Quantifiying the Overhead of Coscheduling on Multi-core Architectures

Abstract

Jan H. Schönherr

Communication and Operating Systems Group
Technische Universität Berlin, Germany
schnhrr@tu-berlin.de

Coscheduling provides various advantages in scheduling on parallel architectures, for example parallel applications can efficiently use fine-grained communication or the operating system can try to reduce resource contention that unfortunately arises on current multi-core systems, where various resources are shared. However, normally coscheduling is associated with considerable overheads making it only suitable in very specific scenarios. For example, when compared to partitioning, applications get larger partitions which is less efficient according to Amdahl's Law, and there are a lot more context switches. While the context switch itself is rather fast, the following cache misses and a possibly period of non-simultaneous execution have to be accounted for.

In this work, we analyze the impact of multi-core architectures on coscheduling with a specific focus on overheads. Based on previous work [1, 2], we quantify different overheads of a coscheduler specifically designed for general purpose workloads. With this, we are able to assess the potential of coscheduling in different domains and derive guidelines for necessary trade-offs, such as time-slice length and partition sizes.

References

1. Jan H. Schönherr, Bianca Lutz, and Jan Richling. Non-intrusive coscheduling for general purpose operating systems. In *Proceedings of the International Conference on Multicore Software Engineering, Performance, and Tools (MSEPT '12)*, volume 7303 of *Lecture Notes in Computer Science*, pages 66–77, Berlin/Heidelberg, Germany, May 2012. Springer.
2. Jan H. Schönherr, Ben Juurlink, and Jan Richling. Topology-aware equipartitioning with coscheduling on multicore systems. In *Proceedings of the 6th International Workshop on Multi-/Many-core Computing Systems (MuCoCoS-2013)*, Piscataway, NJ, USA, September 2013. IEEE. (to appear).