

Multicore performance demystified

Sigurt Bladt Dinesen
sidi@itu.dk

Advisor:
Peter Sestoft

June 4, 2018

Contents

| | |
|---|----------|
| Introduction+abstract | 2 |
| How to read this report | 2 |
| Background | 2 |
| Machine architecture for software designers (focus on memory hierarchy and coherency) | 2 |
| Java and memory (include JOL tool) | 2 |
| Memory operations are expensive (may be "the new FLOPS") | 2 |
| Refer bench-marks: (cyclic) Read-times vs simple math operation . . . | 2 |
| Method: benchmarking, limitations, and concessions (incl. background noise in bmarks, java/hw memory-layout impedance) | 2 |
| The types of "time waste" in multicore programming | 2 |
| "Cache-friendly" gets a new meaning | 2 |
| Experiments | 2 |
| Caches, MESI, store-buffers and invalidation queues (explanation and proof of false-sharing impact) | 2 |
| toyproblems (local int-counters, contended int-counters) | 2 |
| real problems (quicksort {div-and-conquer}) | 2 |
| real problems (kmeans, striped hashmap {locking}) | 2 |
| real problems (? {lock-free cas-based}) | 2 |

Introduction+abstract

How to read this report

Background

Machine architecture for software designers (focus on memory hierarchy and coherency)

Java and memory (include JOL tool)

Memory operations are expensive (may be "the new FLOPS")

Refer bench-marks: (cyclic) Read-times vs simple math operation

Method: benchmarking, limitations, and concessions (incl. background noise in bmarks, java/hw memory-layout impedance)

The types of "time waste" in multicore programming

"Cache-friendly" gets a new meaning

Experiments

Caches, MESI, store-buffers and invalidation queues (explanation and proof of false-sharing impact)

toyproblems (local int-counters, contended int-counters)

real problems (quicksort {div-and-conquer})

real problems (kmeans, striped hashmap {locking})

real problems (? {lock-free cas-based})

References