Compas'2023 : Parallélisme/ Architecture/ Système
LISTIC/USMB - Annecy France, 4-7 juillet 2023

CADiA : A CPU Cache Allocation System for Distributed Applications

Armel Jeatsa, Boris Teabe, Daniel Hagimont

Université de Toulouse,
Laboratoire IRIT

Résumé
Today, we witness an increasing variety of applications in IT. There is a significant collocation of various workloads in data centers. Among these workloads, we can find distributed applications that represent an important portion. Examples of popular distributed workloads include big data applications such as Hadoop and Spark, as well as HPC workloads that utilize libraries such as MPI.
A CPU cache is a fast multi-level memory of limited size that stores the data used by the CPU. Since cache space consists of relatively fast, expensive storage, it is inherently a scarce resource that is shared among multiple CPU. More specifically, the Last Level Cache (LLC) is the cache level that is shared by CPUs. As with most shared resources, the LLC is subject to contention issues. Therefore, inadequate usage can significantly impact application performances. Current processors offer features that allow software partitioning of the LLC. For instance, Intel processors introduce CAT ("Cache Allocation Technology"), which enables the partitioning of the cache and assignment of these partitions to applications. However, the observation stemming from the review of existing works on LLC allocation is that, in most cases, cache usage analysis is conducted locally on a server (node), which may be inadequate and insufficient for distributed applications. In a distributed application the overall execution time is determined by the performance of each individual node. Consequently, the computing speed of each node plays a crucial role in the application's overall execution time. Therefore, there is a need to develop new approaches that consider the distributed nature of these applications in order to optimize cache allocation across multiple nodes.
Our work addresses this issue by providing a two-fold contribution. Firstly, we conducted a thorough analysis of the impact of local cache contention on the performance of a distributed application. Secondly, based on our conclusions, we introduced CaDIA which is a system that proposes two CPU cache allocation strategies, named Bottom-Top and Top-Bottom, specifically designed for distributed applications. We implemented these two allocation strategies on MPI library and evaluated their performance with high-performance computing (HPC) applications. Our results demonstrate that the Top-Bottom strategy enhances the performance of distributed applications by up to 13% in scenarios with varying levels of node contention. Meanwhile, the Bottom-Top approach decreases the miss ratio of co-located applications by up to 20%.
