Roofline model and OpenMP

September 14, 2016

The homework questions are due at the 23:59 on Friday 23 September. Please turn in source codes, compilation, submission scripts used and also output files. Please cite any references you use.

1 OpenMP

Homework question 1

a) What is OpenMP?

b) Download a copy of the latest OpenMP specifications from www.openmp.org. What version number is the latest specification?

c) Explain what each of the following OpenMP directives does:

```
!$OMP PARALLEL
!$OMP END PARALLEL
!$OMP PARALLEL DO
!$OMP END PARALLEL DO
!$OMP BARRIER
!$OMP MASTER
!$OMP END MASTER
```

Homework question 2

Compile and run the parallel bottom up merge sort program from [1] on between 1 and 20 cores on Rocket. Try different thread placement options (scatter, compact, balanced [4]) and problem sizes. Make a graph of your results. What do you conclude? It is strongly recommended to use a script to organize submission of your
jobs, collection of results and plotting. With modifications, you can use the same setup for running other benchmarks. Python (for example https://courses.cs.ut.ee/MTAT.08.037/2015_spring/uploads/Main/extract.py or the CacheExample in the program directory) or GNUplot (http://www.gnuplot.info/) are useful for automating plot generation.

2 Roofline model

**Homework question 3**

It can be difficult to obtain information on theoretical maximum floating point performance and memory bandwidth. One can then try to measure these experimentally.

a) Summarize what the stream triad benchmark [5] measures?

b) Run the stream triad benchmark on one core of Rocket for array sizes from 8 up until $2^{32}$ and record your output.

c) Run the stream triad benchmark on one node (20 cores) on Rocket and record your output. Try several options for thread placement (scatter, compact, balanced [4]).


e) Run the max flops code on one core of Rocket.

e) Run the max flops code on one node (20 cores) of Rocket. Experiment with thread placement (scatter, compact, balanced [4]) to get best performance.

**Homework question 4**

a) Can you find an implementation of the stream benchmark in CUDA?

b) Can you find an implementation of the stream benchmark in OpenCL?

c) On chips with integrated GPUs such as AMD APU, Intel HD graphics and some mobile phones, what do you think is the best method of measuring memory bandwidth? Should one make measurements on both the GPU and CPU parts at the same time or separately? It may be helpful to give a use case.

**Homework question 5**

a) Find out the computer chip and GPU in your mobile phone. What make and model are they [1]? What programming languages do they support?

---

1 If you cannot get this information, use information for another mobile phone chip and GPU.
b) See if you can find bandwidth information and floating point performance (single and/or double precision) for the computer chip and GPU in your mobile phone.

 c) Construct roofline models for the compute chip and GPU in your mobile phone.

d) How do these models compare to the roofline model for the Sunway TaihuLight?

3 Bonus

Disclaimer: The programs associated with these problems have not been fully tested, so you may encounter problems setting them up and running them. They may provide a basis for interesting projects.

Homework question 6

a) Try out NUMA-stream benchmark[2]. How does it compare to the STREAM benchmark?

b) Try out the Servo browser engine[8]. What benchmark results do you obtain on your computer? In what situations do you think a parallel web browser engine would be useful? Can the Servo browser engine utilize a GPU or other accelerator?

c) Run the peak flops and stream triad codes to develop a roofline model for the Xeon Phi co-processor on Rocket. How does the roofline model you develop compare to that for a recent high end Nvidia or AMD GPU (choose one specific model to compare to in your answer). Reference [6] may be helpful for you.

References


