UNIVERSITY OF CAMBRIDGE

Efficient Large-Scale Graph Processing on Hybrid CPU and GPU Systems

Abdullah Gharaibeh, Elizeu Santos-Neto, Lauro Costa and Matei Ripeanu

Reviewer: Varun Gandhi (vg292)

Computer Laboratory

CPU-GPU Hybrid Systems

One of the fastest desktop CPU & GPU

+

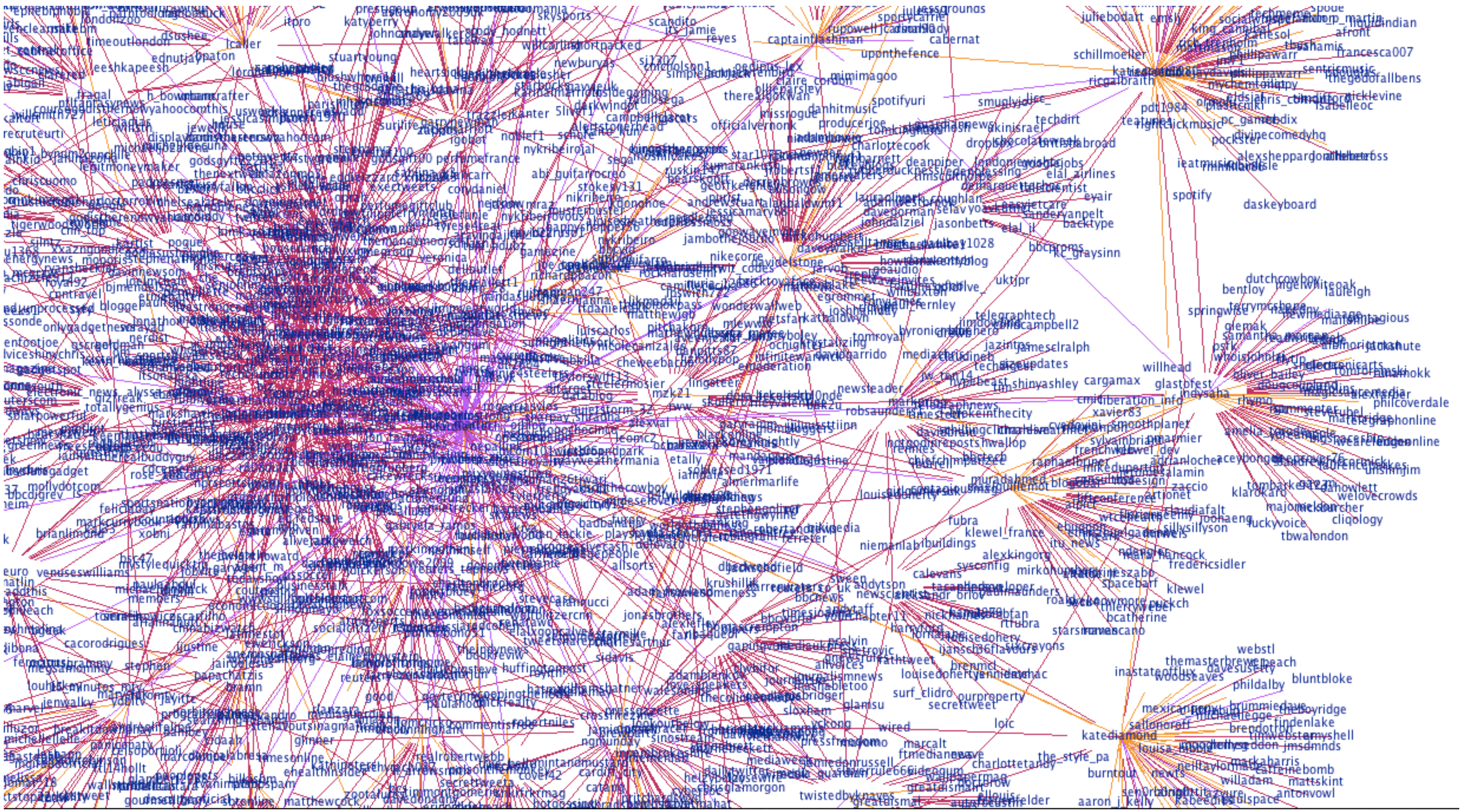
8 cores

2048 CUDA cores

Conventional Applications

New Dimension

Single node graph computation



Real-world graph characteristics

Single node bottlenecks

- High memory foot print
- Heterogenous degree
- Cost of partitioning

Key Idea

- Load balancing across GPU & CPU
- Algorithm agnostic
- Different than GraphCHI[1]

Hybrid Model

- Two processing units
- Communication rate: edges per second
- Majority of edges remain at CPU
- Random partitioning

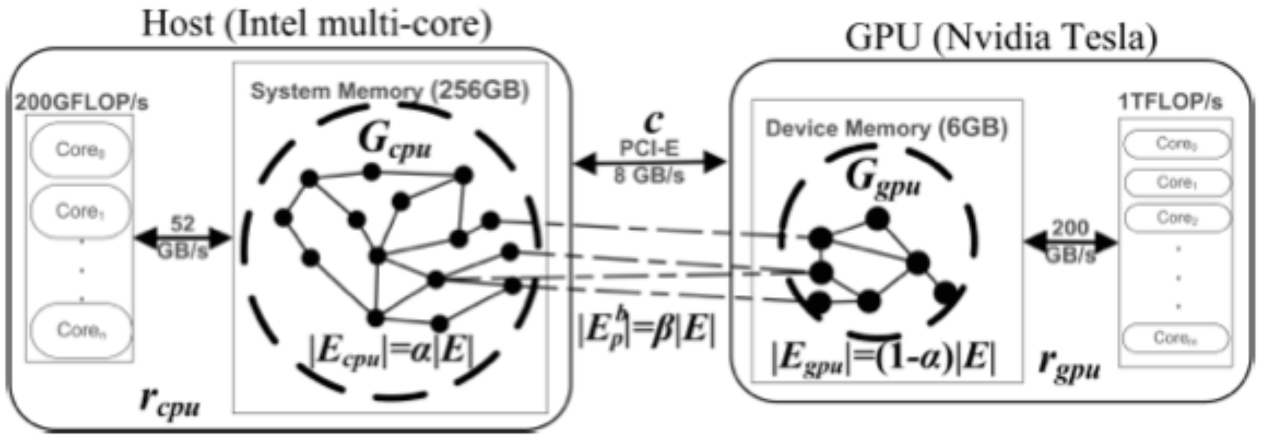
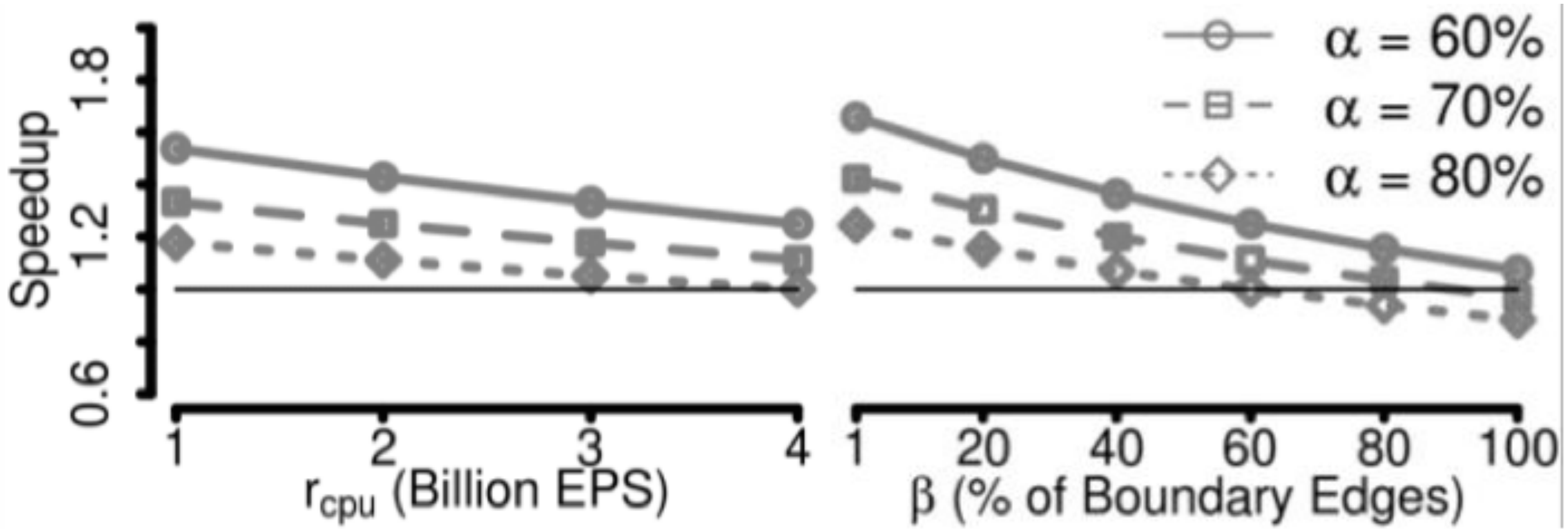


Figure 1: An illustration of the model, its parameters, and their values for today's state-of-the-art commodity components.

r_{cpu} r_{gpu}	*Processing rates on the CPU and GPU*
c	*Communication rate between the host and GPU*
α	*Ratio of the graph edges that remain on the host*
β	*Ratio of edges that cross the partition*

Simulation Results

Predicted gains based on simulated model

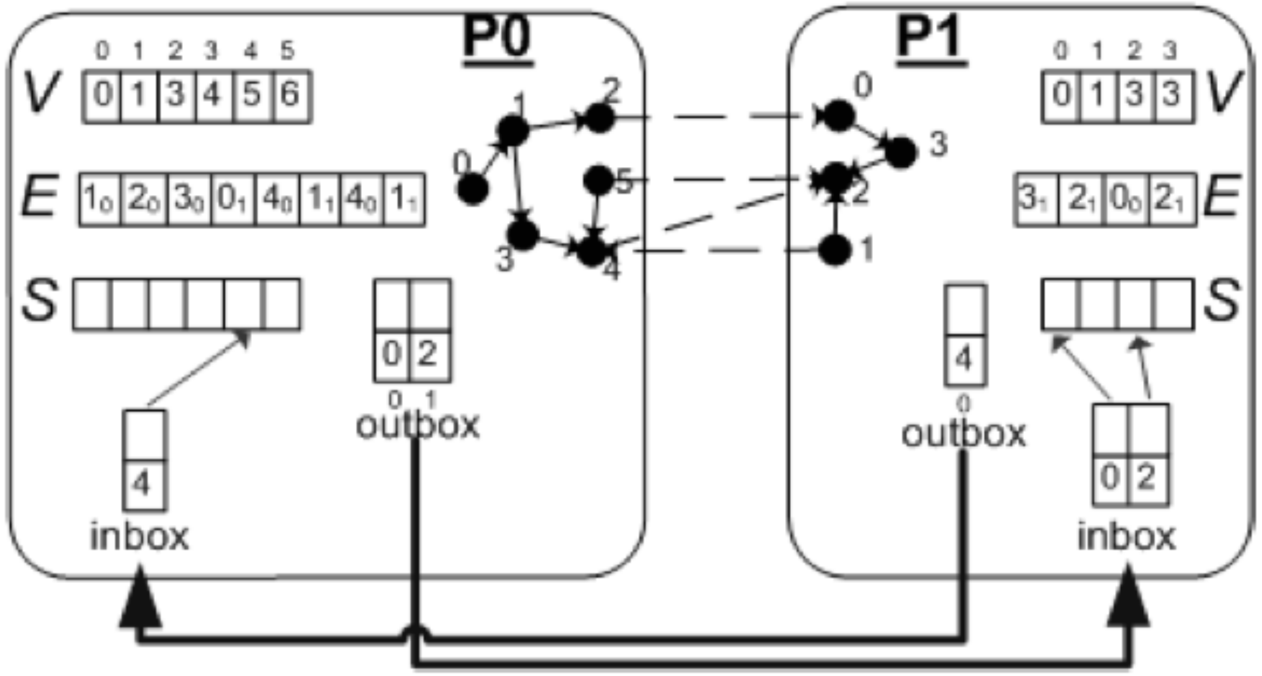


TOTEM

- Implemented in both C & CUDA
- Adopts BSP model
- Computation phase
- Communication phase
- Termination

Trade-off: Graph Representation

- Compressed Sparse rows
- Low memory footprint
- Expensive updates



Trade off: Communication Overhead

- Mutable graph structures expensive
- GPU cannot be leveraged
- Outbox values copied to Inbox
- Aggregate at source
- Transfer based on user-provided callback

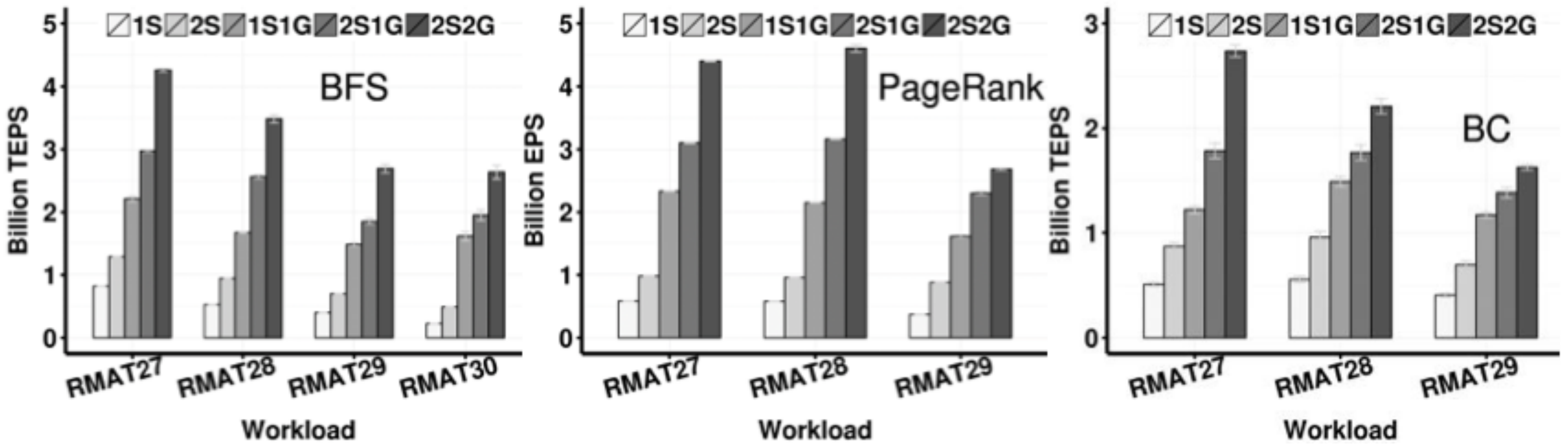
Graph Partitioning

- High degree — GPU
- Low degree — CPU
- Leverages low communication overhead
- Fails to maintain boundary edge threshold

Synthetic Workload

Workload	|V|	|E|
Twitter [Cha et al. 2010]	52M	1.9B
UK-Web [Boldi et al. 2008]	105M	3.7B
RMAT27	128M	2.0B
RMAT28	256M	4.0B
RMAT29	512M	8.0B
RMAT30	1,024M	16.0B

Evaluation


1S 2S 1S1G 2S1G 2S2G
BFS
Billion TEPS
0
1
2
3
4
5
RMAT27
RMAT28
RMAT29
RMAT30
Workload
1S 2S 1S1G 2S1G 2S2G
PageRank
Billion EPS
0
1
2
3
4
5
RMAT27
RMAT28
RMAT29
Workload
1S 2S 1S1G 2S1G 2S2G
BC
Billion TEPS
0
1
2
3
RMAT27
RMAT28
RMAT29
Workload

Conclusions

- CSR representation not ideal
- Dependent on GPU memory
- Keniograph is a possibility
- New paradigm in graph computing