Summary
Helper threads via Virtual multithreading on an experimental Itanium 2 processor-based platform

Venkatesan Packirisamy
February 7, 2005

1 To be completed before class

What are the problems solved by this paper? (50 words)
Widening processor-memory performance gap lead to many techniques which try to improve cache performance. One such successful techniques is to use helper threads to prefetch hard-to-predict delinquent data accesses. This technique was successfully used in systems which hardware support for multithreading, like the hyperthreading in P4. But this technique cannot be applied directly to Itanium processor, since it cannot support multiple threads without invoking the OS. This paper designs a hardware mechanism called virtual multithreading which can run the helper threads to prefetch data.

What are the approaches attempted by this paper? (50 words)
The paper uses a technique calle virtual multithreading. Since we have only one thread of execution, we cannot overlap the execution of the helper thread with the main thread. Only way to run the helper thread is when the main thread suffers a cache miss in the L3 cache. So we need some event triggered multithreading support. Existing multithreading techniques involve the OS, which is a huge overhead. So a hardware mechanism is built on the Processor abstraction layer (PAL) of the Itanium processor. This acts along with the performance monitoring unit. New instructions are inserted (yield and yield conditional) to start the helper thread. When the cache miss occurs and if the yield instruction occurs, the PAL switches control to the helper thread. Techniques are also used to control the helper thread from going too far. Advanced compiler techniques are used to generate the helper threads. Thus a multithreading execution is emulated in a single thread Itanium processor using VMT.

What are the main conclusions of this paper? (50 words)
The VMT technique lead to wall-clock speedups of 5.8% to 38.5% for the workstation benchmarks. The technique also showed good performance for database workloads. So it is showed that through VMT helper thread execution is possible even in a single threaded machine. To achieve high performance with helper threads, the helper threads have to adaptive to the current situation. Since the VMT is tied to program’s dynamic behavior, helper threads through VMT shows good performance. The same techniques can be applied to increase throughput of some programs.

2 To be completed after class

Did this paper address an important issue? Explain. (100 words)

Are the proposed approaches valid? Describe its strength and weakness. (100 words)
Do the results support the conclusions? Explain. (100 words)

Describe the potential future works? (100 words)