ParaTimer: A Progress Indicator for MapReduce DAGs
Kristi Morton, Magdalena Balazinska, and Dan Grossman
University of Washington,
Department of Computer Science and Engineering
http://nuage.cs.washington.edu

Large Scale Data Analysis

Accurate Estimation is Challenging

ParaTimer Contributions

ParaTimer Approach (assuming no failures/data skew)

1. Compute Task Schedule
MapReduce DAG

2. Identify Tasks on Critical Path

3. Compute Time-Remaining of Critical Path
MapReduce tasks: series of blocking operations (pipelines)

1. Compute Task Schedule
MapReduce DAG

2. Identify Tasks on Critical Path

3. Compute Time-Remaining of Critical Path
MapReduce tasks: series of parallel operations (pipelines)

ParaTimer Approach for Failures: Set of Estimates

1. Std. Estimate: Critical path
   • On a failure, recompute critical path

2. Pessimistic Estimate: Failure has worst-case impact on overall time
   • Fails right before finishing and will fail when nothing else can run in parallel
   • On failure, recompute critical path and assume another failure

3. Failure Estimate: Same as Pessimistic Estimate
   • On a failure, recompute critical path: same as Std Estimate

ParaTimer Approach for Data Skew: Set of Estimates

1. Std. Estimate: critical path best guess

2. Upper Bound Estimate: chain n longest reduces together

3. Lower Bound Estimate: chain n shortest reduces together

Experimental results:
• Std Estimate: < 4% avg error
  • Bounds: < 17% avg error