Pregel

A System for Large-Scale Graph Processing
Structure of the talk

I. Features of Pregel
II. Discussion of the method used
III. Future work
I. Features of Pregel:
   a) Computational Model (1/2)

   • A computation is a sequence of iterations
   • Each iteration is called a superstep
   • User defined function computed at each vertex in parallel
I. Features of Pregel:
   a) Computational Model (2/2)

   - Messages are passed between each iteration
   - Vertices can vote to halt
I. Features of Pregel:
b) Programmers point of view

• “Think as a Vertex”
• Aggregators: global communication
• Combiners: merge messages (for efficiency)
I. Features of Pregel: c) Implementation

- Mater/worker model
- Vertices are partitioned based on their ID
- Fault tolerance through checkpointing
- Workers buffer messages between supersteps
- Tree of workers for aggregators
II. Discussion of the method used
   a) Performance

   • Scales very well
   • Much faster than previous work
   • No evaluation of setup time or fault tolerance
II. Discussion of the method used
b) Computational model

• Does not adapt to the graph
• Not clear if it is designed for graphs
II. Discussion of the method used

c) Programming model

• “Think as a vertex” may not always be right
• Maybe this is a good thing
Future Work

• Partitioning based on the graph
• Handle complex parallelizable functions over the whole graph
• Avoid waiting for slow workers