PREGEL A SYSTEM FOR LARGE-SCALE GRAPH PROCESSING (2010)

Malewicz et al. **Pregel: A System for Large-Scale Graph Processing**

DAVID ADEBOYE, MPHIL ADVANCED COMPUTER SCIENCE

CHALLENGES IN LARGE GRAPHS PROCESSING

1. RELYING ON AN EXISTING DISTRIBUTED COMPUTING PLATFORM IS OFTEN ILL-SUITED FOR GRAPH PROCESSING
2. SINGLE-COMPUTER ALGORITHM LIBRARIES LIMIT THE SCALE OF PROBLEMS THAT CAN BE ADDRESSED
3. EXISTING PARALLEL GRAPH SYSTEMS DO NOT ADDRESS FAULT-TOLERANCE OR OTHER ISSUES THAT ARE IMPORTANT FOR LARGE SCALE DISTRIBUTED SYSTEMS

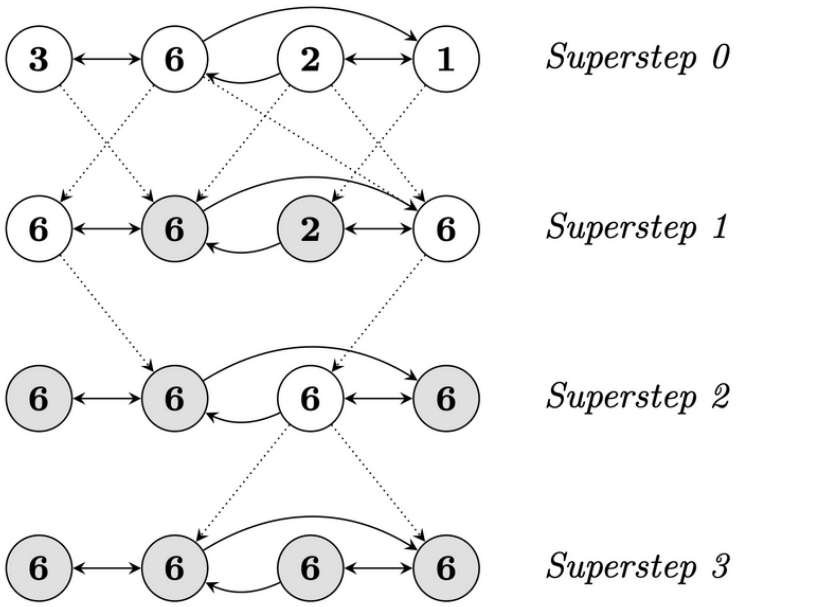
BULK SYNCHRONOUS PARALLEL MODEL

In Pregel, our iterations are called *Supersteps*, in these supersteps, the framework invokes a user-defined function on all each of the active vertices

During the computation, the vertices can exchange data with any other vertex. These messages will be available to the destination vertex at the beginning of the next iteration / superstep

---

Pregel also uses this model to implement fault-tolerance by checkpointing at the end of every superstep.



Leslie G. Valiant, **A Bridging Model for Parallel Computation.**

COMBINERS & AGGREGATORS

Two network-saving optimisations

- ## COMBINERS

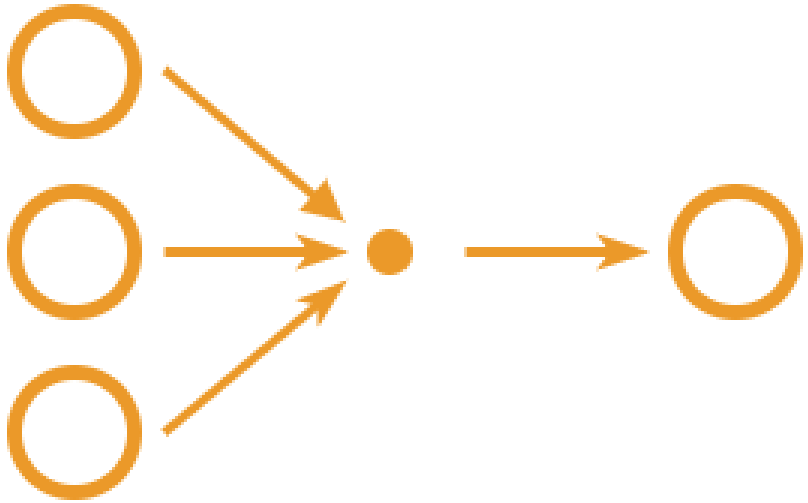
  Combine messages from other vertices

  Used to save network bandwidth

- ## AGGREGATORS

  Mechanisms for global communication

  Used for global coordination, monitoring and data



EVALUATION

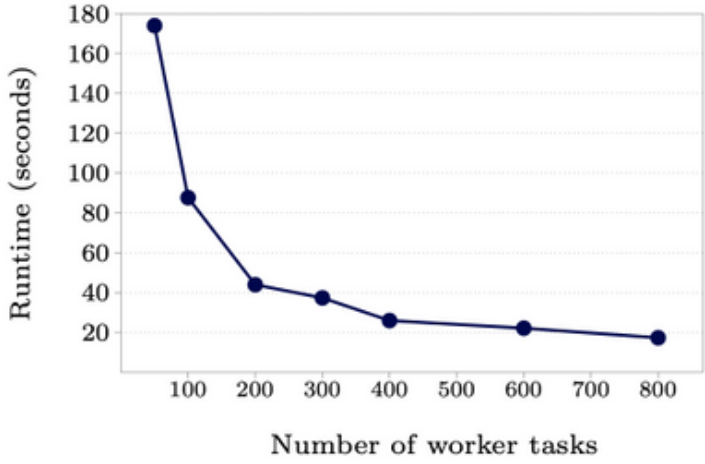
Experiments conducted on a cluster of 300 multicore commodity PCs, calculating the single-source shortest paths

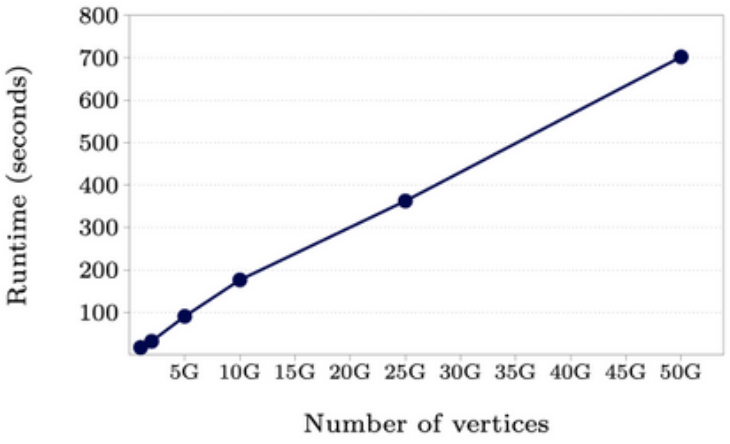
TOP: SSSP 1 billion vertex binary tree

BOTTOM: SSSP varying graph sizes on 800 worker tasks scheduled on 300 multicore machines

---

Even using a naïve implementation of SSSP implemented on Pregel, was comparable than the state-of-the-art system, with relatively little coding effort.





PAPER'S STRENGTHS AND CONTRIBUTIONS

- PROVIDES AN EASY API FOR VERTEX-CENTRIC GRAPH PROCESSING
- HAS LED TO THE DEVELOPMENT OF MANY PREGEL-LIKE SYSTEMS, INCLUDING APACHE GIRAPH, GPS, MIZAN, AND GRAPHLAB, ALL APPEARING WITHIN 3 YEARS OF THE PAPERS' PUBLICATION
- PREGEL EFFICIENTLY SCALES UP WITH GRAPH SIZE
- FAULT-TOLERANCE CLEARLY BUILT INTO THE SYSTEM, RATHER THAN AN AFTERTHOUGHT
- THE BSP SYSTEM REDUCES THE LIKELIHOOD OF RACE CONDITIONS AND DEADLOCKS
- THE OUTCOME OF EACH SUPERSTEP IS IMMEDIATELY KNOWN AND PROVIDES REAL-TIME PROGRESS OF THE ALGORITHM.

MODEL CRITICISMS

- USING A VERTEX-CENTRIX PROGRAMMING MODEL WITHOUT REGARD TO GRAPH PARTITIONING AND DATA LAYOUT ON DISK CAN LEAD TO UNNECESSARY I/O INITIALISATION AND A PERFORMANCE DROP
- MAPPING SHARED MEMORY GRAPH ALGORITHMS TO THIS MODEL IS NOT TRIVIAL AND REQUIRED NEW VERTEX-CENTRIC ALGORITHMS TO BE DEVELOPED
- COMMUNICATION BETWEEN TWO VERTICES CONNECTED BY AN EDGE USUALLY REQUIRES NETWORK I/O AS THE VERTICES MAY RESIDE ON DIFFERENT HOSTS
- MASSIVE GRAPHS CAN IMPOSE COORDINATION OVERHEADS ON THIS DEGREE OF PARALLELISM THAT MAY OUTWEIGH THE BENEFITS

Heidari et al. **Scalable Graph Processing Frameworks: A Taxonomy and Open Challenges**

PAPER CRITICISMS

- C++ API
- NO DIRECT COMPARISON TO EXISTING SOLUTIONS OR SINGLE MACHINE WORKLOADS
- STATIC PARTITIONING
- EVALUATION WAS INCOMPLETE & VAGUE
- CHECKPOINT RECOVERY COST IS HIGH

DISTRIBUTED GRAPH PROCESSING DEVELOPMENTS SINCE

1 RELEASE OF SEVERAL PREGEL-LIKE SYSTEMS

Apache Giraph (2013

Graph Processing System (2013)

Mizan (2013)

GraphLab (2013)

2 EDGE-CENTRIC PROCESSING

Roy et al. **X-Stream: Edge-centric Graph Processing using Streaming Partitions**

3 USE OF GPU-ACCELERATED PROCESSING

Merrill et al. **Scalable GPU Graph Traversal**

4 HARDWARE-ACCELERATED GRAPH PROCESSING

Ma et al. **Minnow: Lightweight Offload Engines for Worklist Management and Worklist-Directed Prefetching**

OPEN CHALLENGES

- ALL SYSTEMS USE MULTI-NODE CLUSTERS AND DO NOT EXPLOIT THE ELASTICITY PROPERTY OF THE CLOUD
- THE HETEROGENEITY OF DIFFERENT GRAPH ALGORITHMS MEANS THERE ISN'T A SINGULAR DOMINANT DISTRIBUTED ARCHITECTURE
- LACK OF A GOOD HIGH-LEVEL ABSTRACTION, COMPARABLE TO RESILIENT DISTRIBUTED DATASETS

THANKS FOR LISTENING