Alexander Powell

10.05.2016

CSCi 680

**Critique of “FAST: Fast Architecture Sensitive Tree Search on Modern CPUs and GPUs”**

**Summary**

In this paper, the authors lay out a new algorithm called FAST search, which exploits the high compute in modern processors for index tree traversal. FAST is a binary tree that is logically organized to optimize for architecture features like page size, cache line size and SIMD width of the underlying software. The motivation for the project was that index trees were previously not optimized for architecture. That is, usually only one node is accessed per tree level, which is ineffective.

There have been previous works that have either optimized for page size, cache size, or SIMD extensions, but no projects were using these techniques together. The main sources of tree source optimization come from hierarchical blocking. This new algorithm was tested on both CPUs and GPUs. With CPUs, the FAST algorithm is more noticeable for larger trees than smaller ones, because they’re more latency bound. The final results indicate that the FAST architecture can achieve a performance boost by as much as 5x on a CPU and 1.7x on a GPU.

**Overall Pros/Cons**

This paper does provide a novel approach at providing a new search algorithm for binary trees. The authors give proper credit to existing works and explain how they either fall short or could be improved. For example, as mentioned above, there were a number of existing works that that performed some component of hierarchical blocking for optimization. However, the significance of this work is that it incorporates many of these together to produce a better result. Also, the area of research is clearly significant because databases are commonly used in both academic but especially commercial enterprises. Also, as technology becomes more and more prevalent in our lives it will become increasingly important to be able to quickly index and search through large data sets.

**Detailed concerns and questions**

The main concern/criticism I had about the paper was that it could have done a better job at explaining some of the more advanced topics and algorithms. I was a little unclear how they’re algorithm for binary tree traversal could be applied to a database, and if there are other ways of representing this information. Also, they were a little vague about why the FAST algorithm performed better on CPUs than GPUs, but stated that new GPUs would be able to outperform CPUs by several factors. This was never clearly explained.