**Time:** 4-5 PM, October 22, 2024

**Location:** KD 102

**Title:** Leveraging Cache Coherence to Detect and Repair False Sharing On-the-fly

**Abstract:**

This talk will discuss our recent work addressing the well-known performance problem of false sharing in multithreaded programs. Despite being researched for the last 30 years, the performance problem stemming from false sharing has affected various production applications like the MySQL database, Netflix, Disruptor library, Boost Spinlock, and the Linux Kernel. Therefore, automatic detection and repair of false sharing is an important research direction. Prior approaches aim to both detect and repair false sharing instances automatically. Still, most of them suffer from one or more of the following drawbacks: (i) high performance overhead due to expensive tracking of shadow memory, (ii) reliance on imprecise hardware events, and (iii) limited applicability and portability.

Since false sharing is an artifact of invalidation-based SWMR coherence protocols, we present extensions to the MESI cache coherence protocol to efficiently identify and mitigate false sharing instances. The FSDetect protocol tracks the frequency of coherence misses per cache block to identify harmful instances of falsely shared lines while incurring negligible performance overhead. The FSLite protocol extends FSDetect to transparently privatize the falsely shared lines on accesses after detection, thereby eliminating the performance problem arising from false sharing. FSLite maintains coherence by performing precise byte-level updates of privatized blocks at the LLC on terminating privatization.

Our simulation results on various multithreaded workloads show that FSDetect can precisely identify all known harmful instances of false sharing. FSLite, on average, improves the performance of applications suffering from false sharing by 1.39X over the unmodified baseline, with a maximum speedup of 3.91x, at the cost of a minimal increase in the chip area. FSLite also reduces energy consumption by 27% and interconnect traffic by 84% by eliminating additional intervention and invalidation messages. Furthermore, our proposed protocols do not negatively impact applications without false sharing.