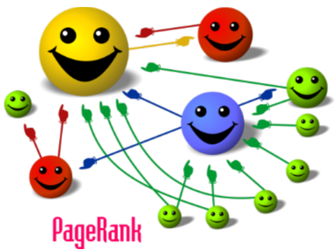
Optimizing Heterogeneous Platforms for Unstructured Parallelism

Pis: Sudhakar Yalamanchili
Hyesoon Kim
Richard Vuduc

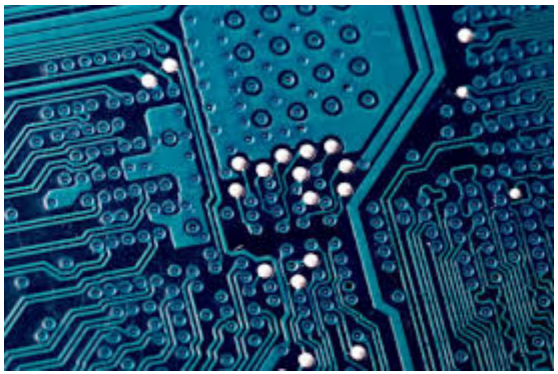


Georgia Tech
comparch

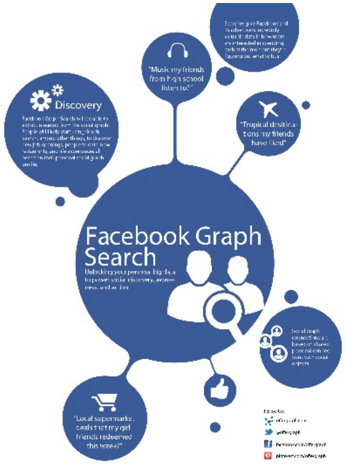
Graph Algorithms



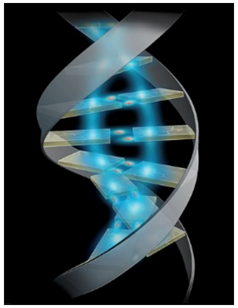
Web Search



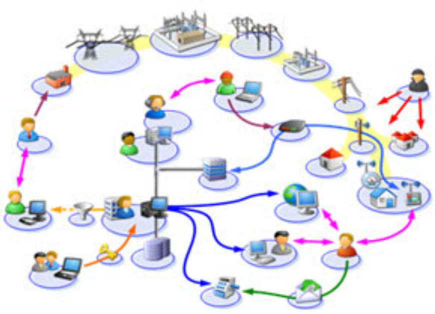
Chip Design



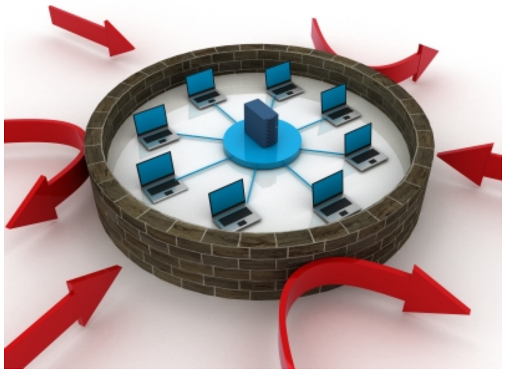
Data Analytics



Bioinformatics



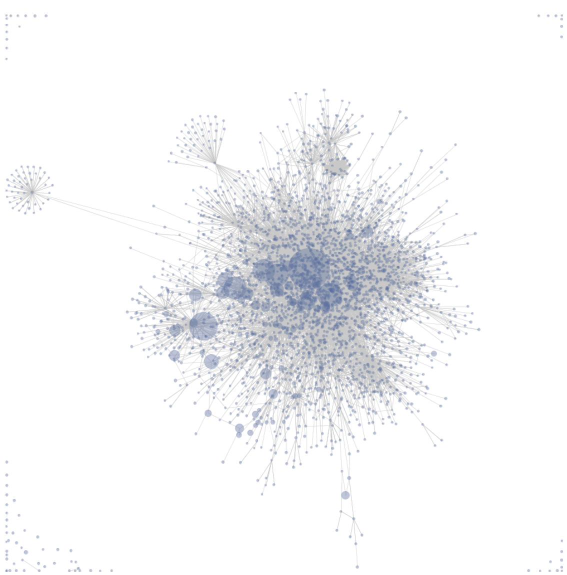
Networks



Network Security

Research Objective

- Coordinated algorithm, compiler and architecture design efforts to support unstructured parallelism applications on Heterogeneous platforms



Algorithms

Compiler

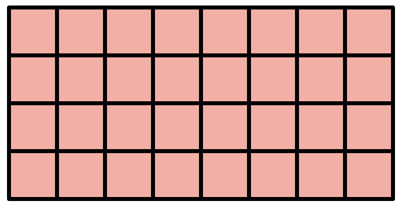
Microarchitecture

CPU | GPU

1) Increase dynamic parallelism
2) Reduce memory divergence
3) Scalable graph algorithms

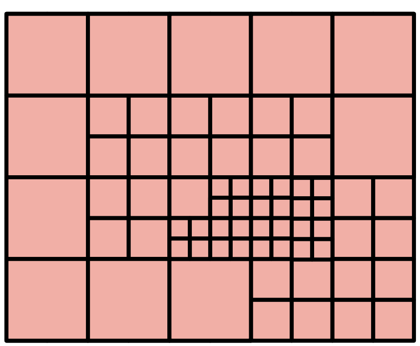
Motivation

- GPUs are effective for structured applications

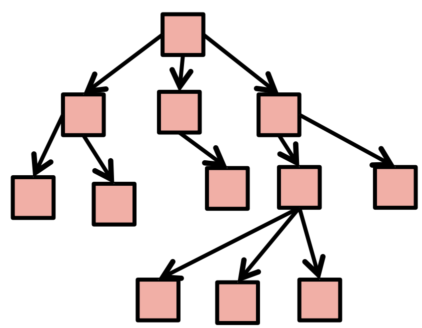


Rigid 1/2/3D data structure

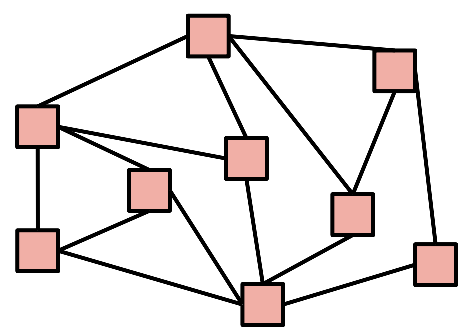
- However, for unstructured applications
  - Poor workload balance -> control flow divergence
  - Un-coalesced memory access -> memory divergence
  - Low Compute Utilization



Adaptive Mesh



Tree

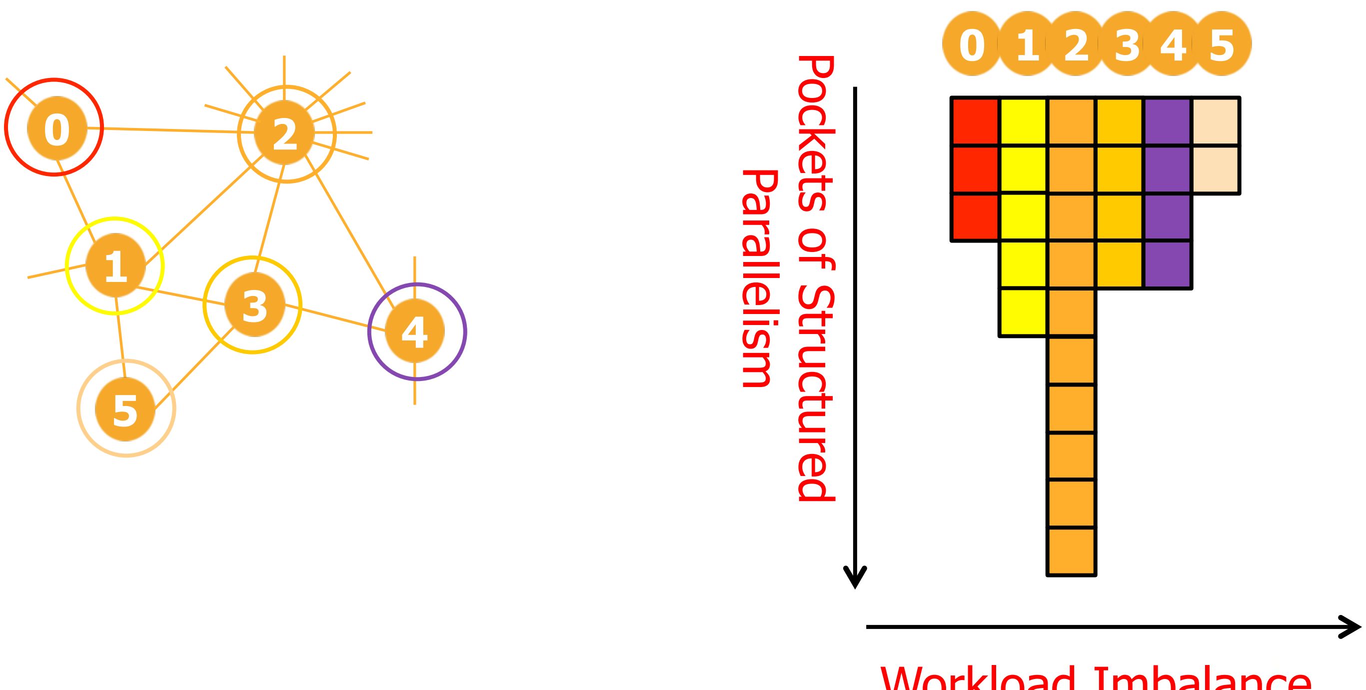


Graph

Dynamically Formed Parallelism (DFP)

- Pockets of Structured Parallelism in irregular applications
  - Locally uniform control flow
  - Locally coalesced memory access

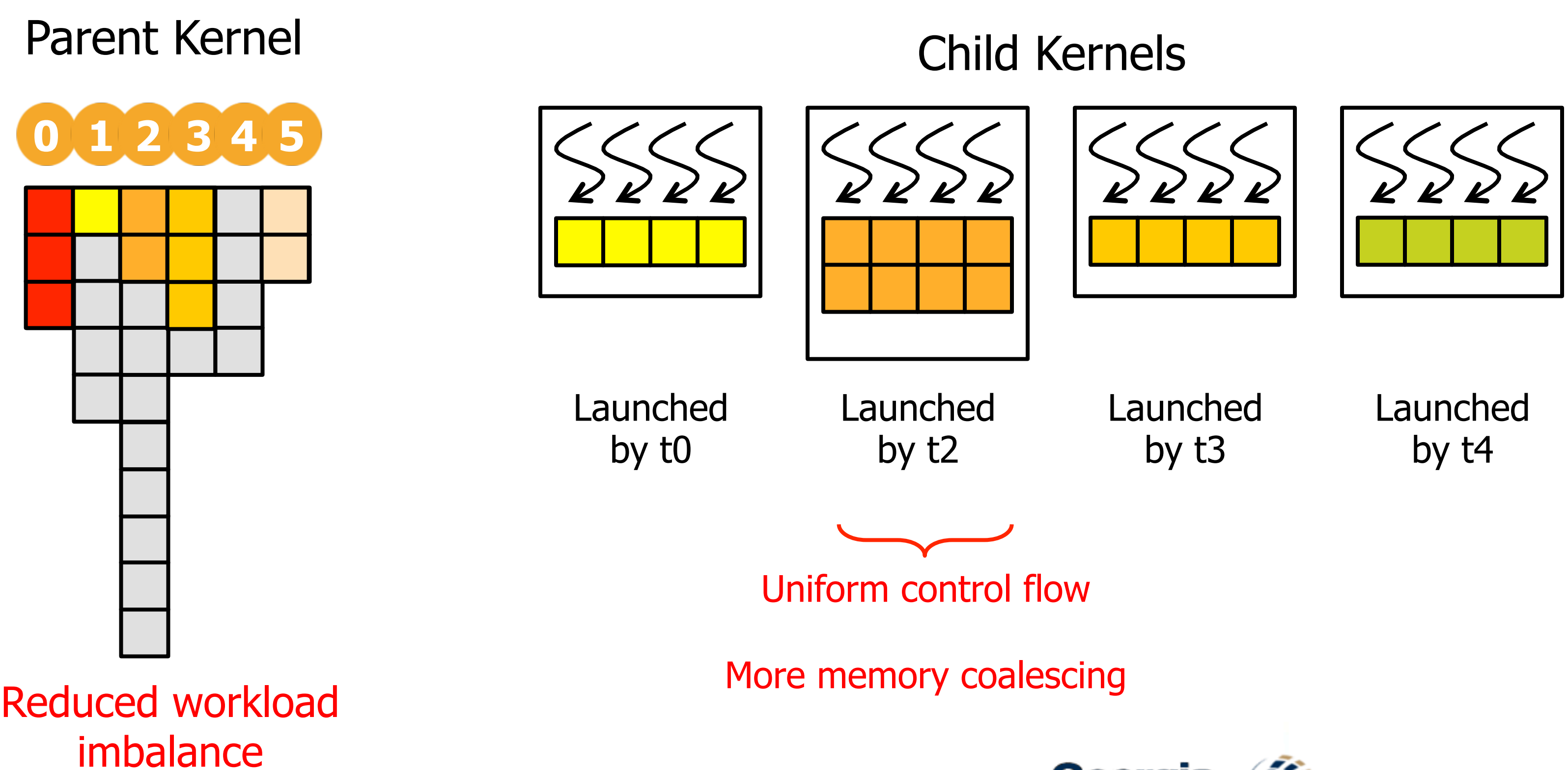
- E.g. Graph Traversal



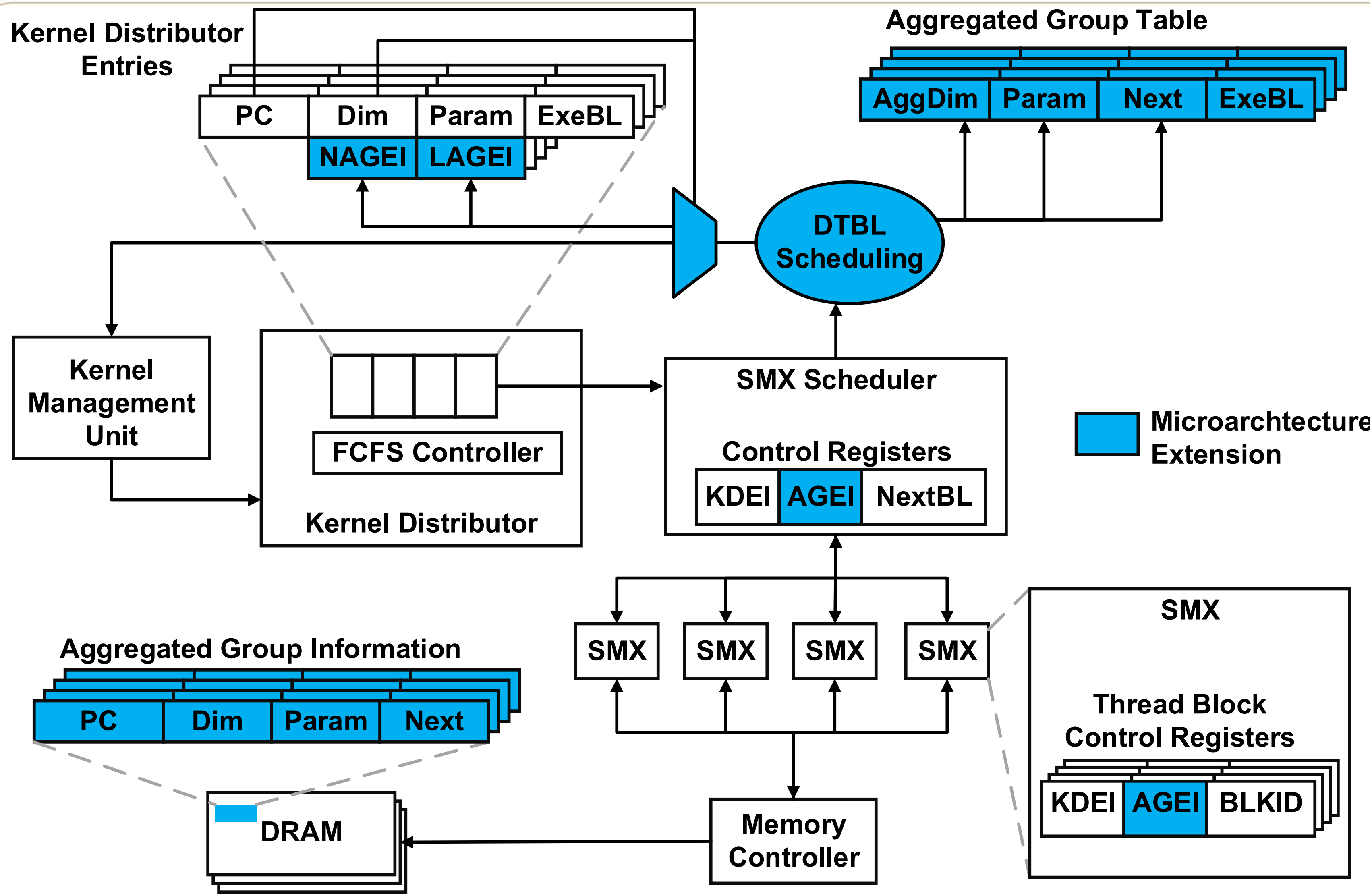
Wang et al. IISWC 2014

Implemented with CUDA Dynamic Parallelism (CDP)

- CDP: launch a kernel from device side, on Kepler GK110 GPU
- Solution for DFP
  - Launch a child kernel for detected DFP
  - Launch only when sufficient parallelism



Dynamic Thread Block Launch



Wang et al. ISCA 2015

Memory Problem

- Memory divergence problem
- Memory latency problem

  - Prefetch a **data-dependent** memory access pattern **found commonly** in graph algorithms

  - Use **spare registers** to make prefetches more **effective**

Lakshminarayana et al. HPCA 2014

SRAP – Inserting Prefetches

- Typical graph traverse patterns in GPUs
- Loop with identified access pattern
  - Two loads into r2 and r3, second load dependent on first load
  - First load has a stride of 4

```
ld r2, [r1]
add r3, r0, r2
ld r4, [r3]
```

Lakshminarayana et al. HPCA 2014

GraphBig Benchmark

Category	Workload	Computation Type	CPU	GPU	Use Case Example
Graph traversal	BFS	CompStruct	√	√	Recommendation for Commerce
	DFS	CompStruct	√		Visualization for Exploration
Graph update	Graph construction (GCons)	CompDyn	√		Graph Analysis for Image Processing
	Graph update (GUp)	CompDyn	√		Fraud Detection for Bank
	Topology morphing (TMorph)	CompDyn	√		Anomaly Detection at Multiple Scales
Graph analytics	Shortest path (SPath)	CompStruct	√	√	Smart Navigation
	K-core decomposition (kCore)	CompStruct	√	√	Large Cloud Monitoring
	Connected component (CComp)	CompStruct	√	√	Social Media Monitoring
	Graph coloring (GColor)	CompStruct		√	Graph matching for genomic medicine
	Triangle count (TC)	CompProp	√	√	Data Curation for Enterprise
	Gibbs inference (GI)	CompProp	√		Detecting Cyber Attacks
Social analysis	Degree centrality (DCentr)	CompStruct	√	√	Social Media Monitoring
	Betweenness centrality (BCentr)	CompStruct	√	√	Social Network Analysis in Enterprise

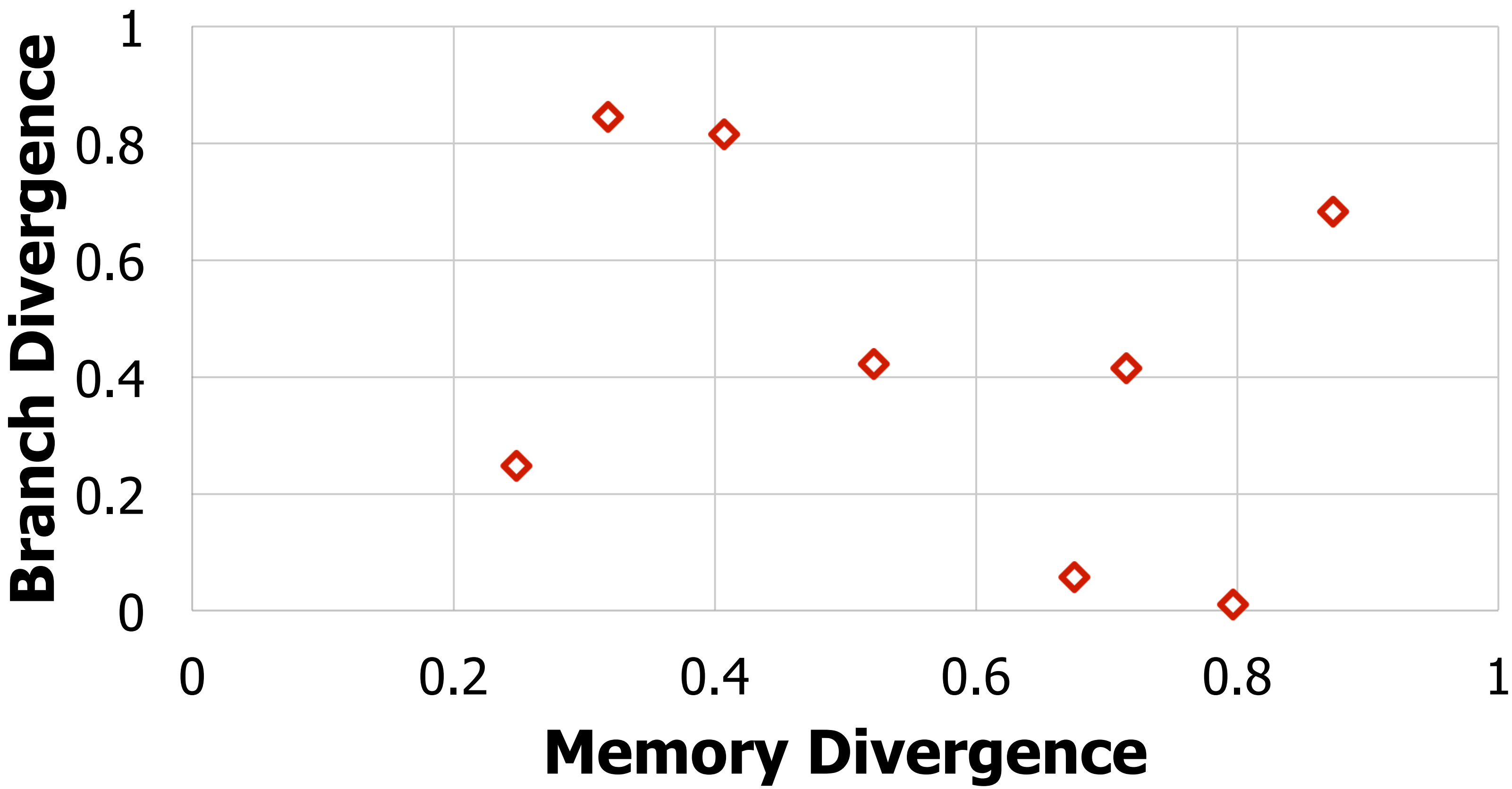
Data Set	Type	Vertex#	Edge#
Twitter Graph	Type 1	120M	1.9B
IBM Knowledge Repo	Type 2	154K	1.72M
IBM Watson Gene Graph	Type 3	2M	12.2M
CA Road Network	Type 4	1.9M	2.8M
LDBC Graph	Synthetic	Any	Any

IBM System-G based benchmarks

https://github.com/graphbig/graphBIG/wiki

Collaboration with IBM

Memory and Control Divergence in GraphBig



Georgia Tech comparch

On-going work

- Algorithm-level work
  - Staleness aware graph computing
    - Reduce synchronization → Increase parallelism
  - BFS 1.5 streaming partitioning algorithms

Project Outcomes

- J. Wang, A. Sidelink, N Rubin, and S. Yalamanchili, "Dynamic Thread Block Launch: A Lightweight Execution Mechanism to Support Irregular Applications on GPUs", IEEE/ACM International Symposium on Computer Architecture (ISCA), June 2015.
- J. Wang and S. Yalamanchili. "Characterization and Analysis of Dynamic Parallelism in Unstructured GPU Applications." IEEE International Symposium on Workload Characterization (IISWC). October 2014.
- Nagesh B Lakshminarayana, Hyesoon Kim, "Spare Register Aware Graph Algorithms on GPUs", HPCA Feb 2014.
- O. Green, M. Dukhan, R. Vuduc. "Branch-avoiding graph algorithms." In Proceedings of the ACM Symposium on Parallel Architectures and Algorithms (SPAA), Portland, OR, USA, June 2015.
- J. Li, C. Battaglino, I. Perros, J. Sun, R. Vuduc. "An input-adaptive and in-place approach to dense tensor-times-matrix multiply." (submitted, April 2015)
- GraphBig Benchmark https://github.com/graphbig/graphBIG/wiki (Submitted, April 2015)

Georgia Tech comparch