MONT-BLANC

http://www.montblanc-project.eu

Initial experiences with ARM processors

Matthew Street

matthew.street@rolls-royce.com

Rolls-Royce

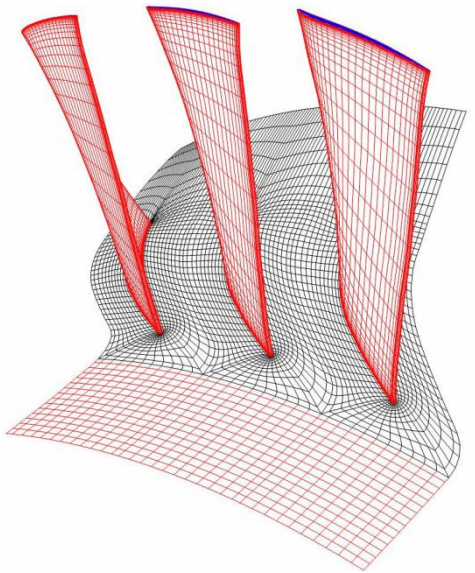
This project and the research leading to these results has received funding from the European Community's Seventh Framework Programme [FP7/2007-2013] under grant agreement n° 288777.

Tested application

- HYDRA
  - In-house Rolls-Royce CFD code
    - Unstructured edge-based multigrid solver
  - Fortran (mostly FORTRAN 77), some C
    - O(100,000) LOC, ~3000 intense lines
  - MPI distributed parallelism
    - MPI + X under development (using OP2 [1])
    - No accelerator capability
  - HDF5, CGNS, + in-house support libraries

- NASA Rotor 37
  - Open benchmark case
  - Steady state calculation

- All jobs run on Odroid cluster (others in progress)



[1] http://www.oerc.ox.ac.uk/projects/op2

http://www.grc.nasa.gov

Resources required

- Mont Blanc hardware & maximum wallclock restricted size of case
  - 1M gridpoints
  - 20 iterations
  - Usual benchmark = 8M gridpoints & 1000 iterations

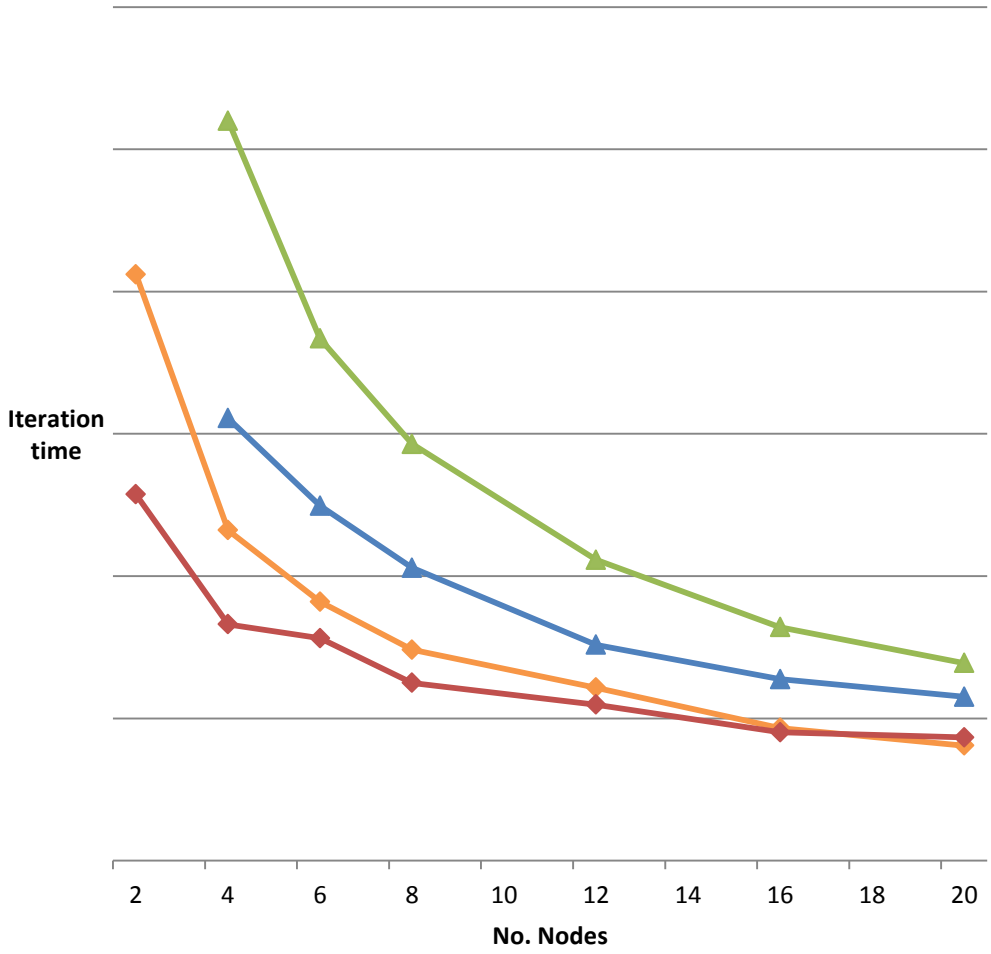
- Memory requirements
  - Usually require 4GB per core for production runs
  - 2GB per node on prototypes very limiting – needs minimum 2 nodes

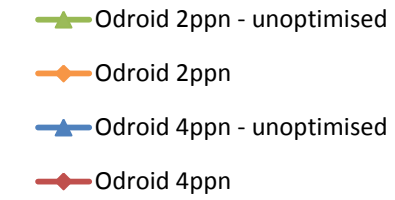
- 60MB input/output file for flow variables
- 440MB input files for mesh grids

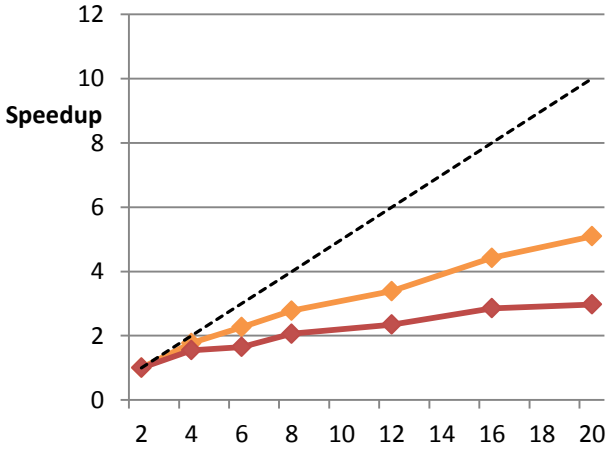
- Set-up time (~20 minutes) inflicts a minimum requirement of 4 cores
  - Iterations will not complete in remaining time otherwise
  - Ignoring setup costs in results - just used a timer for iterations

Performance & scalability

- Optimisation very significant

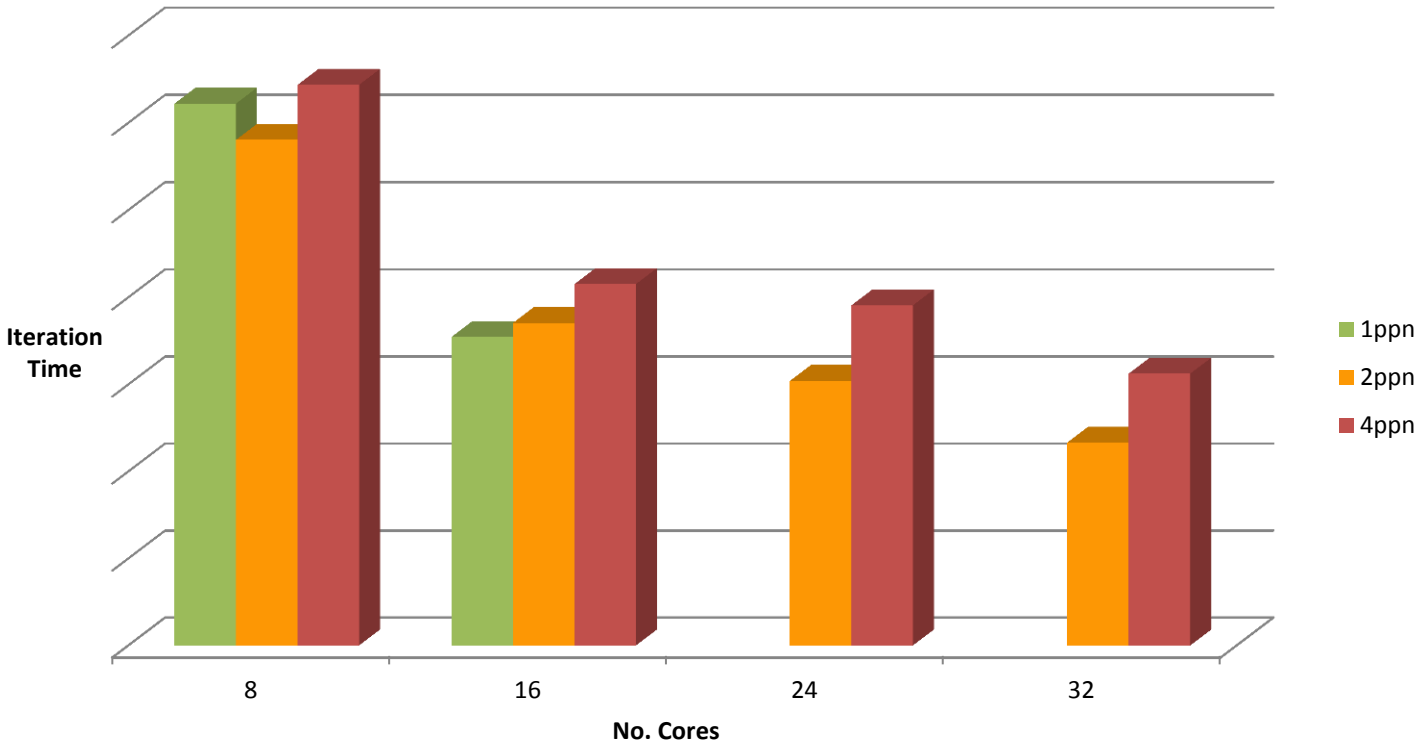






Effect of contention

- Using fewer cores per node is better
  - But using all the cores is better still – see previous slide

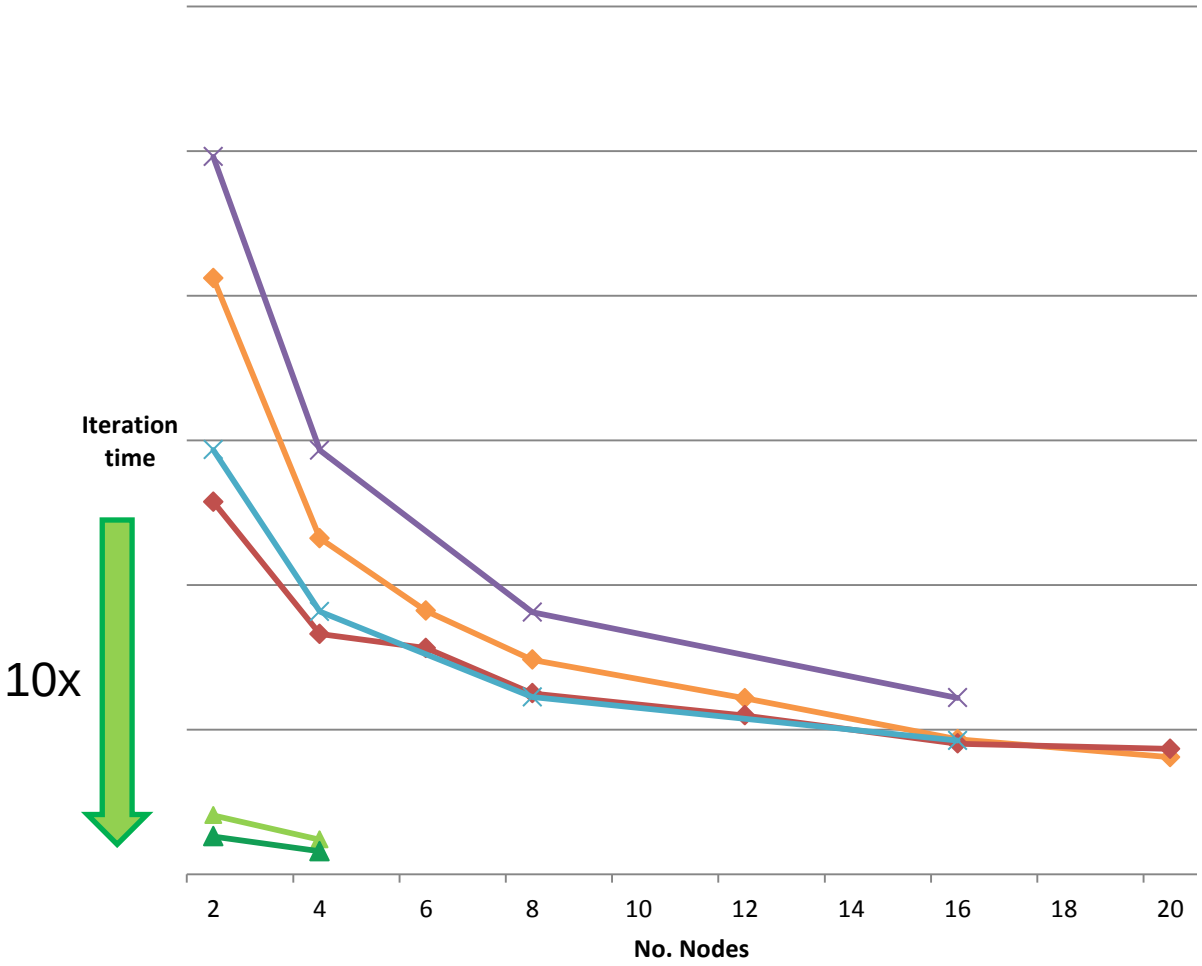


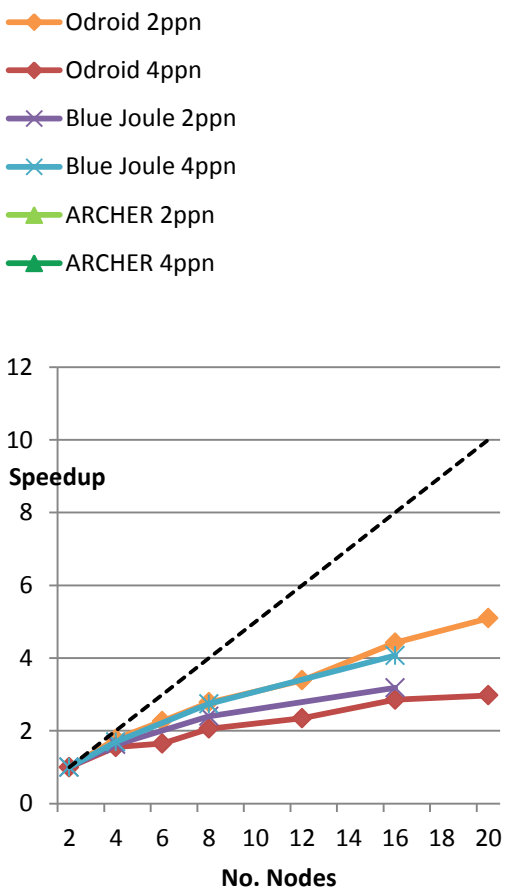
Energy consumption

- No real data, however playing with numbers per node (and ignoring network and disk and memory)
  - Odroid
    - 4-core ARM Cortex A15 1.6GHz (+ Cortex A7 4-core + PowerVR GPU)
    - **~5W**
  - BlueJoule Blue Gene/Q – Hartree Centre, UK
    - 16-core Power BQC 1.6GHz
    - **~55W**
    - → **Needs to be ~16x faster per node than Odroid**
  - ARCHER Cray XC30 – EPCC, UK
    - 2 x 12-core Intel Xeon 2697 v2 2.7GHz
    - **~260W**
    - → **Needs to be ~52x faster per node than Odroid**

Comparison to other platforms

- Core-for-core comparison





Comparison to other platforms

- Run on 2 fully occupied nodes

Platform	Speed up	Comparative power consumption
Odroid 2 x (4c 5W)	1x	1.0
ARCHER XC30 2 x (24c 260W)	26.7x	1.9
BlueJoule BG/Q 2 x (16c 55W)	2.1x	5.2

- (This is not the kind of job that should be run on BG/Q!)

Feedback on platform usability

- Numerical results are consistent between prototypes
  - With and without optimisation
  - Matches BG/Q & Intel Xeon results
- Not a straightforward porting exercise
  - Code hadn't been built with GCC in a long time
  - 'man gcc'
  - Helpdesk and support good, but more on-line documentation & system info would be helpful too
    - e.g. suggested compiler flags, MPI info
    - More pre-installed tools (debugger, profilers)
- Need more GB/core - preferred 2GB/core minimum

Feedback on platform usability

- As expected, no competition with Xeon / Cray XC30 for performance
  - But would probably use ARM in preference to BG/Q (for this class of job)
- At Rolls-Royce, ARM-based systems may be useful for certain classes of problems in the future
  - Small jobs when slower turnaround is acceptable (at lower cost)
  - e.g. shorter, overnight jobs

"The secret is to win going as slowly as possible."

- Niki Lauda, Formula 1 World Champion