Title	Performance Engineering on HPC Clusters
Author(s)	Wang, Chiye
Citation	
Issue Date	2023-09
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/18741
Rights	
Description	Supervisor:田中 清史, 先端科学技術研究科, 修士(情報科学)

Performance Engineering on HPC clusters

2110404 Chiye Wang

The complexity of scientific computing tasks necessitates the design of intricate software and hardware stacks in HPC systems to meet the scalability requirements of computational kernels. However, the increased complexity of HPC systems poses challenges for designing efficient performance analysis methods. Currently, the HPC community employs diverse and complex performance analysis tools to model the performance of HPC systems implemented on different architectures. However, this fragmented analysis approach requires performance experiments that are relatively constrained in design for target applications and systems, leading to experimental methods and results lacking portability and generality.

Our contributions mainly focus on two aspects. Firstly, we propose a multi-level observability approach for HPC systems using eBPF (Extended Berkeley Packet Filter). We describe several widely-used industrial-grade HPC performance tool implementations and performance methodologies, and identify limitations in existing performance approaches. By leveraging native features of the Linux kernel, we enhance the portability and cross-platform generality of code instrumentation and dynamic tracing capabilities. Validation shows that eBPF incurs lower runtime overhead and offers high controllability compared to traditional methods. Moreover, eBPF is language-agnostic, providing robust support for programs implemented in languages which have simpler runtimes.

Secondly, we select specific performance modeling methods from conventional performance engineering approaches, including Profiling for gaining insights into the target application's execution flow and hot functions, Kernel Extraction for understanding the nature of computational kernels and calculating theoretical single-core performance, Dependency Chain Identification for assessing multi-threading performance degradation, Kernel Benchmark for evaluating the target workload's actual performance on the system, and I/O and Memory Benchmark for comprehending the target system's I/O performance. Based on these data, we can use theoretical models of parallel computing to fit data that closely approximates real performance. Furthermore, leveraging the eBPF performance sampling data from the first part, we can implement unsupervised system performance non-regression testing. We hope that this thesis can provide a potential, portable, cross-platform, and WORA performance methodology for future HPC systems.