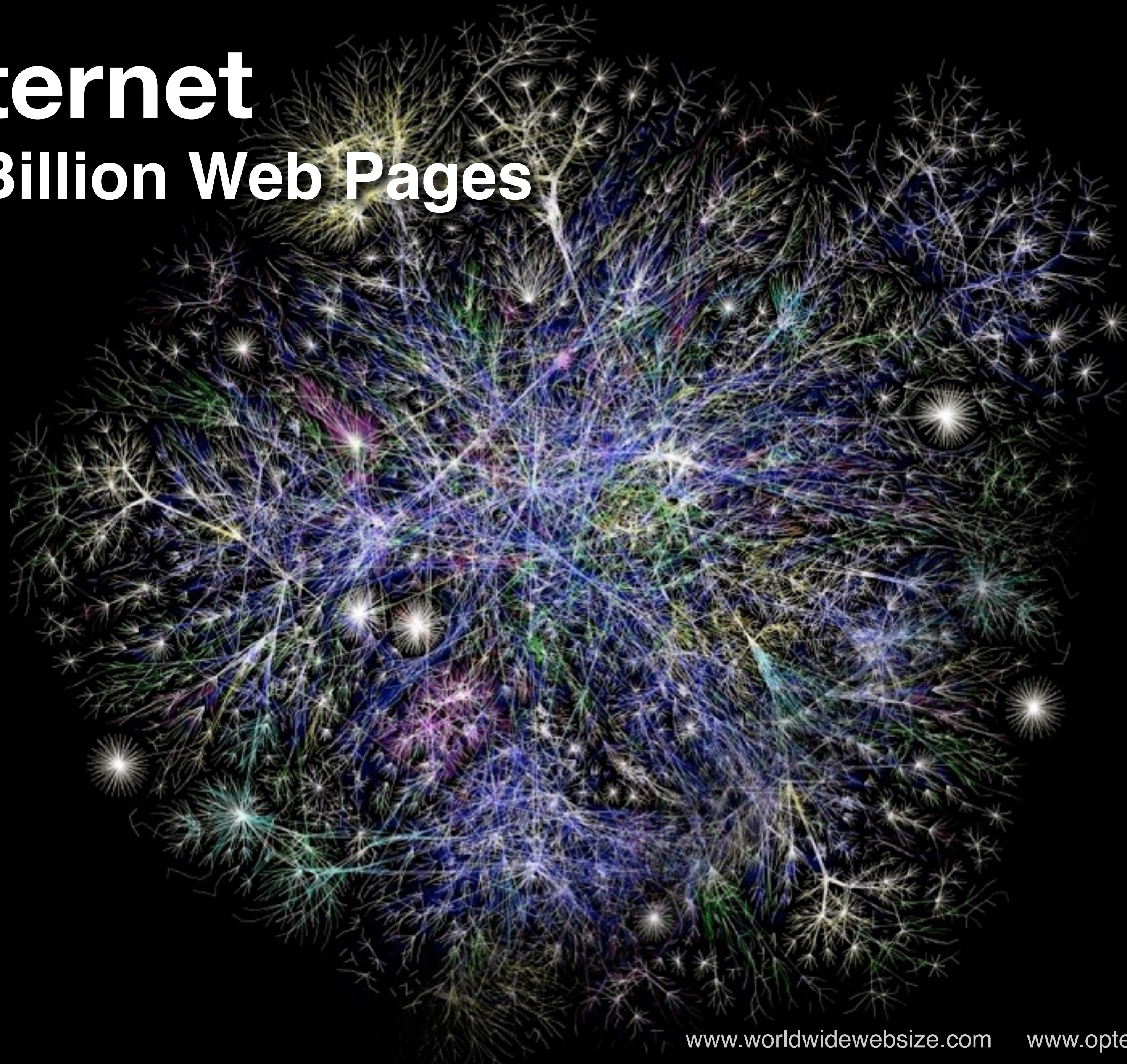
M-Flash:

Fast **Billion-Scale** Graph Computation Using a Bimodal Block Processing Model

Hugo Gualdron	University of Sao Paulo
Robson Cordeiro	University of Sao Paulo
Jose Rodrigues-Jr	University of Sao Paulo
Duen Horng (Polo) Chau	Georgia Tech
Minsuk Kahng	Georgia Tech
U Kang	Seoul National University
Dezhi "Andy" Fang Presenter	Georgia Tech

Internet

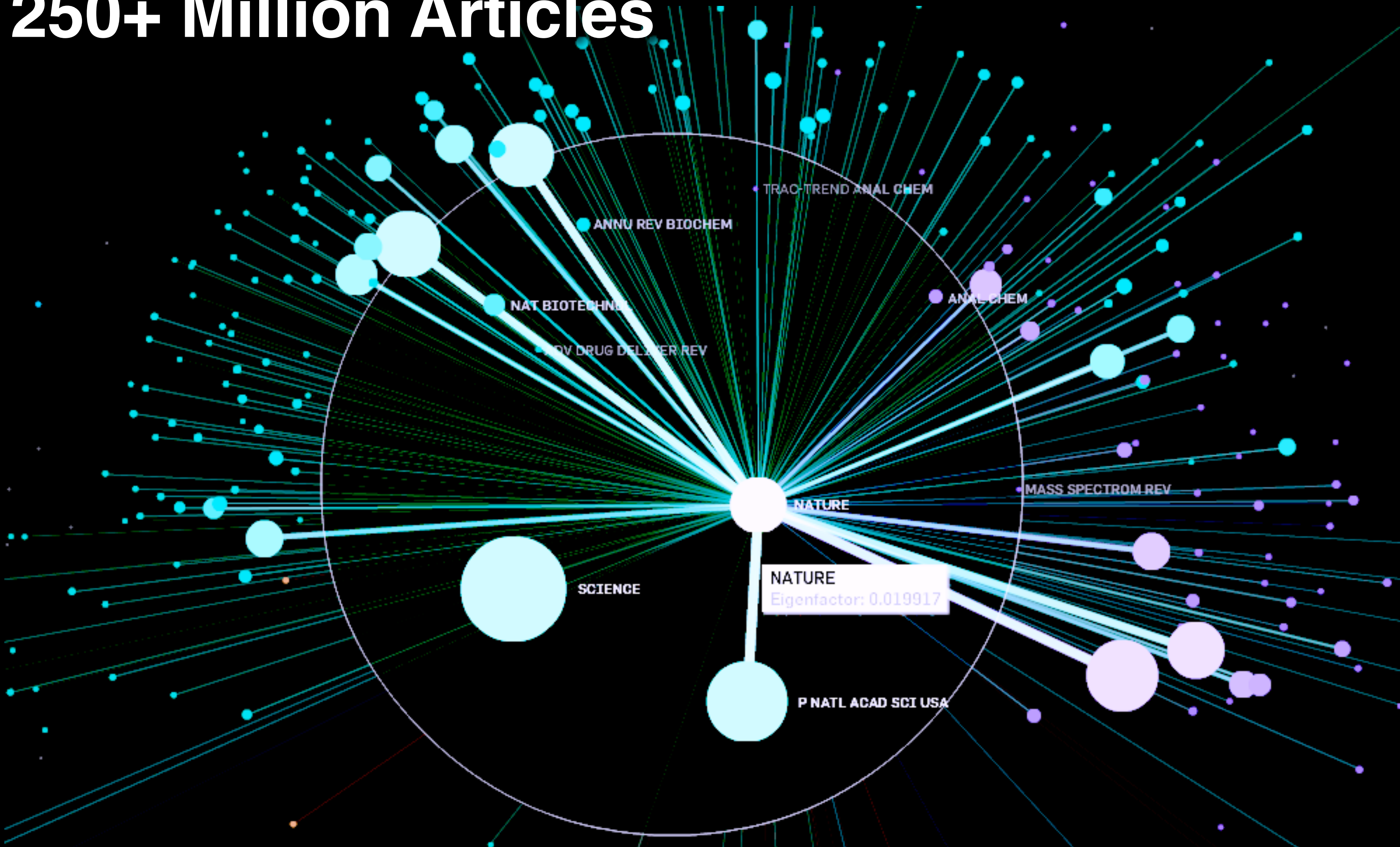
4+ Billion Web Pages



www.worldwidewebsize.com www.opte.org

Citation Network

250+ Million Articles



www.scirus.com/press/html/feb_2006.html#2   Modified from well-formed.eigenfactor.org

Many More

twitter

Who-follows-whom (310 million monthly active users)

amazon

Who-buys-what (300+ million users)

at&t cellphone network

Who-calls-whom (130+ million users)

Protein-protein interactions

200 million possible interactions in human genome

Sources: www.selectscience.net www.phonedog.com www.mediabistro.com www.practicalecommerce.com

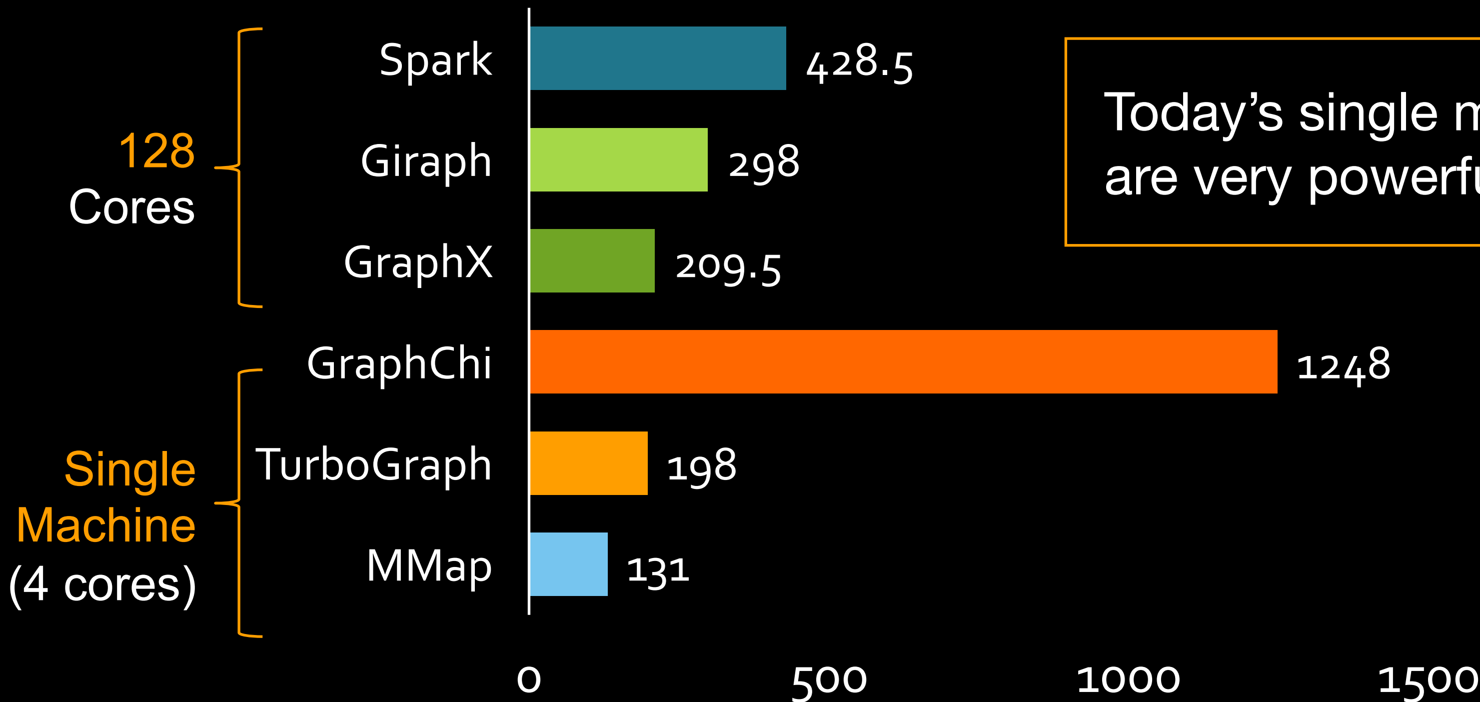
Large Graphs Are Common

Graph	Nodes	Edges
YahooWeb	1.4 Billion	6 Billion
Symantec Machine-File Graph	1 Billion	37 Billion
Twitter	104 Million	3.7 Billion
Phone call network	30 Million	260 Million

Takes Most Space

Scalable Graph Computation on Single Machines

PageRank Runtime (s) on Twitter Graph
(1.5 billion edges; 10 iterations, lower is better)



Today's single machines are very powerful.

Can we do even better?

McSherry, Frank, Michael Isard, and Derek G. Murray. "Scalability! But at what COST?." 15th Workshop on Hot Topics in Operating Systems (HotOS XV). 2015.
Lin, Zhiyuan, et al. "Mmap: Fast billion-scale graph computation on a pc via memory mapping." Big Data (Big Data), 2014 IEEE International Conference on. IEEE, 2014.

M-Flash:

Fast **Billion-Scale** Graph Computation Using a Bimodal Block Processing Model

Our Observation #1:
I/O is Bottleneck

Graph edges need to be stored on disk.

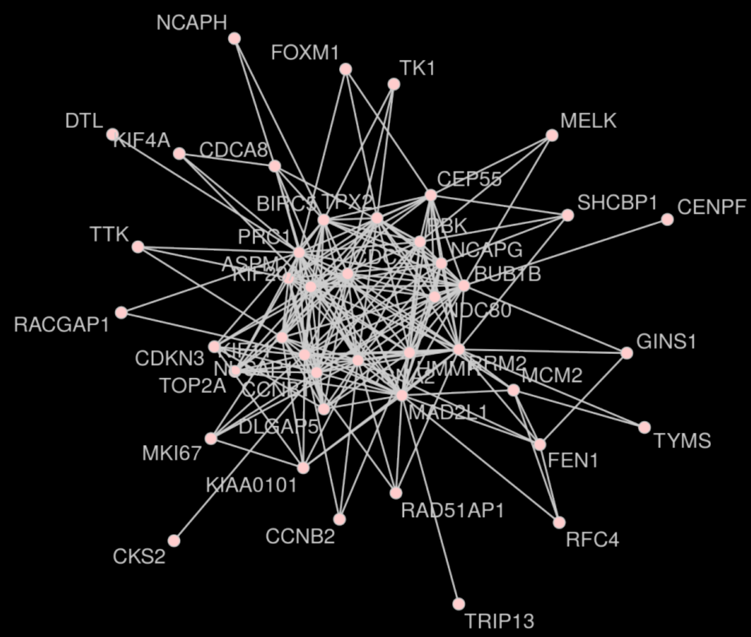
Symantec graph: 37 billion edges, 200+ GB

Disk access is much slower than RAM.

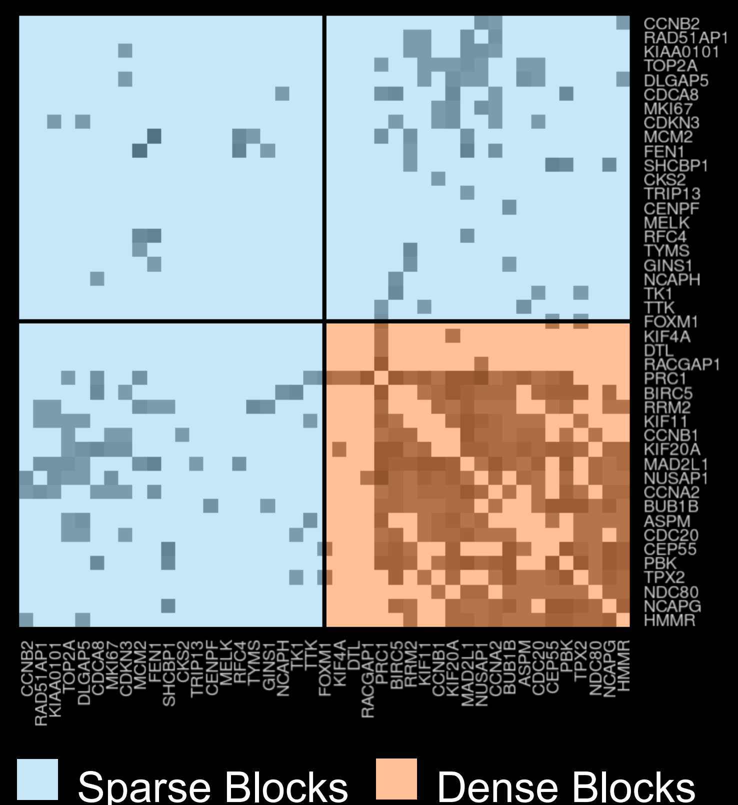
Goal: Reduce I/O, especially random accesses

Our Observation #2:
Real-world graphs are sparse.

Adjacency matrix contains dense and sparse blocks



https://web.stanford.edu/class/bios221/labs/networks/lab_7_networks.html

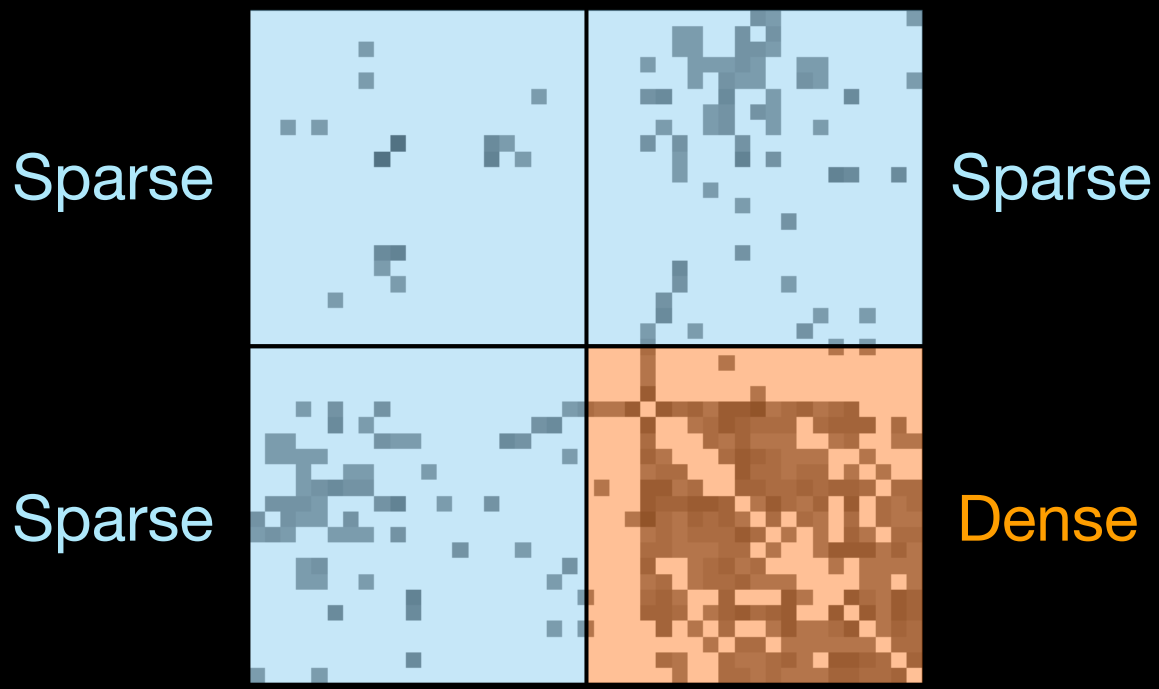


M-Flash's Solutions

1. Determine edge block types (dense and sparse)

2. Design efficient processing approaches for each block type

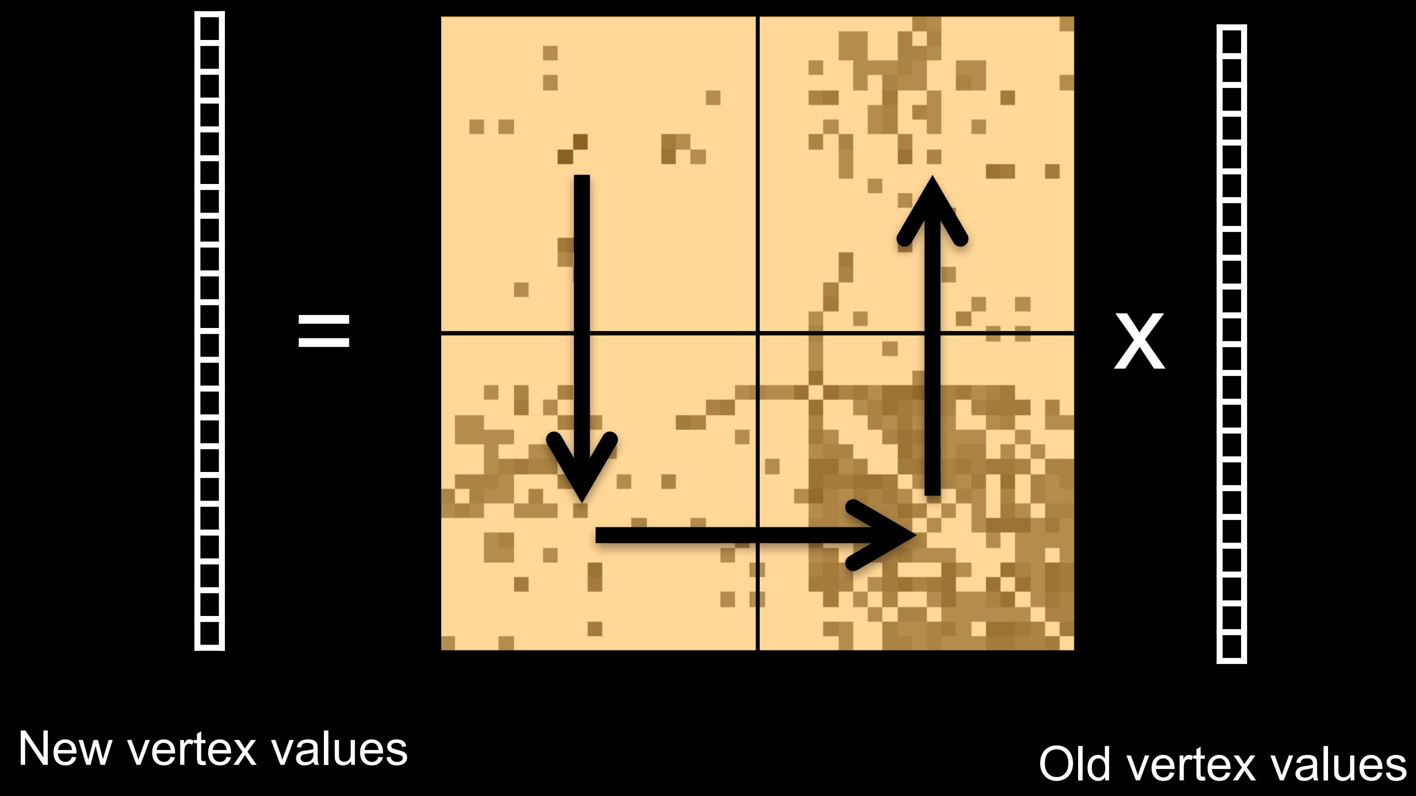
Determine Block Types In Pre-processing



$$\text{BlockType} = \begin{cases} \text{Sparse, if } \dfrac{\text{I/O cost if treated as Sparse}}{\text{I/O cost if treated as Dense}} < 1 \\ \text{Dense, otherwise} \end{cases}$$

Dense Block Processing

(Assuming all blocks are dense)



I/O Cost for Dense Block Processing

Each vertex is read β times and then written once

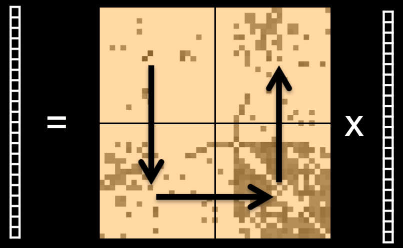
\# Vertex

\# Edge

$$O\left(\frac{(\beta+1)|V| + |E|}{B} + \beta^2\right)$$

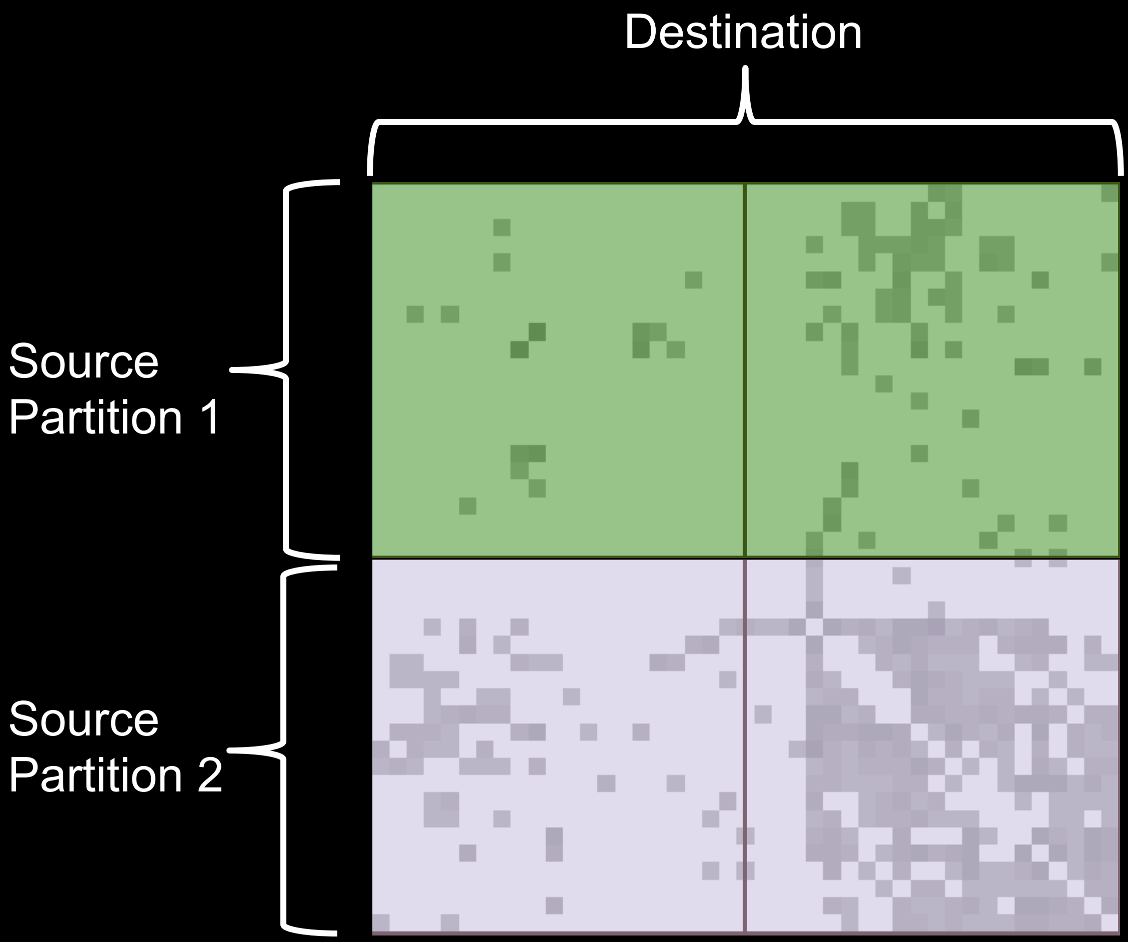
Size of per I/O Operation

#Interval (= #Row = #Column)



Sparse Block Processing

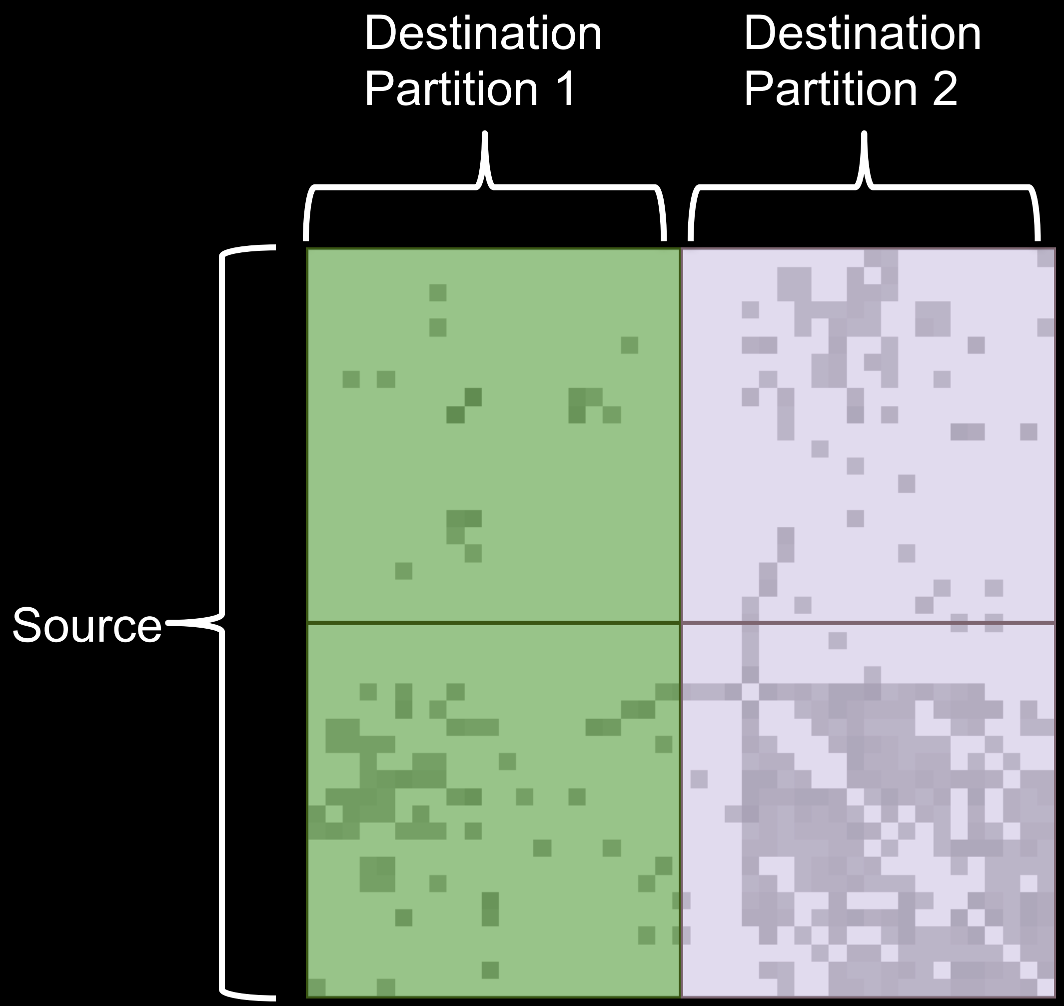
(Assuming all blocks are sparse)



Source Partition: Sequential Read

Sparse Block Processing

(Assuming all blocks are sparse)



Destination Partition:
Sequential Write

I/O Cost for Sparse Block Processing

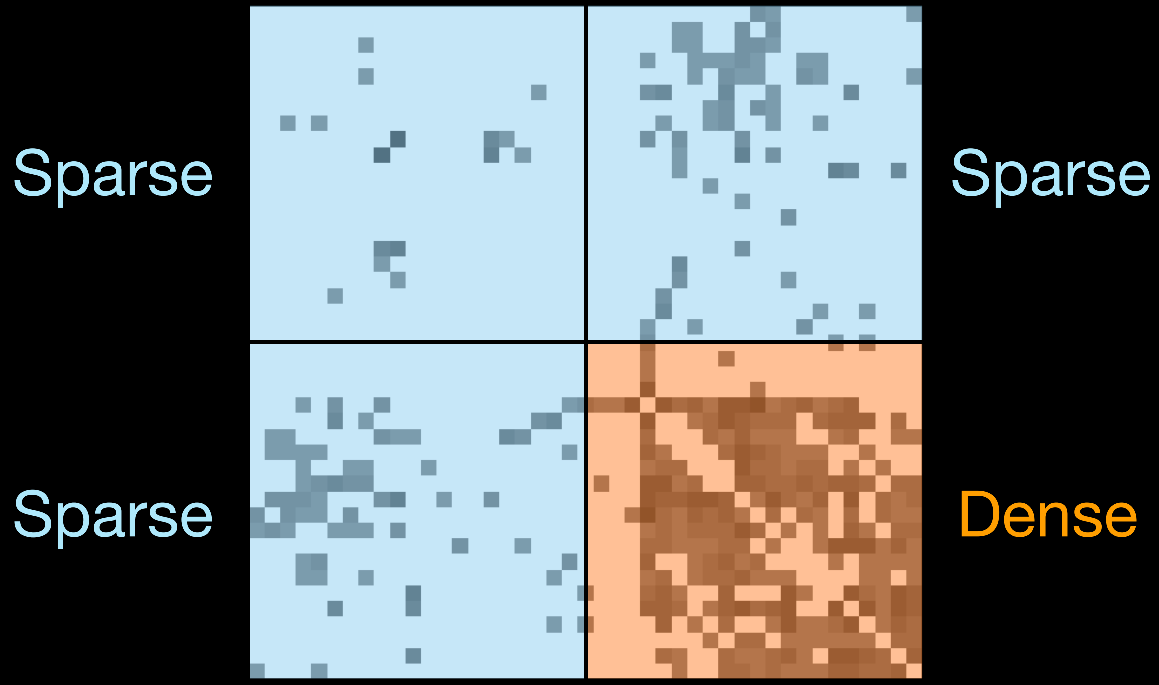
Vertex # Edge Edge with extended information

$$O\left(\frac{2|V| + |E| + 2|E_{extended}|}{B} + \beta^2\right)$$

Size of per I/O Operation

#Interval
(= #Row = #Column)

Bimodal Block Processing

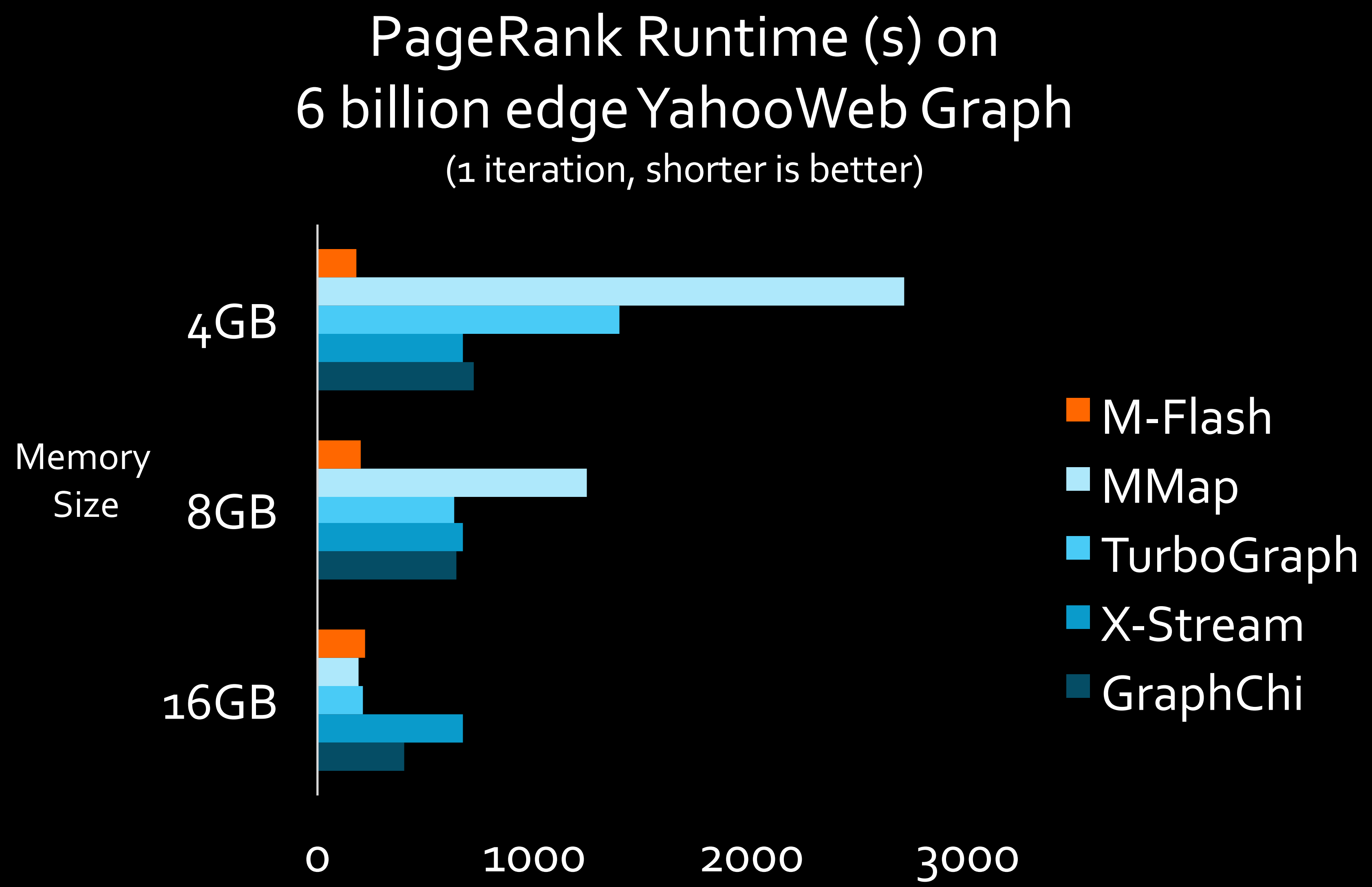


$$\text{BlockType} = \begin{cases} \text{Sparse, if } \dfrac{\text{I/O cost if treated as Sparse}}{\text{I/O cost if treated as Dense}} < 1 \\ \text{Dense, otherwise} \end{cases}$$

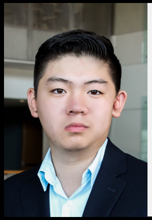
Large Graphs Used in Evaluation

Graph	Nodes	Edges
LiveJournal	5 Million	69 Million
Twitter	41 Million	1.5 Billion
YahooWeb	1.4 Billion	6.6 Billion
R-Mat (Synthetic)	4 Billion	12 Billion

Runtime of M-Flash



Thanks!



Dezhi "Andy" Fang
Georgia Tech CS Undergrad
http://andyfang.me

M-Flash:

Fast **Billion-Scale** Graph Computation Using a Bimodal Block Processing Model

- Fastest single-node graph computing framework
- Innovative bimodal design that addresses varying edge density in real-world graphs

- M-Flash Code: https://github.com/M-Flash/m-flash-cpp
- MMap Project: http://poloclub.gatech.edu/mmap/

CNPq (grant 444985/2014-0), Fapesp (grants 2016/02557-0, 2014/21483-2), Capes, NSF (grants IIS-1563816, TWC-1526254, IIS-1217559) GRFP (grant DGE-1148903), Korean (MSIP) agency IITP (grant R0190-15-2012)