

Course 0: Introduction

Kristaps Dzonsons

01 December, 2011

Course site: *http://kristaps.bsd.lv/minicourse_12_2011*

Problem: Scale

So you have a...

hard scale limit

...supporting software can't handle your data, e.g.,
MatLab's data-size limits or type-width (bit) limits.

soft scale limit

...execution taking too much resources, e.g., runs
taking days or hardware limitations reached.

What do you do?

Solution: Change Environment

In practise, computation is divisible into three parts:

data your input, x (array of floats, list of doubles, lists or arrays, quotients or decimals);

instructions how you operate on your data, $f(x)$ (algorithm and software implementation);

environment your instructions' environment, $g(f(x))$ (NUMA, big-endian, 64- or 32-bit).

Performance: $\pi(g(f(x)))$

Goal: $\pi(g'(f'(x')))) < \pi(g(f(x)))$

Examples: Change

Example changes:

- ① increasing computation capacity
- ② use a different supporting library/language
- ③ convert sequential floating-point division to quotient multiplication
- ④ halving data, doubling processing units, parallelising the algorithm
- ⑤ vector-parallelising consecutive computation

Trade-offs: money (capacity), time (reprogramming)

Not all changes are possible. (Nor is this obvious!)

Mini-course Goals

Explore strategies for change beyond COTS software, Windows, and desktop-grade hardware. . .

UNIX

modern operating systems ideal for high-performance computing

C

simple, minimal programming language with overwhelming support for (and on) HPC

performance hardware

multiple processors and cores, vector extensions, . . .

community

sharing computing and knowledge resources

Mini-course Layout

- ① computing environment
What is the software environment of an HPC system?
- ② computing architecture
What is the hardware of HPC (or in general)?
- ③ measuring performance
How do I measure and compare performance?
- ④ tuning: low-hanging fruit
Attacking the problem: where to begin.
- ⑤ tuning: high-hanging fruit
Attacking the problem: tricks (and pitfalls).

Throughout: best practises and where to find help.