THE SUITABILITY OF BSP/CGM MODEL FOR HPC ON CLOUDS

Alfredo Goldman, Daniel Cordeiro and Alessandro Kraemer
IME - USP
Cetraro, Italy, 28/06/2012
Great Workshop

• High Quality Presentations
• Amazing location
  • even without the old elevator
• Great face to face contacts
  • Jogging with Ian Foster
  • Histories of Steve Wallach
• Discussion about flash with Frank Baetke
• Talk on teamwork with Natalie Bates
  • ....
Distributed Systems

- Two main conferences
  - SBRC - Distributed Systems and Networks
    - 30th Edition
    - 1000 participants
  - SBAC-PAD - Computer Architecture and HPC
    - 24th Edition
    - Papers in English
    - 2012 Edition in New York
Back to the WIP

- Agenda
- Motivation
- Previous Experience
- Some Related Works
- Preliminary Experiments
- Future Work
Motivation

- Paper from HP labs
- Evaluation of HPC Applications on Cloud
  - A. Gupta and D. Milojivic
- Cloud would be suitable for some HPC apps
Main Points

- On the Cloud
  - poor network performance / OS noise
  - can be cost-effective
- Clouds are more cost-effective for:
  - Embarrassingly parallel/tree structured
  - Applications where comm. cost is hidden by computation
OTHER APPLICATIONS?

- Map Reduce
  - Widely spread with hadoop
  - Compared to BSP has limitations
    - (Pace - ICCS, 2012)
- How to deal with the Communication?
  - Try to “minimize” them...
Integrate

- www.integrate.org.br

Opportunistic Grid Middleware

- With support for Parallel Computing

- Bag of Tasks

- Either MPI and BSP
BSP

- Bulk Synchronous Parallel
- Valiant’90
- Model that links software and hardware
  - Given the machine parameters it is easy to estimate the execution time
BSP MAIN POINTS

- Execution performed in super-steps
  - Computation and synchronization phases
- Two communication mechanisms:
  - Direct Remote Memory Access (DRMA)
  - Bulk Synchronous Message Passing (BSPM)
- Several existing implementations
  - BSPLib, Green BSPLib, PUB, BSP-G
INTEGRATE - CHECKPOINTING

- Essential in opportunistic environments
- Checkpoints are stored periodically
- Using BSP
  - Checkpointing on InteGrade is portable and transparent to the programmer
Coarse Grained Model

- Theoretical model proposed by Dehne ’93
- \( n \) data size, \( p \) processors with memory \( O(n/p) \)
  - \( n/p \gg p \)
- At each step processors exchange \( O(n/p) \) data
- Goal: minimize the number of steps
CGM Algorithms

- Randomized List Ranking
  - $O(p \log n)$ with high probability
- All-Substrings longest common subsequence
  - $O(\log p)$
- Euler Tour
- Efficient ways to do the h-relation
- More than 10 thousand results on Google Scholar
Interest on large graphs

- Pregel (2010)
  - suitable for large-scale graph computing
  - Vertex centric approach
  - designed to be
    - efficient, scalable and fault-tolerant
PREGEL (1/2)

- Each process/core is assigned to one vertex
- Loop, for each vertex
  - Receive data from the previous step
  - Change state
  - Send data to other vertices
  - May vote to halt
Pregel (2/2)

- Was applied in clusters with thousands of commodity computers
- Applications:
  - Page Rank
  - Shortest Path
  - Bipartite-Matching
Apache Hama is a pure BSP computing framework on top of HDFS

For massive scientific computations such as matrix, graph and network algorithms

Computation Engines:

- Map Reduce - for matrix computations
- BSP, Dryad - for graph computations
SEVERAL OTHERS

- Apache Giraph
- GPS: Graph Processing System
  - API for global comm., load balancing & distribution
- Golden ORB
- Phoebus
- Bagel
Preliminary Results

- We have conducted some experiments with two classical graph problems:
  - Connected Components and Eulerian Path.
- With one twist: the MapReduce algorithm only tests if it exists a Eulerian Path and find a single connected component while the BSP computes the path and find all connected components.
Experimental Environment

- Private cloud
  - 11 Intel Core Duo 2.66 GHz, 2GBytes, interconnected by a FastEthernet network
  - The PCs are shared by 33 Virtual Machines
- Software used:
  - For BSP/CGM: mpich2, cgmlib 0.9.5 and NFS.
  - For MapReduce: sun java 5, hadoop 1.0.1 and HDFS.
Euler Tour

Euler tour - 1,000 trees and 500,000 nodes

- BSP/CGM
- MapReduce

Number of Virtual Machines vs. execution time (s)
Connected Components

Connected Components - 1,000 trees and 500,000 nodes

- BSP/CGM
- MapReduce

execution time (s)

Number of Virtual Machines

quinta-feira, 28 de junho de 2012
Communication times for BSP

BSP communication times for Euler Tour

BSP communication times for Connected Components

quinta-feira, 28 de junho de 2012
Future Directions

- Explore Scalability
- Apply Locality to place the BSP processes
- Use partial synchronization