HPC in the cloud computing era: challenges, models and tools.

Pascal Bouvry  University of Luxembourg, Luxembourg
Mateusz Guzek
Dzmitry Kliazovich
Johnatan E. Pecero
Valentin Plugaru
Sebastien Varrette

Andrei Tchernykh  CICESE Research Center, Mexico

Samee U. Khan  North Dakota State University, U.S.A.

Albert Y. Zomaya  University of Sydney, Australia
Grand Duchy of Luxembourg
The Grand Duchy of Luxembourg

- **Size:**
  - 2,586 km²

- **Population:**
  - ~ 500,000 inhabitants
  - of which ~ 43% foreigners
    - of which 50% Portuguese or Italian
  - plus 140,000 commuting

- **Capital:**
  - Luxembourg

- **Official languages:**
  - French, German and Luxembourgish
The University

• A new university
  – Created August 2003
  – The one and only in Luxembourg
  – Bologna process right from the start (Bachelor, Master, PhD)

• A multilingual university
  – Three languages (English, German, French)
  – Bilingual and trilingual degrees

• An international university
  – Employees from 20 countries
  – 53% foreign students from 95 countries
  – Over 50 general university agreements for student exchange with universities in Europe, Asia and America as well as 270 ERASMUS agreements for different programmes.
  – Bachelor students have to spend one semester abroad.
2 geographic sites
4 clusters: chaos+gaia, granduc, nyx.
→ 291 nodes, 2944 cores, 27.363 Tflops
→ 1042TB shared storage (raw capa.)
3 system administrators

>5 M USD (Cumul. HW Investment) since 2007
Motivation of the presentation

Ideas in air related to Cloud Computing

• Just pay as you go
• Just drop things in the cloud
• Scale up and down, sky is the limit
Plan

• Cloud
  – History
  – Definition

• Experimental testbed
  – IaaS overhead benchmark
  – Cloud energy model validation

• Greencloud, a cloud infrastructure model and simulator
  – Data center models
  – Network elements

• New CA - DAG model for cloud scheduling

• Conclusion and perspectives
HPC and Cloud

• Historically we had 2 communities:
  – HPC
    • How to benefit from concurrent resources for increasing performance
  – Distributed computing
    • How to remotely access resources, transactional world

• One first attempt to bridge those:
  – Grid computing
    • Public research centers joining forces
From Grids to Clouds

• Limitations of grids:
  – No real commercial focus (i.e. no clear billing)
  – Complex bundle of various public providers

• Cloud opportunities:
  – Offer coming from the big commercial players
  – Single point of contact providing SLA
  – Virtualization of resources
Cloud Computing

- Cloud computing market: $241 billion in 2020
- Main focus is on Software-as-a-Service (SaaS)

Business Benefits of Cloud Computing

- Reduced IT Costs
  - No upfront capital expenses
  - Convert fixed costs into variable costs
- No unused capacity (pay-as-you-go)
- Better ROA
- Fewer IT resources

- Business Agility
  - Time-to-market innovation
  - IT resources focused on innovation
  - Procurement accelerator
  - Bypass slow IT acquisition process
Cloud abstraction layers

- **Business process**
  - Business orchestration

- **Software**
  - Gmail
  - Office365

- **Platform**
  - Google cloud
  - Container based approach

- **Infrastructure**
  - Amazon web services
  - Virtual machines

- **Hardware**
  - Real hardware
  - Hosting
Visual Model Of NIST Working Definition Of Cloud Computing
http://www.csrc.nist.gov/groups/SNS/cloud-computing/index.html

Essential Characteristics
- Broad Network Access
- Rapid Elasticity
- Measured Service
- On-Demand Self-Service

Resource Pooling

Delivery Models
- Software as a Service (SaaS)
- Platform as a Service (PaaS)
- Infrastructure as a Service (IaaS)

Deployment Models
- Public
- Private
- Hybrid
- Community
Adoption Challenges

- Legal Issues
  - ISO / PSF Certification
- Audit Ability
- Service Level Agreement
- Security and Privacy Challenges
- Cloud Integration and Composition
- Quality of Service
HPC Today

- Computational Chemistry
  - Quantum Mechanics
- Computational Chemistry
  - Molecular Dynamics
- Computational Biology
- Structural Mechanics
  - Implicit

- Reservoir Simulation
- Rendering Ray Tracing
- Climate / Weather
  - Ocean Simulation
- Data Analytics
• Bridging number crunching and big data!
  – FLOPs but also #los, i.e. latency & bandwidth
HPC in the cloud

**Horizontal scalability**: perfect for replication/ HA (High Availability)
  → best suited for runs with minimal communication and I/O
  → usability for true parallel/distributed HPC runs?

**Cloud Data storage**
  → Data locality enforced for performance
  → Data outsourcing vs. legal obligation to keep data local
  → Accessibility, security challenges

"**Cost effectiveness**"
  → chaos+gaia usage: 11,154,125 CPU hours (1273 years) since 2007
  → 15,06M$ on EC2 cc2.8xlarge vs. 4 Me cumul. HW investment

**Virtualization layer impact on performance?**

Let’s check the virtualization and communication issues
Cloud Middleware Components: Hypervisors

- Hypervisor: core virtualization engine / environment
  - VM running under an hypervisor = *guest* machine
- 2 types of hypervisors
  - native (bare-metal) or hosted
Cloud Middleware Components: Hypervisors

- Native Hypervisors
  - Xen, KVM, ESXi, Hyper-V
- Hosted hypervisors
  - VMWare Fusion, VirtualBox

<table>
<thead>
<tr>
<th>Hypervisor:</th>
<th>Xen 4.0</th>
<th>KVM 0.12</th>
<th>ESXi 5.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host architecture</td>
<td>x86, x86-64, ARM</td>
<td>x86, x86-64</td>
<td>x86-64</td>
</tr>
<tr>
<td>VT-x/AMD-v</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Max Guest CPU</td>
<td>128</td>
<td>64</td>
<td>32</td>
</tr>
<tr>
<td>Max. Host memory</td>
<td>1TB</td>
<td>-</td>
<td>2TB</td>
</tr>
<tr>
<td>Max. Guest memory</td>
<td>1TB</td>
<td>-</td>
<td>1TB</td>
</tr>
<tr>
<td>3D-acceleration</td>
<td>Yes (HVM Guests)</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>License</td>
<td>GPL</td>
<td>GPL/LGPL</td>
<td>Proprietary</td>
</tr>
</tbody>
</table>

Deployment of the same Debian instance on a Grid
Benchmark

Selected to represent various use cases of HPC systems:

**HPCC** : new reference benchmark suit for HPC
  - includes HPL
  - 7 tests to stress CPU/disk/RAM/network usage

**Bonnie++** : a *file system* benchmarking suite
**IOZone** : cross-platform benchmark of *file operations*
  - read, write, re-read, re-write, read backwards/strided, mmap...

Jul 29, 2013
Pascal.Bouvry@uni.lu
Hardware Model

Node
PowerEdge R310

Processor
Intel® Xeon® X3430, 2.4 GHz, 8M Cache, Turbo

Memory
4GB Memory (2x2GB), 1333MHz Single Ranked UDIMM

Storage
500GB 7.2K RPM SATA 3.5in No Raid

Network
Broadcom 5709 Dual Port 1GbE NIC w/TOE iSCSI, PCIe-4

1 Gbps
1 Gbps
Resource and allocation model

3-tier model: Task, Virtual Machine, Hardware
Virtualization cost

- HPL (TFlops)
- DGEMM (GFlops)
- RandomAccess (GUPs)
- STREAM (MB/s)
Virtualization cost

IOzone 64MB file, 1MB record test
- stremi baseline
- taurus baseline
- stremi KVM
- taurus KVM
- stremi Xen
- taurus Xen
- stremi ESXi
- taurus ESXi

rewrite
read
reread
random_write
random_read
GreenCloud: A Packet-level Simulator of Energy-aware Cloud Computing Data Centers

http://gforge.uni.lu/greencloud
Resource Requirements of Cloud Applications

- Computing
- Network Bandwidth
- Communication delays (tolerance)
- Degree of interactivity
- Storage
Resource Requirements of Cloud Applications

- Computing
- Network Bandwidth
- Communication delays (tolerance)
- Degree of interactivity
- Storage
Cloud Computing Applications

Classification of Cloud Applications

<table>
<thead>
<tr>
<th>Cloud Application</th>
<th>Computing</th>
<th>Bandwidth</th>
<th>Low Communication Delay</th>
<th>Degree of Interactivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud gaming</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Video conferencing</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Online office</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Collaborative editing</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>CRM</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Remote desktop</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Cloud Synchronization</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Video streaming</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Cloud storage</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Cloud backup</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Voice conferencing</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>Social networking</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>HPC</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
</tbody>
</table>

H: High, M: Medium, L: Low
Cloud Computing Applications

- Traditional resource allocation and scheduling
  - Distribute incoming jobs to the pool of servers
  - Communication requirements and networking are not taken into account
GreenCloud: Data Center Architectures

- Supported data center architectures

Two/Three-tier data centers

Modular data centers
Scheduling in Data Centers

Network congestion!!!
Scheduling in Data Centers

Network is balanced !!!

July 29, 2013
Pascal Bouvry (pascal.bouvry@uni.lu)
GreenCloud: Data Center Architectures

• Future data center architectures
GreenCloud: Data Center Architectures

GreenCloud Architecture

Data Center

Cloud User

Workload Generator

Workload Trace File

Task Schedulers

Data Center Characteristics

L3 Switch

L3 Energy model

Send Agent

Receive Agent

Connect ()

L2 Energy model

Rack Switch

Computing Server

Server Energy model

Scheduler

Server Characteristics
GreenCloud: Simulator Components

• Servers
  – Responsible for task execution
  – Single/multi-core nodes
  – Preset processing limit in MIPS or FLOPS
  – Preset RAM/Disk configuration

• Supported power management modes
  – DVFS: Dynamic Voltage/Frequency Scaling
  – DNS: Dynamic Shutdown (or stand-by)
  – Both: DNS if server is idle, DVFS otherwise
GreenCloud: Simulator Components

- Energy Model for Hosts

\[ P = P_{\text{fixed}} + P_f \cdot f^3 \]

Idle servers consume 65% of the peak load for all CPU frequencies

July 29, 2013

Pascal Bouvry (pascal.bouvry@uni.lu)
GreenCloud: Simulator Components

• Switches
  – Most common Top-of-Rack (ToR) switches typically operate at Layer-2 interconnecting gigabit links in the access network
  – Aggregation and core networks host Layer-3 switches operating at 10 GE (or 100 GE)

• Links
  – Transceivers’ power consumption depends on the quality of signal transmission in cables and is proportional to their cost
  – 1 GE links: 0.4W for 100 meter transmissions over twisted pair
  – 10 GE links: 1W for 300 meter transmission over optical fiber

• Supported power management modes
  – DVFS, DNS, or both
GreenCloud: Simulator Components

- Energy model for a network switch

\[ P_{\text{switch}} = P_{\text{chasis}} + n_{\text{linecards}} \cdot P_{\text{linecard}} + \sum_{i=0}^{R} n_{\text{portsr}} \cdot P_r \]

Chassis  
~ 36%

Linecards  
~ 53%

Port transceivers  
~ 11%
GreenCloud: Simulator Components

• Workloads
  – Model cloud user applications (social networking, instant messaging, content distribution, etc.)

• Workload properties
  – Computational: MIPS, duration
  – Storage: memory usage
  – Communicational: internal and external communications characteristics

• Generation
  – Trace-driven
  – Using random distribution
GreenCloud: Simulation Results

Data center
503kW·h

- Servers: 355kW·h (70%)
- Access switches: 75.6kW·h (15%)
- Aggregation switches: 1.74kW·h (10%)
- Core switches: 0.87kW·h (5%)

Switches

- Linecards: 53%
- Chassis: 36%
- Port tranceivers: 11%

Computing Servers
301 W

- CPU: 130W (43%)
- Motherboard: 25W (8%)
- Other: 48W (16%)
- Disks: 12W (4%)
- Memory: 36W (12%)
- Peripheral: 50W (17%)

Linecards: 53%
Port tranceivers: 11%
Chassis: 36%
CPU: 130W (43%)

July 29, 2013
Pascal Bouvry (pascal.bouvry@uni.lu)
GreenCloud Innovative Solutions

Energy-Efficient Network-Aware Scheduling

- Placing computing jobs to where it will take less energy
- Balance between energy and performance
- IEEE/ACM GreenCom [Best paper award]

DENS is architecture specific

\[ M = \alpha \cdot f_s + \beta \cdot f_r + \gamma \cdot f_m \]
GreenCloud Innovative Solutions


#1 • Treat communication and computing demands equally

#2 • Optimize energy efficiency and load balancing of network traffic

#3 • Formal model for selection of servers, racks, and network modules

July 29, 2013
Pascal Bouvry (pascal.bouvry@uni.lu)
GreenCloud Usage and Benefits

• GreenCloud tools cover complete optimization workflow

  1: Client data center analysis
  2: Applying optimization solutions
  3: Validation and proof of concept

• Can be used to
  – Optimize existing data centers
  – Guide capacity extension decisions
  – Help to design future data center facilities

July 29, 2013
Pascal Bouvry (pascal.bouvry@uni.lu)
GreenCloud: Screenshots
GreenCloud: Screenshots

```
encloud@greencloud:~/greencloud
$ cd greencloud/
encloud@greencloud:~/greencloud$ ./run

***************
BUILDING TOPOLOGY
***************
Data center architecture: three-tier debug
Creating switches CORE(2) AGGREGATION (4) ACCESS(6)...
Creating 30 servers...
Creating 1 cloud user(s)...

***************
SIMULATION PARAMETERS
***************
Simulation time: 60.0 seconds
Data center computing capacity: 30000030 MIPS
Power management of computing servers: No
Power management of network switches: No
Progress to  0 %
Progress to 10 %
Progress to 20 %
Progress to 30 %
Progress to 40 %
Progress to 50 %
Progress to 60 %
Progress to 70 %
Progress to 80 %
Progress to 90 %```
GreenCloud: Screenshots

Simulation Results

Summary

Simulation Duration (sec.): 60.0
Datacenter Architecture: three-tier
Switches (core): 2
Switches (agg.): 4
Switches (access): 3
Servers: 30
Users: 1
Power Mgmt. (servers): No
Power Mgmt. (switches): No
Datacenter Load: 28.7 %
Average Load/Server: 0.3 %
Total Tasks: 521
Average Tasks/Server: 521
Total Energy: 417.5 W*h
Switch Energy (core): 96.0 W*h (23%)
Switch Energy (agg.): 191.9 W*h
Switch Energy (access): 14.9 W*h
Server Energy: 114.7 W*h

Overview

Energy Summary
Total: 417.5 W*h

Server Energy 114.7 W*h (27%)
Switch Energy (core) 96.0 W*h (23%)
Switch Energy (agg.): 191.9 W*h (46%)
Switch Energy (access): 14.9 W*h (4%)

Data Center
Modeling Cloud Computing Applications

CA-DAG
Modeling of Cloud Applications

• Directed Acyclic Graphs (DAGs)
  – Vertices represent computing tasks of a job
  – Edges represent task dependencies and order of execution

How to model communication processes?
Modeling of Cloud Applications

- Communication-unaware model
- Edges-based model
Modeling of Cloud Applications

• Communication-unaware model
  – Each vertex represents both computing and communication processes of a task

• Main drawback
  – Having a single vertex for both computing and communications makes it impossible to make separate scheduling decisions
Modeling of Cloud Applications

• Edge-based model
  – DAG edges represent communication processes of a task

• Main drawback
  – Two different computing tasks cannot have the same data transfer to receive input as a single edge cannot lead to two different vertices
Proposed Communication-Aware DAG model
Modeling of Cloud Applications

• Proposed CA-DAG: Communication-Aware DAG model
  – Two types of vertices: one for computing and one for communications
  – Edges show define dependences between tasks and order of execution

• Main advantage
  – Allows separate resource allocation decisions, assigning processors to handle computing jobs and network resources for information transmissions
Modeling of Cloud Applications

• Proposed CA-DAG: Communication-Aware DAG model

  – Represented by a directed acyclic graph

  – Set of vertices is composed of computing tasks and communication tasks

  – A computing task is described by a pair with the number of instructions (amount of work) that has to be executed within a specific deadline

  – A communication task is described by parameters and defined as the amount of information in bits that has to be successfully transmitted within a predefined deadline

  – The set of edges consists of directed edges representing dependence between node and node
Modeling of Cloud Applications

• Example of webmail cloud application
  – Step 1: Receive user request and process it
  – Step 2: Generate personalized advertisement
  – Step 3: Request list of email messages from database
  – Step 4: Generate HTML pages and send it to the user
CA-DAG: Communication-Aware DAG

- **Step 1:** Receive user request and process it
- **Step 2:** Generate personalized advertisement
- **Step 3:** Request list of email messages from database
- **Step 4:** Generate HTML pages and send it to the user
CA-DAG: Communication-Aware DAG

Step 1: Receive User Request
Step 2: Process User Data
   - Analyze user social profile
Step 3: Request Database
   - Retrieve Personalized Ad
Step 4: Generate list of email messages
   - Group conversations
Step 5: Generate HTML page
Step 6: Send output to user

Legend:
- Computing task
- Communication task

Network
Processor

Makespan: 7

July 29, 2013
Pascal Bouvry (pascal.bouvry@uni.lu)
CA-DAG: Communication-Aware DAG

- Communication unaware model
  - Step 1: Receive user request and process it
  - Step 2: Generate personalized advertisement
  - Step 3: Request list of email messages from database
  - Step 4: Generate HTML pages and send it to the user
CA-DAG: Communication-Aware DAG

- Communication unaware model

Step 1

0. Receive User Request
1. Process User Data

Step 2

2. Analyze user social profile
3. Retrieve Personalized Ad

Step 3

4. Request Database
5. Generate list of email messages
6. Group conversations

Step 4

7. Generate HTML page
8. Send output to user

Network

0. Receive User Request
3. Receive User Request
4. Receive User Request

Processor

3. Receive User Request
4. Receive User Request
8. Send output to user

Makespan: 9
CA-DAG: Communication-Aware DAG

- Edges-based model
  - Step 1: Receive user request and process it
  - Step 2: Generate personalized advertisement
  - Step 3: Request list of email messages from database
  - Step 4: Generate HTML pages and send it to the user

July 29, 2013

Pascal Bouvry (pascal.bouvry@uni.lu)
CA-DAG: Communication-Aware DAG

- Edges-based model

0. Receive User Request

1. Process User Data

2. Analyze user social profile

3. Retrieve Personalized Ad

4. Request Database

5. Generate list of email messages

6. Group conversations

7. Generate HTML page

8. Send output to user

Network:
- 0. Receive User Request
- 4. Request Database
- 3. Retrieve Personalized Ad
- 4. Request Database

Processor:
- 4. Request Database
- 3. Retrieve Personalized Ad
- 4. Request Database

Makespan: 8

July 29, 2013
Pascal Bouvry (pascal.bouvry@uni.lu)
CA-DAG: Communication-Aware DAG

- Comparison of schedules

CA-DAG model

Communication-unaware model

Edges-based model
CA-DAG: Communication-Aware DAG

• Comparison of models’ makespan

<table>
<thead>
<tr>
<th># of Processors</th>
<th># of Network links</th>
<th>Communication-unaware model</th>
<th>Edges-based model</th>
<th>Proposed CA-DAG model</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>9</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>9</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>7</td>
<td>8</td>
<td>7</td>
</tr>
</tbody>
</table>

Achieves minimum makespan with the least resources
Properties of Communication Tasks/Vertices
Properties of Communication Tasks/ Vertices

- Task parallelization
- Multipath routing
- Task completion time
- Available bandwidth
Task Parallelization

• Each communication task/vertex can be divided into different independent communication tasks that can be executed in parallel.

• The smallest size of communication task is one bit as all bits in the message are independent.

Diagram:

- Original task graph:
  - 1
  - 2 (connected to 3 and 4)

- Parallelized task graph:
  - 1
  - 2.1, 2.2, ..., 2.n (connected to 3 and 4)
Multipath Routing

• Most of existing solutions rely on static network topology and fixed pre-allocation which implies circuit switching and pre-defined routing.

• In reality, datacenter networks are packet switched with routing decisions taken at every hop.

• The availability of multiple paths is essential to benefit from parallelization property of communication tasks.
Task Completion Time

• Execution of communication task involves
  – Packet transmissions on multiple links
  – Sequential processing, variable bitrates

• Communication delay components
  – Processing delay
  – Queuing delay
  – Transmission delay
  – Propagation delay

\[ d_{comm} = \sum_{i=1}^{N} (d_{proc}^i + d_{queue}^i + d_{tx}^i + d_{prop}^i). \]
Available Bandwidth

- Residual Bandwidth
  - Bandwidth left unoccupied

- Available bandwidth
  - Bandwidth that a new flow can obtain (residual bandwidth + portion of the used bandwidth)

- Utilization performance of communication protocols
  - TCP throughput

\[
B(p) = \frac{MSS}{RTT \cdot \sqrt{p}},
\]
Efficiency of CA-DAG Model
Efficiency of CA-DAG model

- System Architecture
  - Only one node can communicate at a time
Efficiency of CA-DAG model

• Workloads
  – Winkler graph generator
  – DAGs with occasional and frequent communications

• Communication-to-Computation Ratio (CCR)
Efficiency of CA-DAG model

• Scheduling Algorithm
  – Offline (deterministic) scheduling
  – Zero release time of DAGs
  – Clairvoyant execution and communication time
  – Adapted list scheduling is employed
  – A processor allowing minimum execution time is selected
Efficiency of CA-DAG model

• Scheduling Criteria
  
  – Schedule efficiency
  
  – Approximation factor
Efficiency of CA-DAG model

Schedule efficiency:

Ratio of sequential execution time to the makespan by the number of computing resources

\[
\text{eff}(S) = \frac{\sum_{i=1}^{n} p_i}{C_{\text{max}} \times m}
\]
Efficiency of CA-DAG model

• Schedule efficiency
  - Apps. with occasional communications
  - Apps. with frequent communications

![Graph showing efficiency comparison between CA-DAG, Comm-unaware DAG, and Edge-based DAG for different CCR values.](image)

The higher the better

---

Efficiency (%)

Communication-to-Computation Ratio (CCR)

- CA-DAG
- Comm-unaware DAG
- Edge-based DAG

July 29, 2013

Pascal Bouvry (pascal.bouvry@uni.lu)
Efficiency of CA-DAG model

\[ \rho = \frac{C_{\text{max}}}{C^*_{\text{max}}} \]

\[ C^*_{\text{max}} \geq \tilde{C}_{\text{max}}^{*} = \max \left\{ \max (blevel(t_i)), \frac{\sum_{i=1}^{n} (p_i)}{m} \right\} \]
Efficiency of CA-DAG model

- Approximation factor

  - Apps. with occasional communications
  - The lower the better

  - Apps. with frequent communications
  - The lower the better

July 29, 2013
Pascal Bouvry (pascal.bouvry@uni.lu)
CA-DAG model

• Cloud applications use communication resources excessively

• New communication-aware model of cloud applications, named CA-DAG, is proposed

• CA-DAG includes separate vertices to represent communication processes to allow making separate resource allocation decisions (computing jobs to processors, communication jobs to the network)

• CA-DAG model enables the design of novel solutions with mixed scheduling policies optimized for cloud computing
Conclusion

• We have
  – Benchmarked classical hypervisors
  – Highlighted the communication issue
  – Proposed a new cloud simulator called Greencloud
  – Proposed an enhanced DAG model called CA-DAG

• Cloud computing is there but comes at a cost.
Perspectives and other aspects

• New generations of VMs and HPC PaaS
• Network coding
• Hybrid cloud (public/private) solutions and cloud brokering
• New generation HW, mixing ARMGs and GPUs
• Legal and security aspects
Thank you!

Contact information:

Pascal Bouvry
University of Luxembourg
Pascal.Bouvry@uni.lu