Controlling Performance Interference in Multi-Tenant Containerized Environments

M.Reza HoseinyFarahabady Albert Y. Zomaya

Center for Distributed and High Performance Computing University of Sydney, School of Computer Science

22nd Intl. Symp. on Network Computing & Applications, NCA 2024

25 Oct 2024

Motivation

- Linux containers: multiple isolated Linux systems on a single host
- Kernel namespaces for resource isolation and control groups mechanism cgroups
- Containerized applications enhance resource efficiency
- Consolidating multiple applications on a single machine lead to resource contention
- Competing for shared cache or conflicting disk I/O patterns
- Co-residency problem

Motivation...

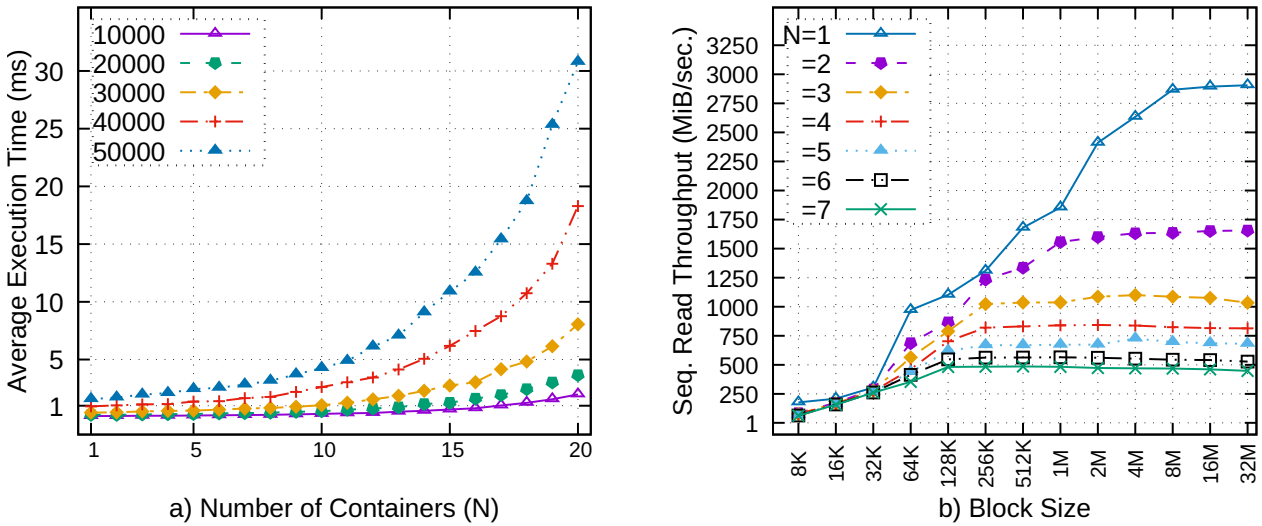

Figure: Performance interference measurements for (left) CPU-bound and (right) I/O-bound workloads

Main Contribution

- Designing a layer to predict and mitigate the interference effects within a containerized platform
- Monitoring component: collects performance metrics for all containerized applications at runtime using Linux profiling utilities
- Interference Analysis and Performance Prediction component
- Leverages the classic cycles per instruction (CPI) model and I/O read/write bus transaction statistics to identify interference effects
- How to define a Slowdown factor?
- Controller component: based on model predictive controller
- Calculating slowdown factor by containers and triggering migration for each container with a slowdown factor greater than its threshold value
- We use a Cost Benefit Analysis

Configuration and Performance Analysis

- Running in a four-node cluster
- A range of functional workloads (Data Management Service, Web Server, Data Analytics) with three SLA classes

Result Summary

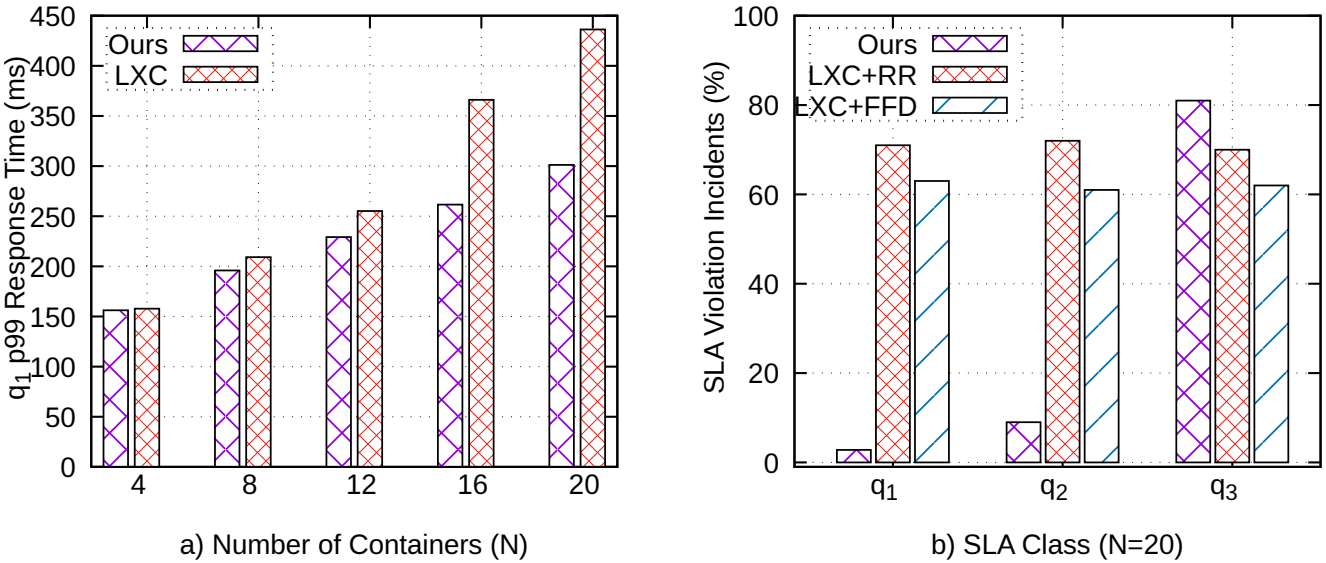

Figure: Comparison of (left) p-99 response time of web service applications in class q_1 ($4 \leq N \leq 20$) and (right) SLA violation incidents in different classes $q_{1 \leq i \leq 3}$ (N=20)

Result Summary ...

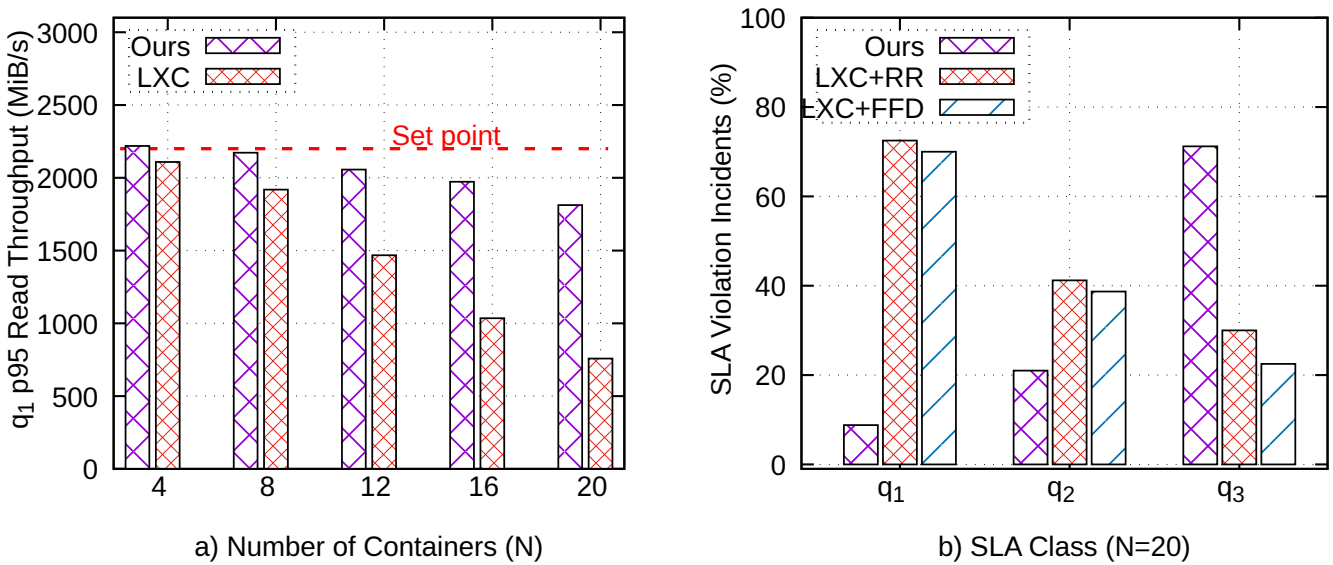

Figure: Comparing (a) the p-95 total I/O bandwidth share of containers in class q_1 (with $4 \leq N \leq 20$) and (b) the occurrence of SLA violations across classes $q_{1 \leq i \leq 3}$ (when $N = 20$) during the post-warm-up period for IO-intensive

Conclusion

- We designed an interference-aware controller for the LXC platform
- Adjusting computing resource usage while adhering to constraints imposed by high-priority applications

Acknowledgment

- **Professor Albert Y. Zomaya** would like to acknowledge the support of the Australian Research Council Discovery Project (DP200103494)
- **Dr. M. Reza HoseinyFarahabady** acknowledges the continued support of the Center for Distributed and High Performance Computing at the University of Sydney

Thank you!

Questions?