The 2018 International Conference on High Performance Computing & Simulation (HPCS 2018)

Special Session on Virtualization in High Performance Computing and Simulation (VIRT 2018)

XVI Workshop de Processamento Paralelo e Distribuído (WSPPD)

Understanding and Minimizing Disk Contention Effects for Data-Intensive Processing in Virtualized Systems

Kassiano Matteussi[1], Claudio Geyer[1].

kjmatteussi@inf.ufrgs.br

[1]Federal University of Rio Grande do Sul, Porto Alegre, Brazil

Sep, 2018

UFRGS

INSTITUTO DE INFORMÁTICA UFRGS

Agenda

- **Introduction**
  - Problem statement and motivation
- **Data-intensive Processing and Virtualization**
- **Disk I/O Characterization**
- **Disk I/O Contention and Resource Allocation**
- **Towards a Dynamic Disk Resource Allocation**
  - Proposal, evaluation and results
- **Conclusion and Future Work**
- References

Introduction

“The information attracts a great deal of attention in recent years and led organizations to find out ways to handle data processing in an efficient, reliable and cost-effective way”

“The emergence of many data-processing frameworks”

Introduction

Problem Statement

- **Virtual datacenters**
  - Hosts many virtual machines
- **Shared environment**
  - Shared pool of resources: CPU, Memory, Network and disk
- **Resource contention**
  - Applications requesting resources at the same time
- **Resource contention**
  - **Disk contention problem**

Introduction

Disk contention problem

Interference implications

- Performance degradation for applications and frameworks
- Inter-VM resource interference
- Unexpected completion time of applications

Related Works

Some investigated works

- CPU and memory solutions
- Hypervisor based virtualization
- YARN (CPU and memory)
- Instrumented frameworks and applications

Opportunities

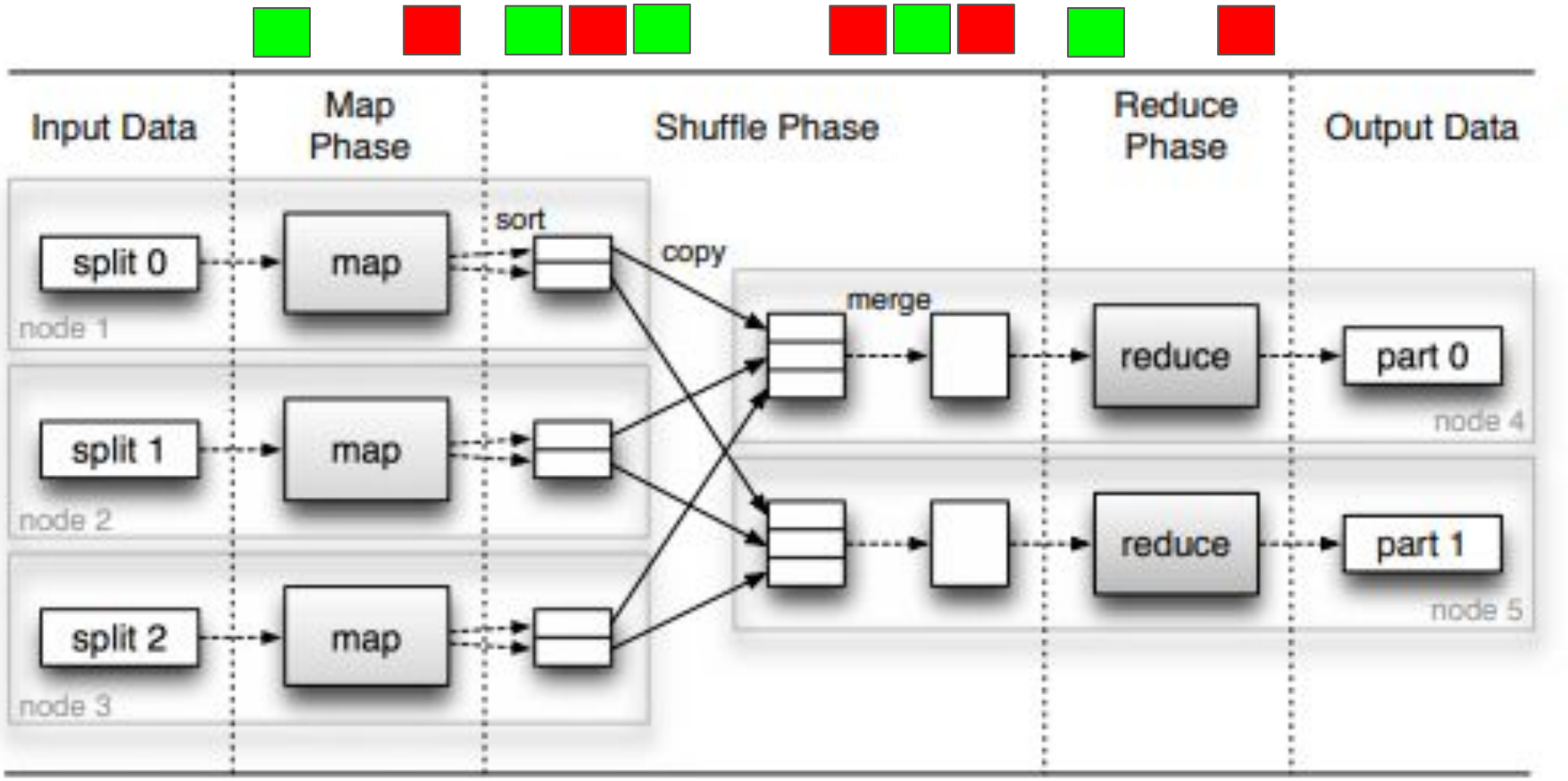
- Container-based virtualization
- Already solved problems in hypervisor-based virtualization are open again!

Introduction

Proposal

This work proposes a technique to minimize disk contention effects in order to improve applications' performance in virtualized systems.

Data-intensive Processing and Virtualization

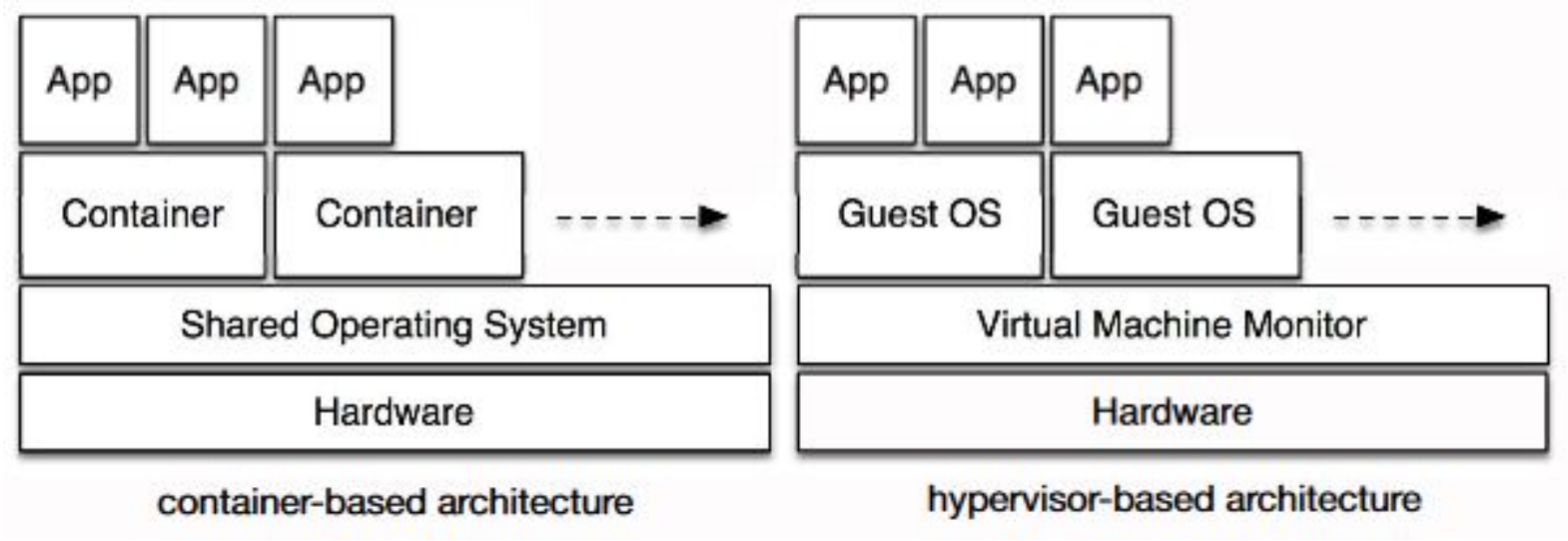


MapReduce data flow [1]

Read Operations

Write Operation

Data-intensive Processing and Virtualization



Comparison of container- and hypervisor-based virtualization architectures [2]

Today, the container-based virtualization is widely used for data-intensive processing

Disk I/O Characterization

Metrics

We observed some I/O characteristics such as data generation process, disk read/write bandwidth, data access patterns and application completion time.

Disk I/O Characterization

Testbed

- A real cluster consisting of 4 identical servers, each equipped with 8 x86 64 cores at 2.27 GHz, 16GB of RAM and one disk (160GB) per node.
- Hadoop MapReduce (1.1.2), the most popular open-source implementation for Big Data analysis.
- HiBench, a realistic benchmark suite based on the MapReduce programming model with real-world applications.

Disk I/O Characterization

WORKLOAD CHARACTERISTICS

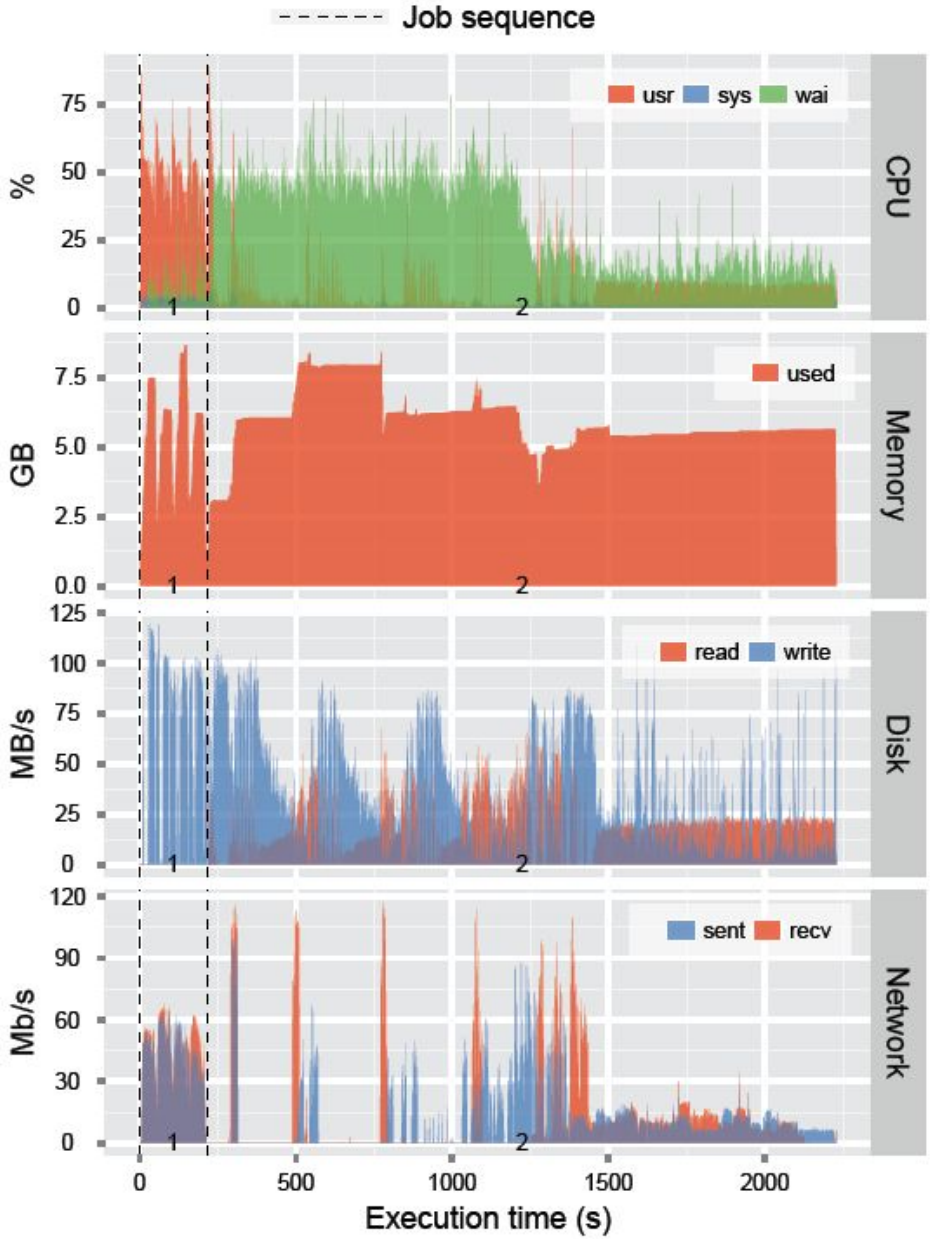
Application	Load in	Load out	Time (s)
Sort	15,9Gb	15,1Gb	2225
Terasort	37,2Gb	37,2Gb	1265
WordCount	11,4Gb	1Mb	2286
Nutch	509Mb	208,4Mb	266
Pagerank	1,3Gb	476Mb	160
K-means	46,4Gb	57Gb	3038
Bayes	352Mb	1Mb	401
Hive	5Gb	1,7Gb	788

Performed applications

Disk I/O Characterization

Sort - a disk intensive application

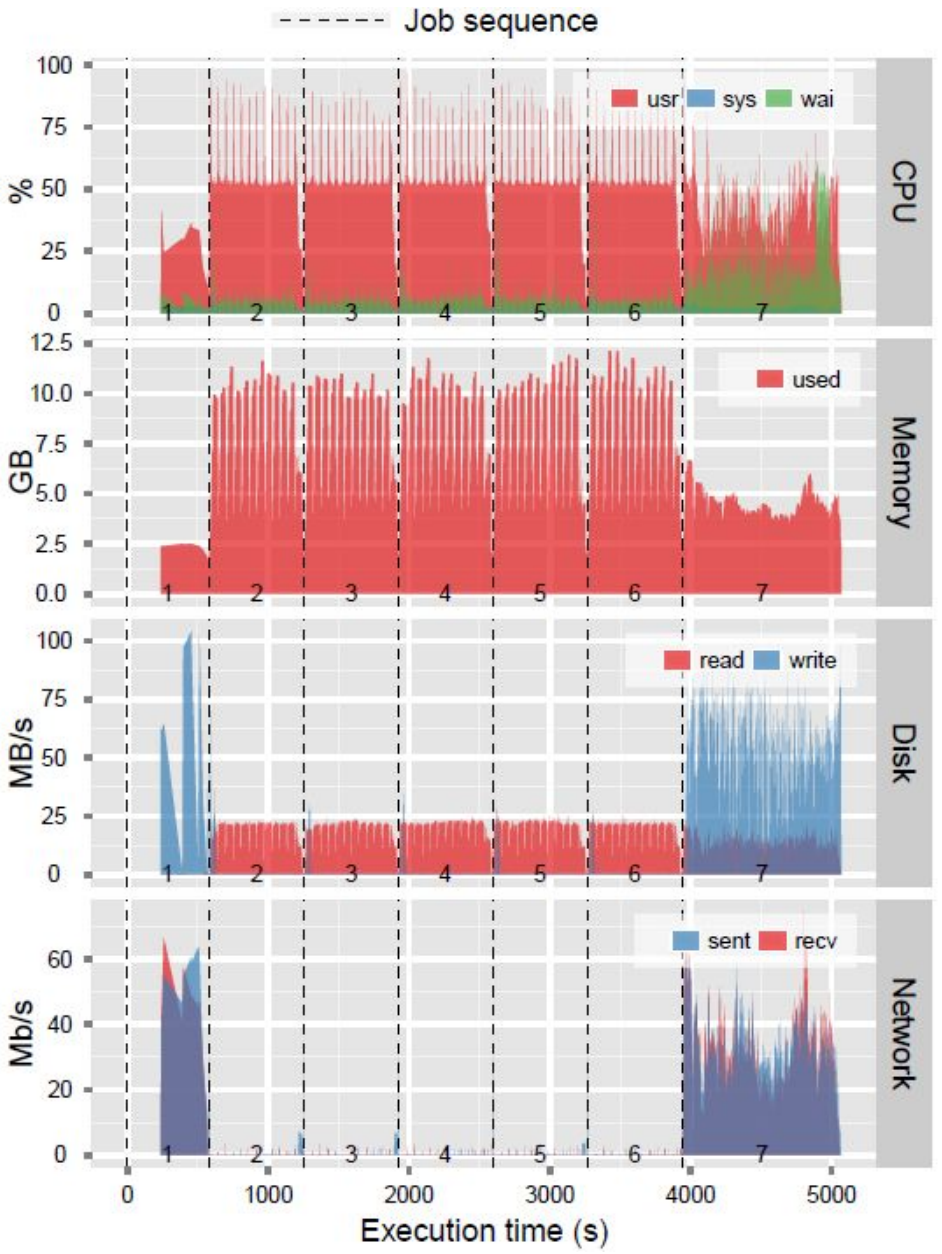
- Well-defined resource patterns
- Overlapping
- Resource correlation



Disk I/O Characterization

Kmeans - a non-disk intensive application

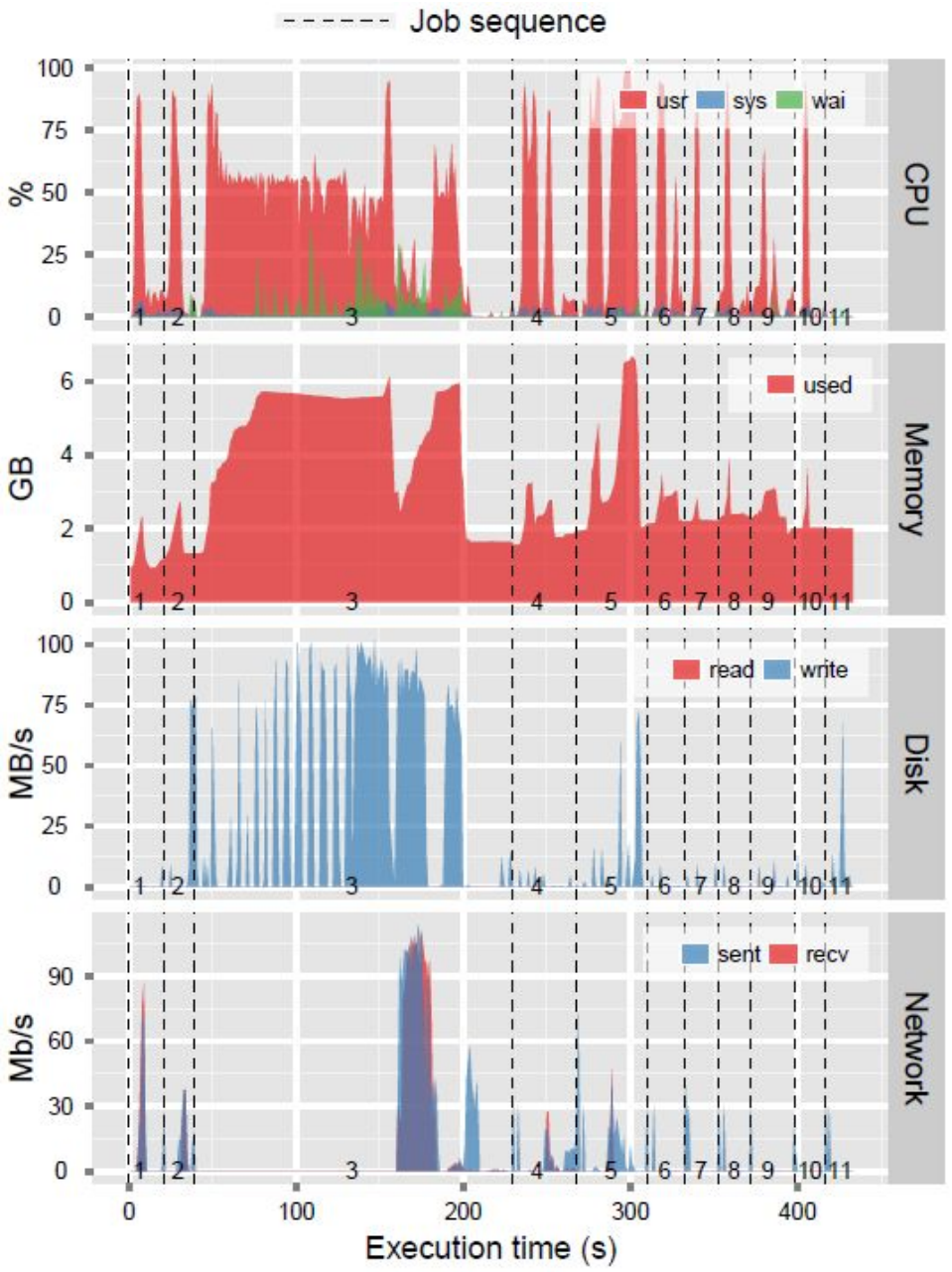
- Well-defined resource patterns
- Resource correlation
- A long period with no disk write operations
- Sub utilization of resources
- Other applications can run during this time slice



Disk I/O Characterization

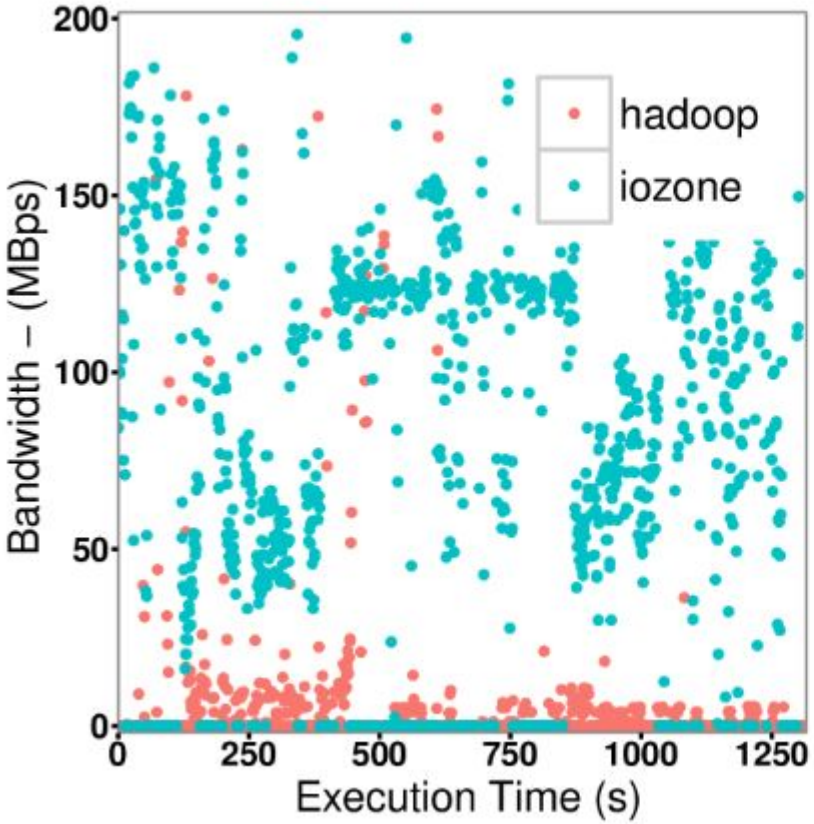
Bayes- a moderate disk application

- Resource usage variation/fluctuation
- Write operations are more intensive than read operations.



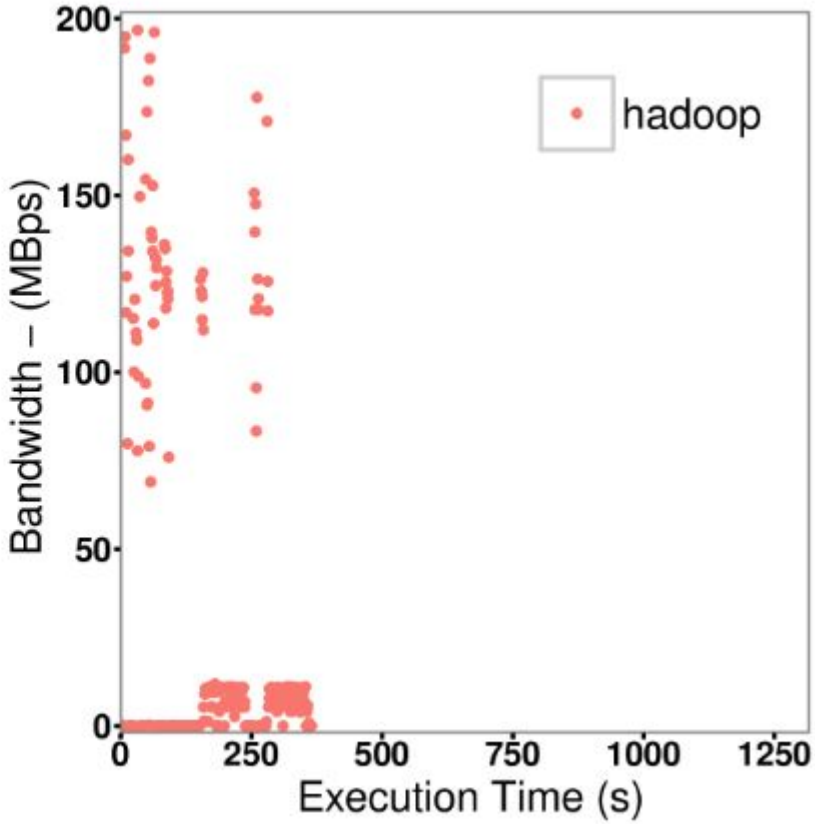
Disk I/O Contention and Resource Allocation

A



Contention Scenario

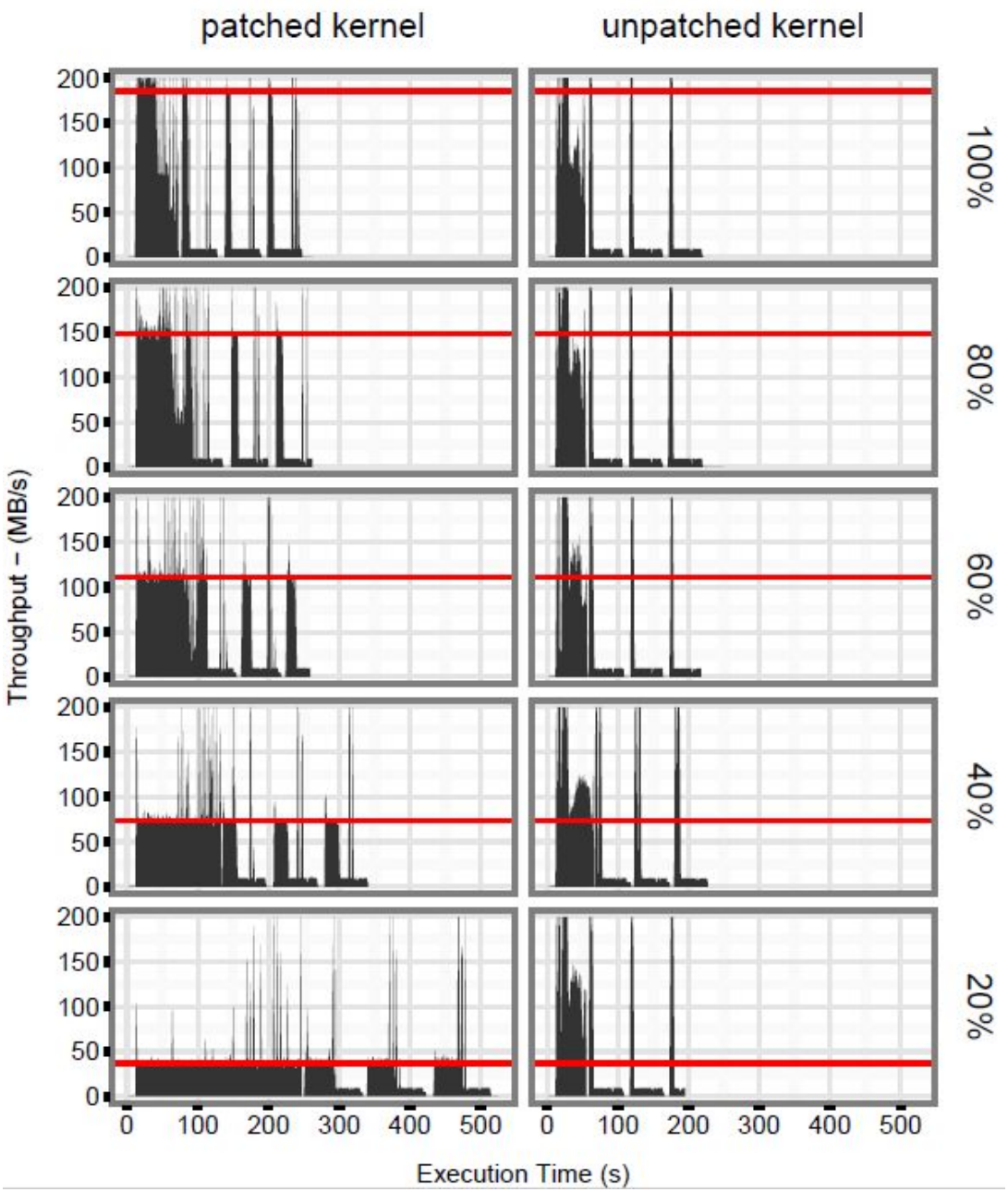
B



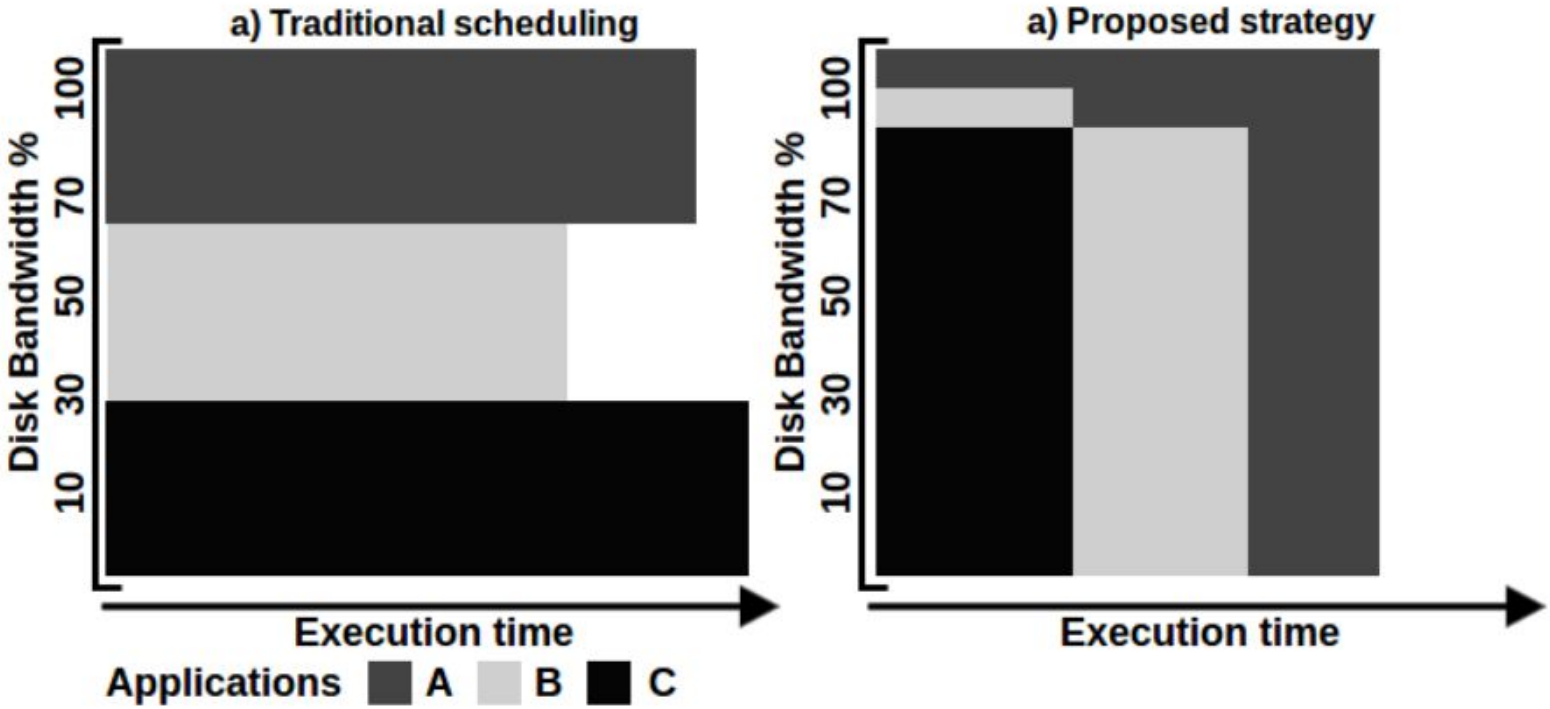
Non-contention scenario

Disk I/O Contention and Resource Allocation

- Cgroups (control groups)
  - Isolate a collection of process (CPU, memory,disk and network)
  - Block I/O controller (blkio)
- Measures Disk bandwidth (average)
- Kernel < 4 need a special patch
- Isolation faults



Towards a Dynamic Disk Resource Allocation



Goal: adjust the disk I/O utilization rates dynamically

Methodology: uses a variation of the Shortest Job First (SJF) algorithm. It is a greedy algorithm that selects the process with the shortest execution time.

Towards a Dynamic Disk Resource Allocation

Methodology

- We compared all OS I/O Schedulers (CFQ, noop, deadline)
- The metric observed was makespan (MS) - represents the application completion time considering the beginning and end of a sequence of jobs or tasks.
- Measurements have a confidence interval of at least 95%.
- Two evaluation scenarios
  - Homogeneous: same application and varied workload
  - Heterogeneous: varied applications and workloads
    - Real-world scenario

Towards a Dynamic Disk Resource Allocation

Testbed

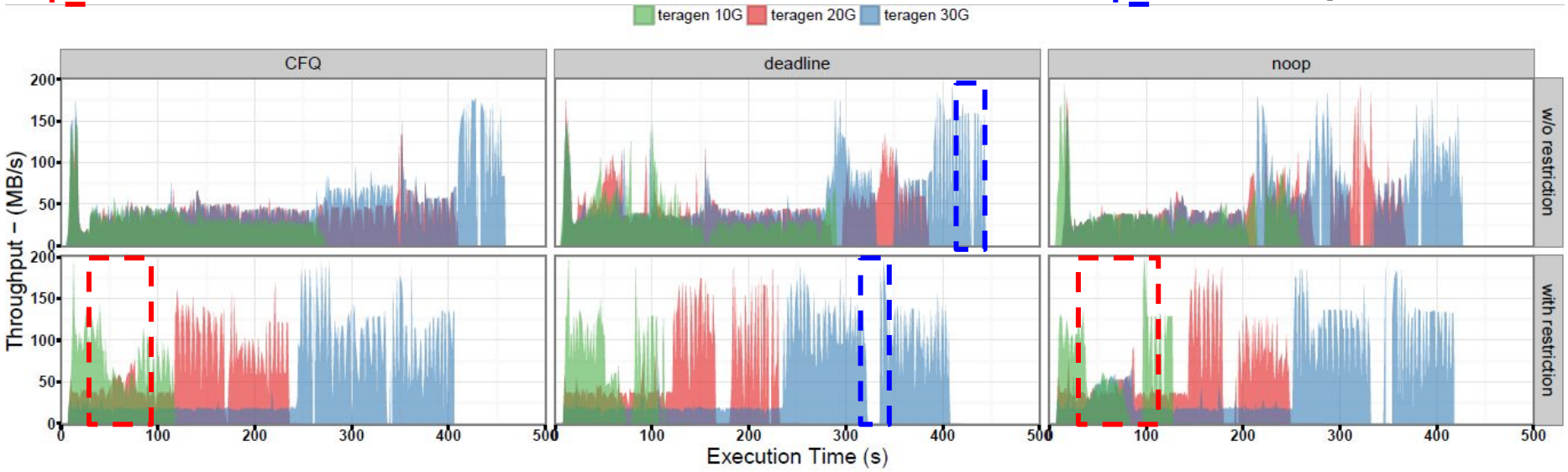
- Two identical Dell PowerEdge M610 servers, 24GB of RAM and one 300GB 10K RPM SAS disk;
- Ubuntu Server 14.04.1 LTS, patched with a custom Kernel (version 3.3) with container-based virtualization (LXC)
- A custom IOtop monitor
- Hadoop MapReduce (1.1.2), the most popular open-source implementation for Big Data analysis.
- HiBench, a realistic benchmark suite based on the MapReduce programming model with real-world applications.

Towards a Dynamic Disk Resource Allocation

Homogeneous evaluation

Isolation faults

Shuffle phase



MAKESPAN IMPROVEMENT BY SCHEDULER

State	CFQ	deadline	noop
w/o restriction (sec)	382	395	422
restriction (sec)	307	297	293
performance improvement (%)	19,6	24,8	30,56

Towards a Dynamic Disk Resource Allocation

Heterogeneous evaluation (real-world scenario)

Terasort (20GB), Sort (1.1GB) and K-means (3GB).

REAL WORLD SCENARIO: MAKESPAN IMPROVEMENT BY SCHEDULERS

State	CFQ	deadline	noop
w/o restriction (sec)	1614	1381	1525
restriction (sec)	1448	1131	1123
performance improvement (%)	10	18	26

DETAILED PERFORMANCE ANALYSIS PER APPLICATION

Schedulers	**CFQ**			**Noop**			**Deadline**		
Applications	**K-means**	**Sort**	**Terasort**	**K-means**	**Sort**	**Terasort**	**K-means**	**Sort**	**Terasort**
w/o restriction (sec)	877	1252	2713	1417	1158	2000	1287	987	1546
Restriction (sec)	737	859	2749	676	564	2131	699	522	1871
Gain (%)	15,9	31,3	-1,6	52,2	51,2	-6	45,6	47,1	-20

Conclusion and Future Work

The proposed strategy is feasible and effective to the presented scenarios.

- The strategy improves the application completion time up to 26%.
- Gains were obtained with all OS-system I/O Schedulers.

As future work, we would like to suggest:

- A fine-grained dynamic model to resize resources (CPU, memory, disk, network) on-demand.
- A Disk I/O scheduler for data-intensive processing.
- Towards a Dynamic Network Resource Allocation for data-intensive processing (batch and stream).

References

XAVIER M.G., **MATTEUSSI K.J.,** LORENZO F.E., DE ROSE C.A.F.; **Understanding performance interference in multi-tenant cloud databases and web applications**. In 4th Workshop on Scalable Cloud Data Management Co-located with the IEEE BigData Conference (BigData 2016), Washington D.C, p. 50-56.

[2] XAVIER, M.; DE OLIVEIRA, I.; ROSSI, F.; DOS PASSOS, R.; **MATEUSSI, K.;** DE ROSE, C.A.F. A **Performance Isolation Analysis of Disk-intensive Workloads on Container-based Clouds.** In: 23rd Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP 2015), Turku, Finland, p. 253-260

M. Malensek, S. L. Pallickara, and S. Pallickara, "**Alleviation of disk I/O contention in virtualized settings for data-intensive computing**," in Proceedings of the 2nd IEEE/ACM International Symposium on Big Data Computing (BDC), Dec 2015, pp. 1–10.

P. Mishra, M. Mishra, and A. K. Somani, "**Bulk I/O storage management for big data applications**," in Proceedings of the 24th IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), Sept 2016, pp. 412–417.

F. D. Rossi, G. D. C. Rodrigues, R. N. Calheiros, and M. D. S. Conterato, "**Dynamic network bandwidth resizing for big data applications**," in Proceedings of the 13th IEEE International Conference on e-Science (e-Science), Oct 2017, pp. 423–431.

References

B. Hou, F. Chen, Z. Ou, R. Wang, and M. Mesnier, "**Understanding I/O performance behaviors of cloud storage from a client's perspective**," Journal of ACM Transactions on Storage (TOS), vol. 13, no. 2, pp. 16:1–16:36, June 2017.

V. K. Vavilapalli, A. C. Murthy, C. Douglas, S. Agarwal, M. Konar, R. Evans, T. Graves, J. Lowe, H. Shah, S. Seth, B. Saha, C. Curino, O. O'Malley, S. Radia, B. Reed, and E. Baldeschwieler, "**Apache hadoop YARN: Yet another resource negotiator**," in Proceedings of the 4th Annual Symposium on Cloud Computing (SOCC), Oct. 2013, pp. 5:1–5:16.

S. Huang, J. Huang, J. Dai, T. Xie, and B. Huang, "**The HiBench Benchmark Suite: Characterization of The MapReduce-Based Data Analysis**," in Proceedings of the 26th IEEE International Conference on Data Engineering Workshops (ICDEW), March 2010, pp. 41–51.

[1]. M. V. Neves, "**Application-aware software-defined networking to accelerate mapreduce applications**," Ph.D. dissertation, Pontifıcia Universidade Catolica do Rio Grande do Sul, 2015.

C. Rista, D. Griebler, C. A. F. Maron, and L. G. Fernandes, "**Improving the network performance of a container-based cloud environment for hadoop systems**," in Proceedings of the International Conference on High Performance Computing & Simulation (HPCS), July 2017, pp 619–626.

Thanks!

Any questions?

You can find me at
kjmatteussi@inf.ufrgs.br