Design of a Storage Hierarchy in Multithreaded Architectures

Lucas Roh — Argonne National Lab
Math. & CS Division
roh@mcs.anl.gov

Walid Najjar — Colorado State University
Computer Science Dept.
najjar@cs.colostate.edu

Motivation

Multithreaxed execution: * masks latency (network and memory).
* higher CPU utilization.

Thread execution models: blocking, non-blocking, strict, non-strict.

High intra-thread locality: by definition or construction.

Inter-thread locality:

* to what extent?
* how can it be exploited?
* impact on execution time.

Pebbles Multithreaded Execution Model

Dataflow dynamic scheduling of threads:

* multiple threads instances distinguished by a context.
* execution upon data availability.

Statically generated non-blocking threads:

* maximize intra-thread locality $\Rightarrow$ minimize synchronization overhead.
* preserve code-block level parallelism (loop and functions).
* threads terminated by long latency operations.
* non-blocking $\Rightarrow$ no live registers accross threads.
* extensive optimizations, global and local.

Machine Model

Data values carried by tokens.

Synchronization Unit: stores tokens
for thread instance in its framelet.

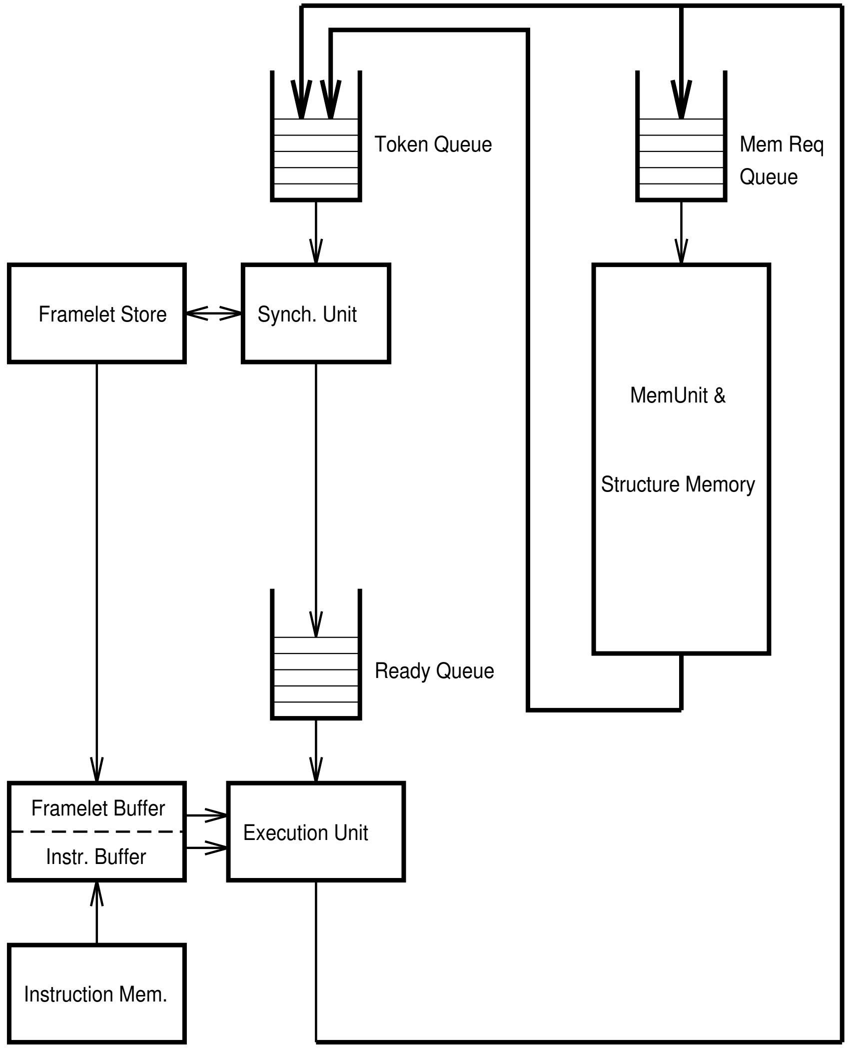
Thread characteristics:

size: 10 to 30 instructions;

avg. 22instructions.

avg. inputs/thread: 8.

avg. ilp within a thread: 4.



Benchmarks

SGA: local minima search genetic algorithm.

FFT: 1-D fft routine.

PSA: parallel scheduler.

SDD: elliptic PDE solver.

SIMPLE: Langrangian 2-D hydro code.

AMR: unsplit integrator from adaptive mesh refinement code.

WEATHER: one level barotropic weather prediction code.

Benchmark Charateristics

Benchmark	No. Threads	No. Inputs	Avg. Input Set Size (bytes)	% Input Set Size < 128 bytes
SGA	1,372,099	8,712,006	25.3	99.6 %
FFT	225,706	4,142,029	69.1	96.2
PSA	1,664,336	7,098,854	18.1	100
SDD	1,716,892	10,520,838	29.6	100
SIMPLE	1,328,271	9,269,106	34.4	98.1
AMR	205,774	3,207,898	85.1	84.2
WEATHER	828,983	8,045,883	35.9	99.8
Overall	7,342,061	50,996,614	30.5	99.0

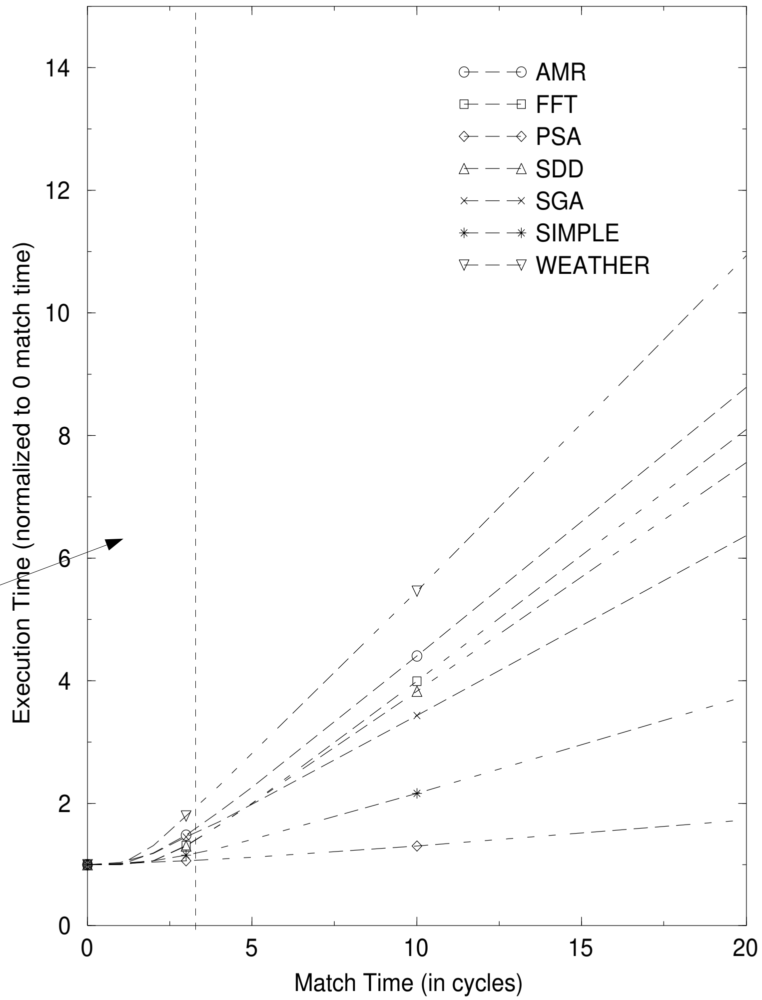
Impact of Synchronization

To what extent is synchronization cost masked by multithreaded execution?

In the seven benchmarks:
 * large variance in effects of synch. cost.

 * minimal effect when ≤ 3 cycles.

Target Zone



Pebbles Storage Hierarchy

Famelet: *stores all input data to a thread instance.*

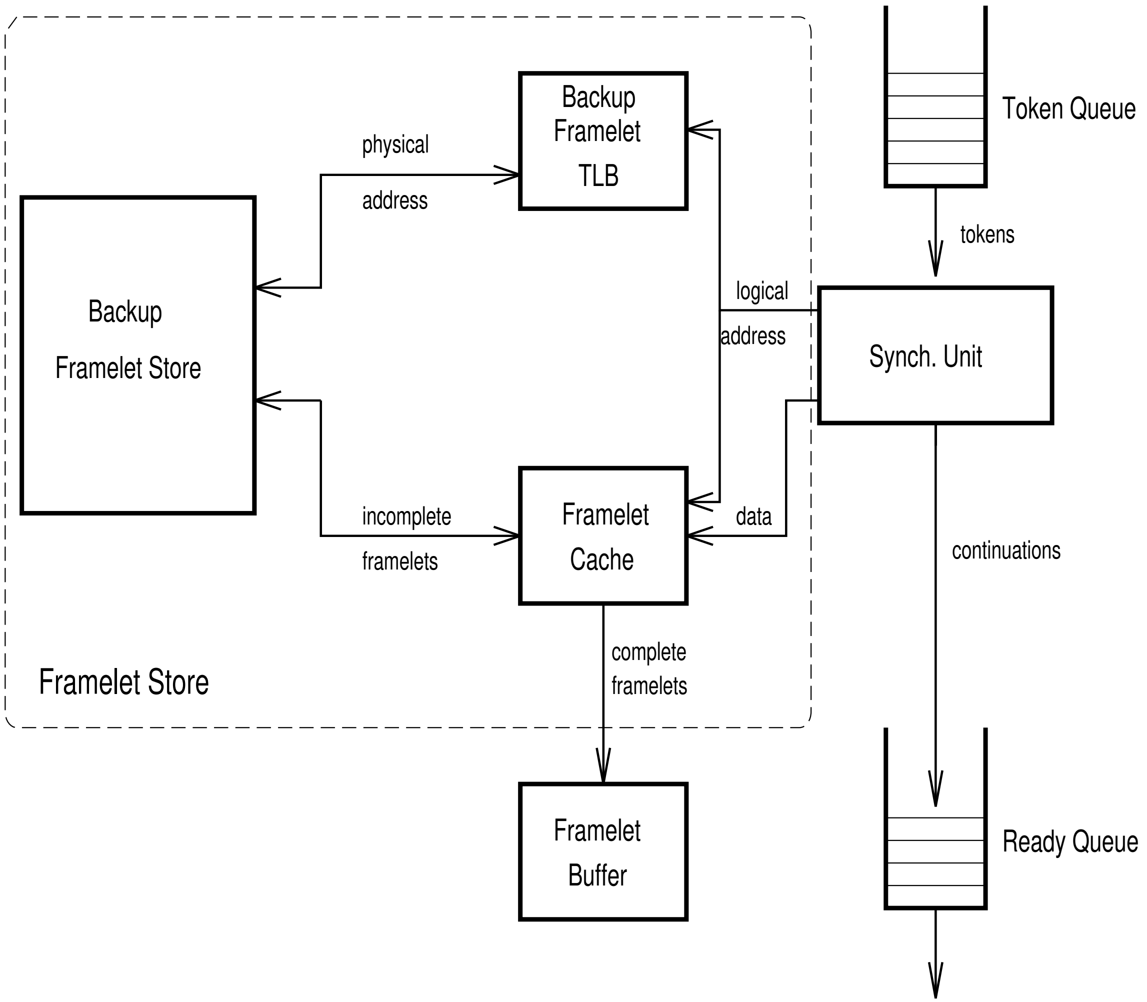
Address = <context#,thread#>

Fixed size framelets: 99% of all threads have ≤ 16 inputs.

Framelet size = one cache block.

Framelet complete:
  * deallocated: content moved to Framelet Buffer.
  * continuation sent to Ready Queue.
  * cache block marked *free*

Exclusive cache and TLB.

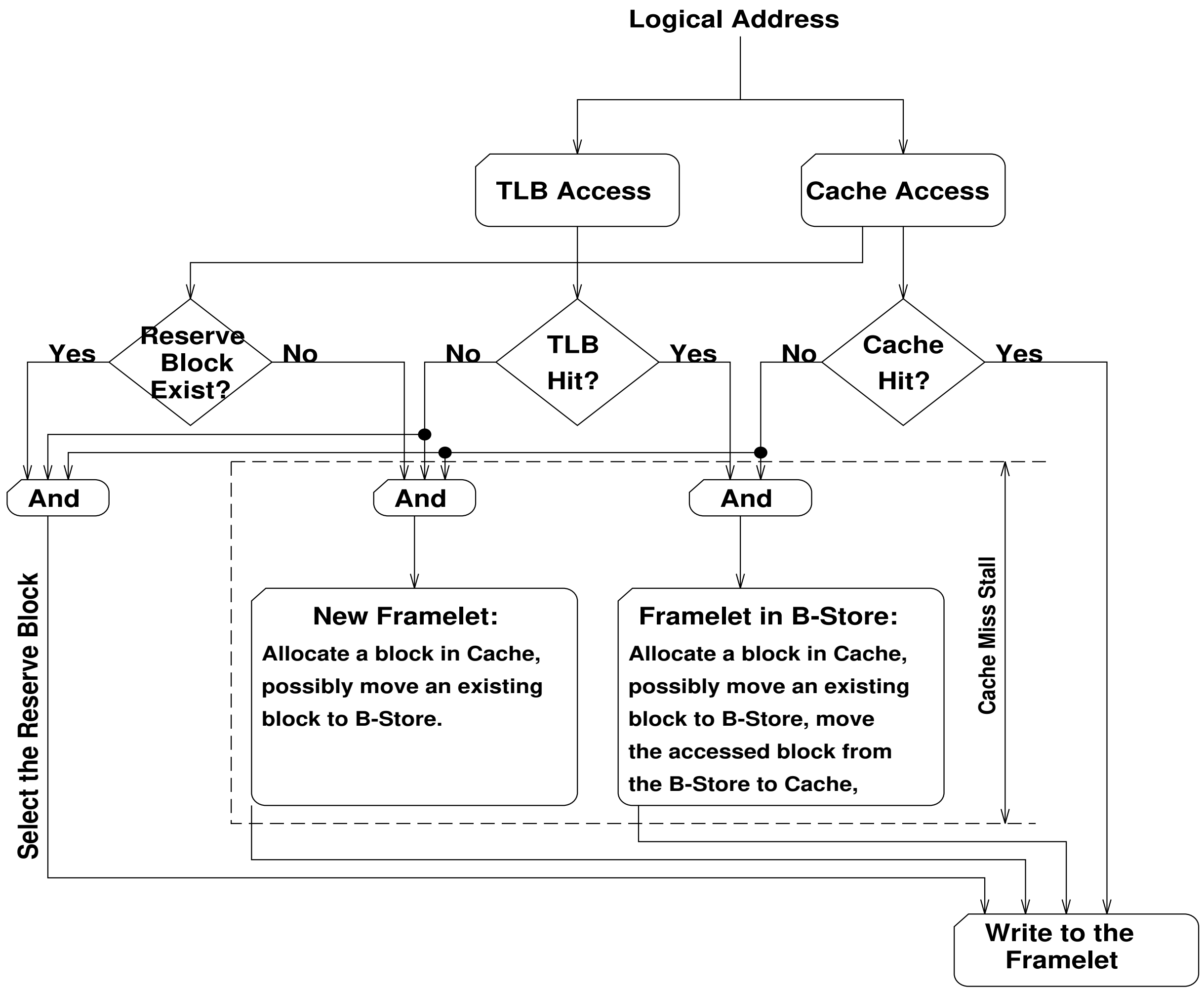


Reseve Block Cache Operation

Cache hit time: 2 cycles.

Reserve block: 1 free block/set reserved for compulsory misses $\Rightarrow$ 2 cycles.

Capacity miss: use free block when available; else evict and replace a block.



Cache Performance

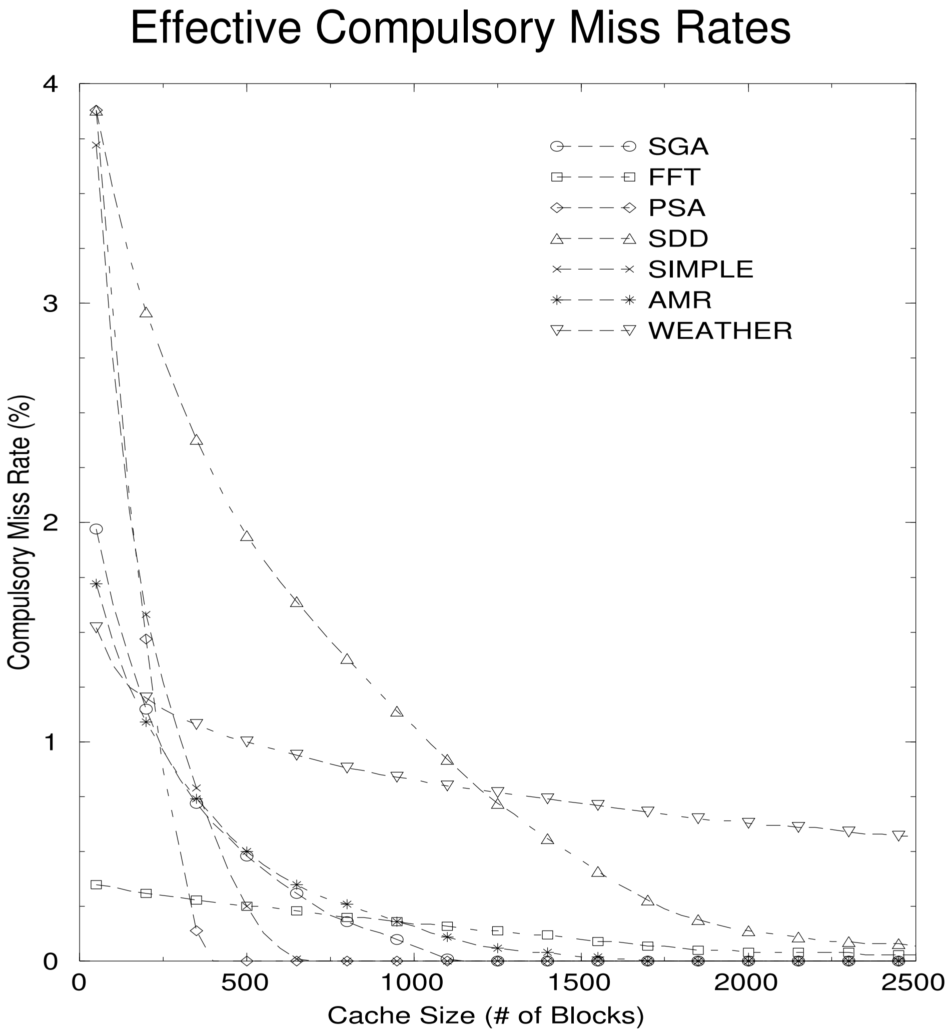
Compulsory miss: first reference to a logical framelet, misses in both cache and TLB.

Basic cache performance data (2000 blocks, fully assoc.):
- capacity misses: 0.34 to 3.52%, under 1% in all but two benchmarks (SDD & WEATHER).
- compulsory misses: 6.4 to 23.4%, high.

Reserve block: one free block per set is reserved for compulsory misses; reduces most compulsory misses.

LRU and FIFO: near identical performance, high regularity in the references.

Performance of Reserve Block



2000 blocks cache:

effective compulsory miss rate
< 0.25% (except WEATHER)

Overall Performance Results

Total Effective Miss Rates (%)		
Benchmark	Basic Cache	Reserve Block
SGA	16.4	0.01
FFT	5.79	0.14
PSA	24.0	0.01
SDD	19.8	0.45
SIMPLE	14.8	0.01
AMR	6.88	0.01
WEATHER	12.8	2.31

Effective miss rate < 1% (except WEATHER).

Effective synchronization cost < 3 cycles.

Further Work:
* set-assoc. cache
* victim cache.
* 32 & 64-bytes block.
⇒ effective miss rate < 2%.

* comparison to Frame based storage model.

Related Work

* Culler et al. (IFIP 93): Limits on multithreaded performance by storage hierarchy

* Takesu (IEEE TC 92): Cache design for fine-grained dataflow machine DFM-II.

* Kavi et al. (ISCA 95): Cache design for dataflow multithreaded machine.

* Tekkath & Eggers (ISCA 94): Effects of thread placement on cache contention, performance and design.