Perphon: a ML-based Agent for Workload Co-location via Performance Prediction and Resource Inference

Jianyong Zhu[1], Renyu Yang[23*], Chunming Hu[1], Tianyu Wo[1], Shiqing Xue[1], Jin Ouyang[4], Jie Xu[23*]
[1]Beihang University [2]Edgetic Ltd. UK [3]University of Leeds [4]Alibaba Group

ABSTRACT

Cluster administrators are facing great pressures to improve cluster utilization through workload co-location. Guaranteeing performance of long-running applications (LRAs), however, is far from settled as unpredictable interference across applications is catastrophic to QoS [2]. Current solutions such as [1] usually employ sandboxed and offline profiling for different workload combinations and leverage them to predict incoming interference. However, the time complexity restricts the applicability to complex co-locations. Hence, this issue entails a new framework to harness runtime performance and mitigate the time cost with machine intelligence: i) It is desirable to explore a quantitative relationship between allocated resource and consequent workload performance, not relying on analyzing interference derived from different workload combinations. The majority of works, however, depend on offline profiling and training which may lead to *model aging* problem. Moreover, multi-resource dimensions (e.g., LLC contention) that are not completely included by existing works but have impact on performance interference need to be considered [3]. ii) Workload co-location also necessitates fine-grained isolation and access control mechanism. Once performance degradation is detected, dynamic resource adjustment will be enforced and application will be assigned an access to specific slices of each resources. Inferring a *"just enough"* amount of resource adjustment ensures the application performance can be secured whilst improving cluster utilization.

We present Perphon, a runtime agent on a per node basis, that decouples ML-based performance prediction and resource inference from centralized scheduler. Figure 1 outlines the proposed architecture. We initially exploit sensitivity of applications to multi-resources to establish performance prediction. To achieve this, *Metric Monitor* aggregates application fingerprint and system-level performance metrics including CPU, memory, Last Level Cache (LLC), memory bandwidth (MBW) and number of running threads, etc. They are enabled by Intel-RDT and precisely obtained from resource group manager. Perphon employs an Online Gradient Boost Regression Tree (OGBRT) approach to resolve model aging problem. *Res-Perf Model* warms up via offline learning that merely relies on a small volume of profiling in the early stage, but evolves with arrival of workloads. Consequently, parameters will be automatically updated and synchronized among agents.

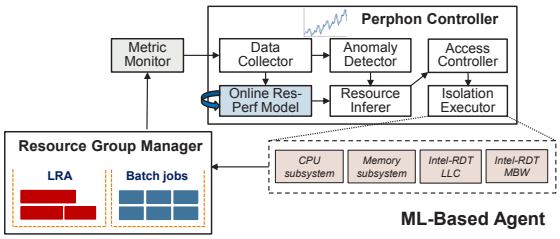

Figure 1: Architecture overview of Perphon

Anomaly Detector can timely pinpoint a performance degradation via LSTM time-series analysis and determine when and which application need to be re-allocated resources. Once abnormal performance counter or load is detected, *Resource Inferer* conducts a gradient ascend based inference to work out a proper slice of resources, towards dynamically recovering targeted performance. Upon receiving an updated re-allocation, *Access Controller* re-assigns a specific portion of the node resources to the affected application. Eventually, *Isolation Executor* enforces resource manipulation and ensures performance isolation across applications. Specifically, we use cgroup *cpuset* and *memory* subsystem to control usage of CPU and memory while leveraging Intel-RDT technology to underpin the manipulation of LLC and MBW. For fine-granularity management, we create different groups for LRA and batch jobs when the agent starts. Our prototype integration with Node Manager of Apache YARN shows that throughput of Kafka data-streaming application in Perphon is 2.0x and 1.82x times that of isolation execution schemes in native YARN and pure cgroup cpu subsystem.

CCS CONCEPTS

• **Computer systems organization** → **Cloud computing**; • **Software and its engineering** → *Software performance.*

KEYWORDS

Performance Modelling, Resource Inference

ACM Reference Format:
Jianyong Zhu[1], Renyu Yang[23*], Chunming Hu[1], Tianyu Wo[1], Shiqing Xue[1], Jin Ouyang[4], Jie Xu[23]. 2019. Perphon: a ML-based Agent for Workload Co-location via Performance Prediction and Resource Inference. In *ACM Symposium on Cloud Computing (SoCC '19), November 20–23, 2019, Santa Cruz, CA, USA.* ACM, New York, NY, USA, 1 page. https://doi.org/10.1145/3357223.3365440

REFERENCES

[1] Christina Delimitrou and Christos Kozyrakis. 2013. Paragon: QoS-aware scheduling for heterogeneous datacenters. In *ACM ASPLOS 2013.*
[2] Jacob Leverich and Christos Kozyrakis. 2014. Reconciling high server utilization and sub-millisecond quality-of-service. In *ACM EuroSys 2014.*
[3] Rathijit Sen and Karthik Ramachandra. 2018. Characterizing resource sensitivity of database workloads. In *IEEE HPCA 2018.*

*J.Zhu and R.Yang contributed equally to this work. C.Hu is corresponding author.
This work is supported by China National Key R&D Program (2016YFB1000503)

SoCC '19, November 20–23, 2019, Santa Cruz, CA, USA

ACM ISBN 978-1-4503-6973-2/19/11.
https://doi.org/10.1145/3357223.3365440