

CMU 18-447 INTRODUCTION TO COMPUTER ARCHITECTURE, SPRING 2012
HANDOUT 16/ HW 7: PREFETCHING, PRE-EXECUTION, CACHE COHERENCE

Prof. Onur Mutlu, Instructor
Chris Fallin, Lavanya Subramanian, Abeer Agrawal, TAs

Given: Thursday, May 3, 2012

1 Prefetching

An architect is designing the prefetch engine for his machine. He first runs two applications A and B on the machine, with a stride prefetcher.

Application A:

```
uint8_t a[1000];
sum = 0;
for (i = 0; i < 1000; i += 4)
{
    sum += a[i];
}
```

Application B:

```
uint8_t a[1000];
sum = 0;
for (i = 1; i < 1000; i *= 4)
{
    sum += a[i];
}
```

i and sum are in registers, while the array a is in memory. A cache block is 4 bytes in size.

- (a) What is the prefetch accuracy and coverage for applications A and B using a stride prefetcher. This stride prefetcher detects the stride between two consecutive memory accesses and prefetches the cache block at this stride distance from the currently accessed block.
- (b) Suggest a prefetcher that would provide better accuracy and coverage for
 - i) application A?
 - ii) application B?
- (c) Would you suggest using runahead execution for
 - i) application A. Why or why not?
 - ii) application B. Why or why not?

2 Amdahl's Law

Amdahl's abstraction of a parallel program represents a parallel program as two portions, one in which the program is completely serial (the serial portion) and the other in which the program is running completely in parallel (the parallel portion). In practice, the parallel portion doesn't achieve perfect speedup. Why is this so? Give three reasons, no more than 10 words each.

3 Cache Coherence

- (a) What is the advantage of the MESI protocol over the MSI cache coherence protocol?
- (b) A designer has built a directory-based cache coherent system using the MESI invalidation protocol. Under certain workloads, however, the system performs very poorly, and after a simulation run and closer examination, the designer finds that there is a constant stream of invalidation requests between four of the nodes for one particular cache block. Why does this happen?
- (c) Where and how is the problem best solved?

4 Markov Prefetchers vs. Runahead Execution

- (a) Provide two advantages of runahead execution over markov prefetchers.
- (b) Provide two advantages of markov prefetchers over runahead execution.
- (c) Describe one memory access pattern in which runahead execution performs better than markov prefetchers. Show pseudo-code.
- (d) Describe one memory access pattern in which runahead execution performs worse than markov prefetchers. Show pseudo-code.

5 Pre-Execution

A machine's designer is trying to use thread-based pre-execution to speed up an application A's performance. The machine has two cores. Each core has its own private L1 and L2 cache. The cores share an L3 cache and the memory. The designer tries to improve application A's performance by i) running a pruned speculative thread on a separate core and ii) running a pruned speculative thread on a separate thread context on the same core.

- (a) Give one reason why executing the speculative thread on a separate core could provide better performance than executing the speculative thread on a separate thread context on the same core.
- (b) Give one reason why executing the speculative thread on a separate thread context on the same core could provide better performance than executing the speculative thread on a separate core.
- (c) The designer finds that executing the speculative thread on a separate core provides better performance for application A. Now, the core executing the speculative thread is hit by gamma rays and a bit in its register file flips. How does this affect the correctness of application A's execution?
- (d) The designer finds a way to parallelize application A and splits its work into two threads. He expects that running the two threads on the two cores would provide better performance than running a speculative thread on one of the cores and the single-threaded version of the program on the other core, as in (c). However, he is surprised to see that the opposite is true. Why do you think this is the case?
- (e) When the two threads of application A are executing on the two cores, the second core is hit by gamma rays and a bit in its register file flips. How does this affect the correctness of application A's execution?

6 Network Contention

- (a) Assume we have a 2-dimensional mesh network. When multiple packets arriving from different input ports need to go to the same output port, we say the output port is being contended for. There are three general ways to handle this contention. First is to _____ some of the packets, second is to _____ some of the packets, third is to _____ some of the packets.
- (b) Assuming contention is extremely rare, which option would you choose? Explain why, clearly, in less than 20 words.

7 Runahead Execution

Suppose a branch is fetched in runahead mode of a runahead execution processor. Suppose we know magically whether or not the branch is mispredicted. The below questions deal with whether continuing runahead execution is always useless after the fetch of this branch.

- Is runahead execution always useless after a mispredicted branch that does not depend on an L2 cache miss? Why or why not?
- Is runahead execution always useless after a mispredicted branch that depends on an L2 cache miss? Why or why not?

8 Asymmetric CMPs

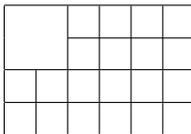
A multi-core processor consists of 16 simple cores on a single chip. A program consists of code that is 80% perfectly parallelizable and 20% serial.

- What is the maximum speed-up obtained by executing this program on the 16-core chip versus running it on a chip that only has a single simple core?
- Suppose instead we execute this program on a chip where 4 of the 16 simple cores have been replaced by one heavyweight core that processes instructions 2 times faster than a simple core. What is the speed-up of running on this chip versus running on a chip that has 16 simple cores? Assume that when the chip is processing the parallelizable code, the heavyweight core is idle.

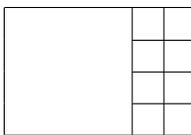
9 More Asymmetric CMPs and Amdahl's law

Let us assume that you are designing a multi-core processor to be fabricated on a fixed silicon die area budget of A . As the architect, you are to partition this total area of A into one large core and many small cores. The large core will have an area of S , while the small cores will each have an area of 1 (but there will be A/S of them). Assume that the single-thread performance of a core scales with the square root of its area. On this multiprocessor, we will execute a workload where P fraction of its work is infinitely parallelizable, and where $1 - P$ of its work is serial.

- Configuration X: $A = 24, S = 4$ (One large core, 20 small cores.)



- Configuration Y: $A = 24, S = 16$ (One large core, eight small cores.)



Assume that the serial portion of the workload executes only on the large core and the parallel portion of the workload executes on both the large and small cores.

- What is the speedup of the workload on configuration X? (Compared to the execution time on a single-core processor of area 1.)

- (b) What is the speedup of the workload on configuration Y? (Compared to the execution time on a single-core processor of area 1.)
- (c) For what range of P does the workload run faster on Y than X?
- (d) For workloads that have limited parallelism, which configuration would you recommend and why? (< 20 words.)