Summary
A Scalable Approach to Thread-Level Speculation

Kiran Yellayosula
January 31, 2005

1  To be completed before class

What are the problems solved by this paper? (50 words)
The paper addresses the issue of developing parallel software to exploit the available hardware resource to the fullest potential. The current sequential programs have arbitrary memory patterns and the authors propose a scalable method which scales over all architectures by extending the cache coherence protocol. The paper also addresses the usability of this approach on multi-chip multiprocessors where the communication latencies are significant.

What are the approaches attempted by this paper? (50 words)
The cache lines are extended with a few bits which indicate whether the line was speculatively loaded or modified. The authors extend the multiprocessor Cache Coherence Protocol with two states namely, speculative exclusive and speculative share and design the protocol for any sequence of memory operations. Each communication message is added a thread-id to distinguish if the operation has any effect on the current thread.

An Ownership Required Buffer (ORB) is supplied to store all the memory addresses modified by a given processor, so that they can be invalidated or the speculation flag cleared in case of a failure of success respectively.

The authors propose other schemes to optimize the performance such as, synchronizing communications with wait and signal, extending the number of cache states to allow dirty and speculative loads, suspending future epochs causing cache violations.

What are the main conclusions of this paper? (50 words)
The authors demonstrated results showing program speedups ranging between 8% to 46% on a 4 processor single chip processor. Only some tightly coupled programs are sensitive to latency while many programs are insensitive to latencies in tens of cycles. Multiple speculative writers can be avoided by static strategies like loop unrolling. Allowing speculative invalidations to invalidate non-speculative cache lines provides a 10% increase in speedup. The splitting of processors into nodes provides a tradeoff in performance due to increase in cache storage and inter-processor communication latency.

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)