

# A ROUND-ROBIN SCHEDULING FOR COMPUTER CLUSTERS WITH COMPATIBILITY CONSTRAINTS

Thomas Bonald<sup>1</sup>, Céline Comte<sup>2,1</sup>

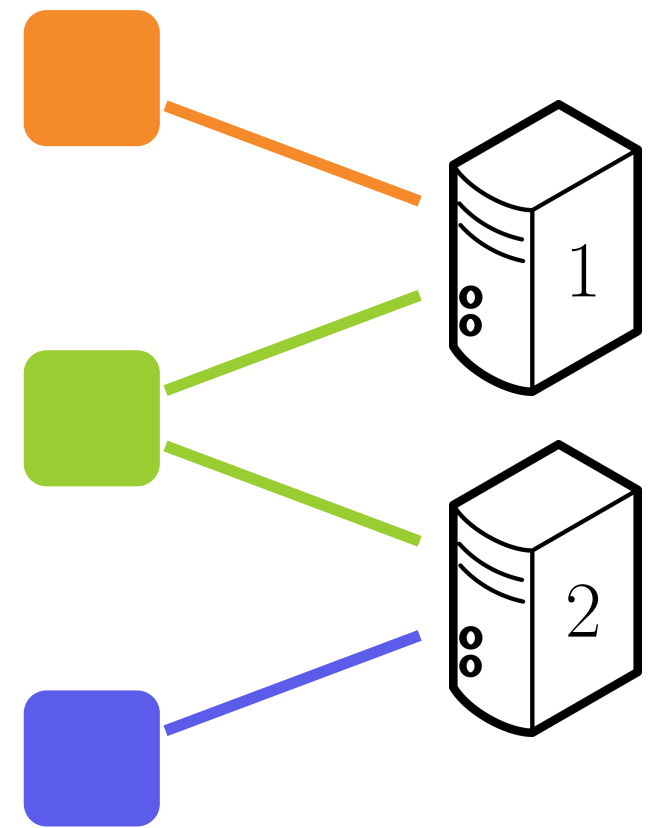
<sup>1</sup>Télécom ParisTech, Université Paris-Saclay  
<sup>2</sup>Nokia Bell Labs, Nozay



## Context

We consider a computer cluster where the compatibilities between jobs and computers are described by a bipartite graph. This graph may be chosen by the service provider or imposed by technical constraints like data availability. Assuming jobs can be processed in parallel on several machines, we design a scheduling algorithm achieving **balanced fair sharing** [1] of the computing resources, similar to round-robin scheduling in the presence of a single computer. The performance of this algorithm is analyzed by introducing a new class of queueing system.

## System



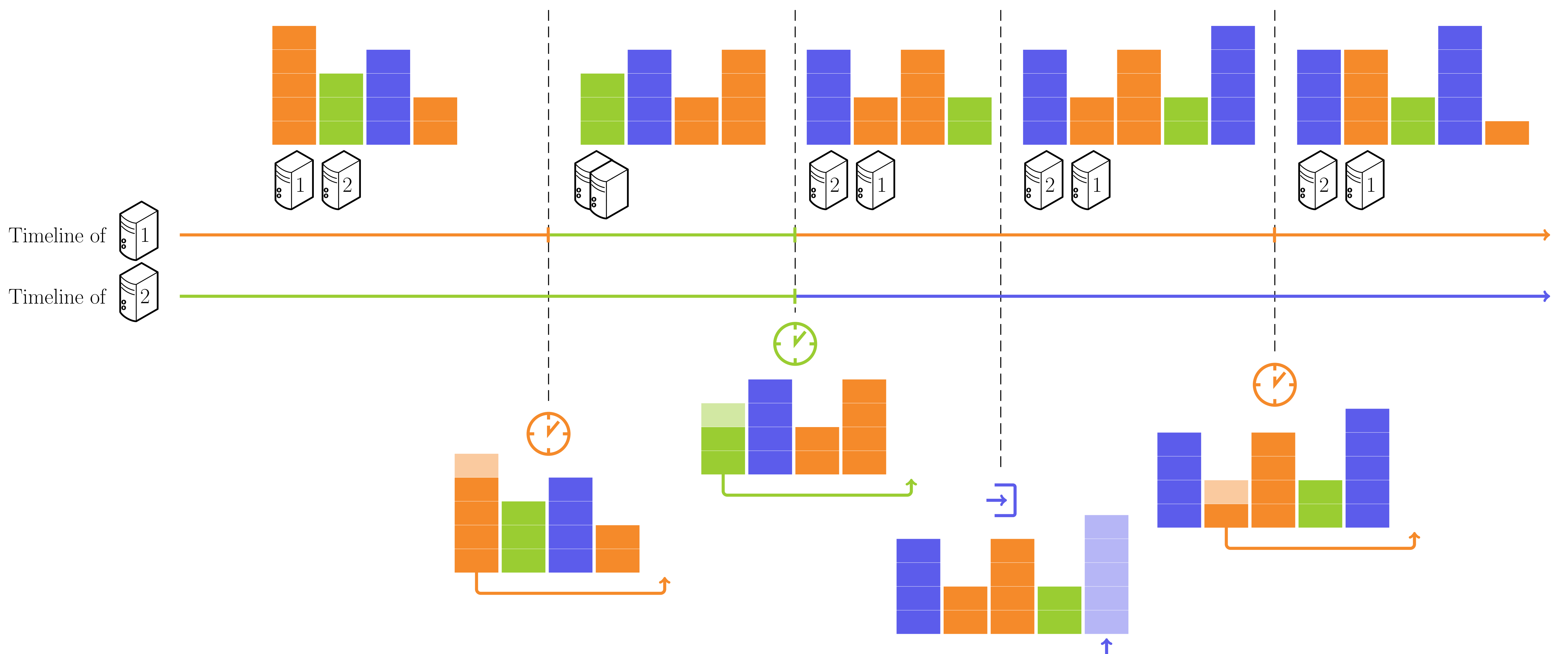
- **Compatibility constraints.** Bipartite graph of compatibility between job classes and computers.
- **Sequential processing** at each computer.
- **Parallel processing.** If a job is in service on several computers, its service rate is the sum of the capacities of these computers. This assumption is valid when the overload induced by the coordination is negligible.

## Scheduler

**Extension of round-robin scheduling algorithm to multiple computers.**

- Each server has a timer which expires after an exponentially distributed random time. The service of a job is interrupted whenever the timer of *one* of its computer expires.
- A job which is interrupted is moved to the end of the queue. The computers that were processing this job are reallocated to other jobs in the queue.
- Any incoming job is added to the end of the queue.

## Execution

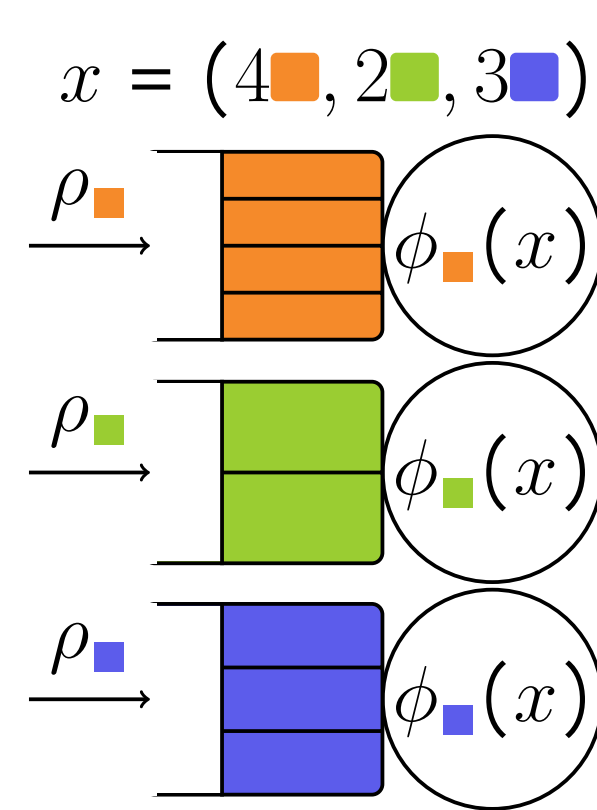


## Theory

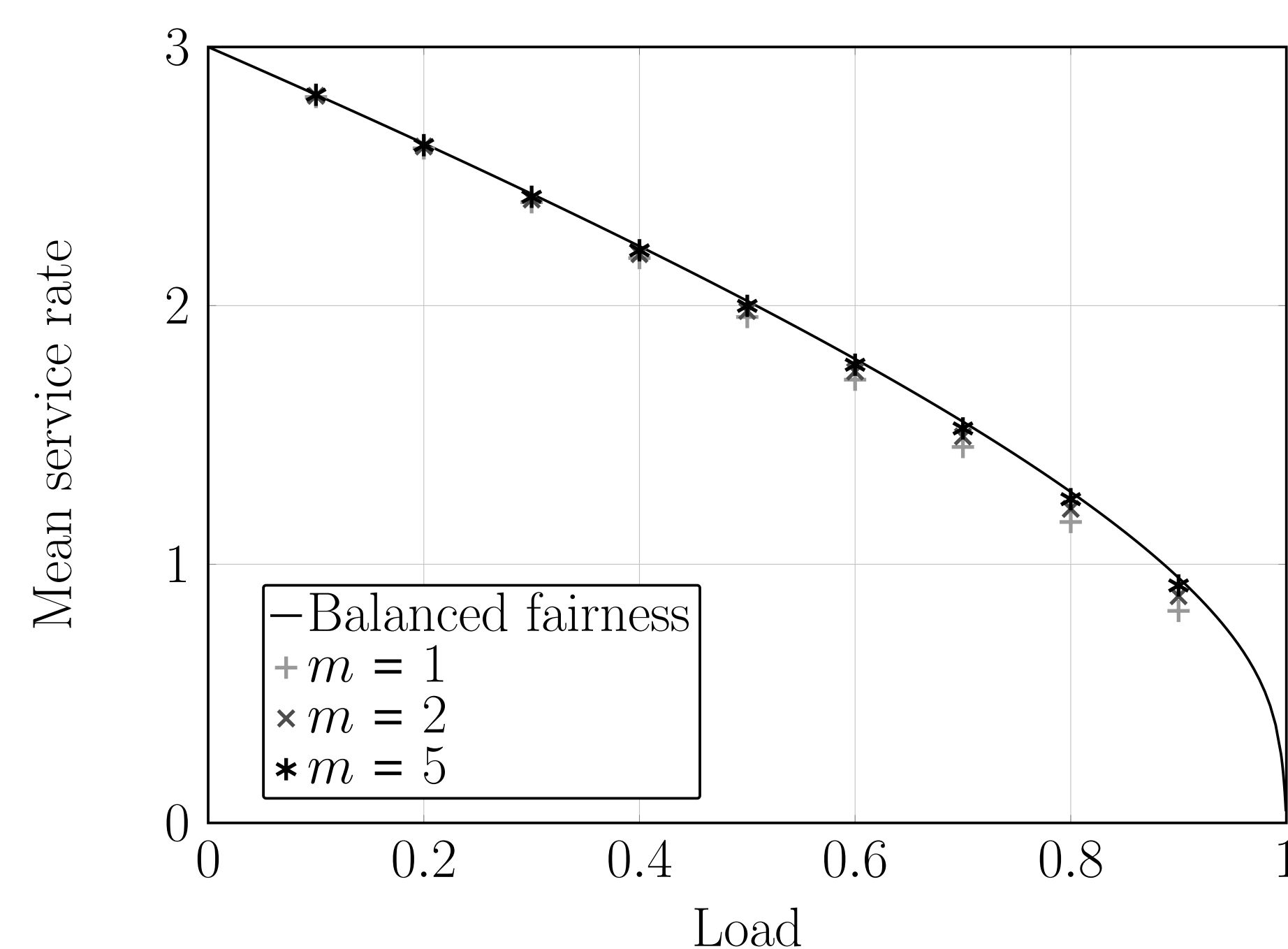
- **Small time scale: Order-independent queue** [2]
  - Independent Poisson arrivals per class, exponentially distributed job sizes.
  - Interruptions and resumptions reinterpreted in terms of random routing.
  - Product-form stationary distribution.

- **Aggregate behavior at a larger time scale: Network of processor-sharing queues**

- Combine all states containing the same number of jobs of each class.
- Stabilizes the maximum set of acceptable vectors of traffic intensity.
- Insensitivity. The steady-state distribution is independent of the detailed traffic characteristics beyond the traffic intensity.
- Robust and explicit recursion formulas for the performance metrics.



## System simulations



- Random assignment of 3 servers among 100.
- Hyperexponential job size distribution with mean 1, standard deviation approximately equal to 2.5.
- **Average number  $m$  of interruptions per job.**
- The performance is approximately insensitive for reasonable values of  $m$ .

## Conclusion

We have introduced a new scheduling algorithm that extends round-robin scheduling to computer clusters with compatibility constraints. We have assessed the performance of this algorithm with a new queueing model and showed that the corresponding average resource allocation is insensitive to the job size distribution.

For the future works, we would like to gain more insight on the impact of the mean number of interruptions per job on the sensitivity of the resulting resource allocation.

## References

- [1] T. Bonald and A. Proutière. "Insensitive Bandwidth Sharing in Data Networks". In: *Queueing Syst.* 44.1 (2003), pp. 69–100.
- [2] A. E. Krzesinski. "Order Independent Queues". In: *Queueing Networks: A Fundamental Approach*. Ed. by Richard J. Boucherie and Nico M. van Dijk. Boston, MA: Springer US, 2011, pp. 85–120.
- [3] T. Bonald and C. Comte. "Balanced Fair Resource Sharing in Computer Clusters". In: *CoRR* abs/1604.06763 (2017). URL: <https://arxiv.org/abs/1604.06763v2>.