XDS news August 2016

Kay Diederichs

Universität Konstanz

Xeon Phi “Knights Landing” (KNL)

64, 68 or 72 cores in a single socket; 4-fold (efficient!) Hyper-Threading; ≤ 384 GB ECC-DDR4; 16 GB built-in high-bandwidth (5 times faster) MCDRAM

Boots and runs Linux (e.g. RHEL7) natively; has many new features e.g. AVX512, a vector instruction w/ 512 bits (AVX=256 bits; SSE=128 bits); needs recompilation for highest performance

Vector peak performance: >3 TeraFlops Double Precision; >6 TeraFlops SP

System tried: 1 KNL 7210 processor; 64*4=256 threads; 192 GB RAM; 750W power supply; (e.g. workstation SuperServer 5038K-i ~6.000€)

Note A: until now, systems reaching 3 TeraFlops cost >60.000€ (8 * Xeon E7)
Note B: SuperServer 5028TK-HTR: 2U-rackmount; 4 KNL 7210 (>1.000 threads): ~20.000€

First findings on KNL

Test data: 3600 Eiger 16M frames; HDF5 (using `H5ToXds`)

- XDS runs “out of the box”; re-compiled for <=72 threads (old: <=32 threads)
- Compiling (ifort v16 or v17 beta) with `-xMIC-AVX512`: ~30% faster
- Using MCDRAM instead of normal RAM in “Flat” mode: 5% faster in COLSPOT; >10% faster in INTEGRATE (`memkind` not yet tried; “Cache” and “Hybrid” mode not yet tried);
- NUMA configuration & environment variables need special attention
- Work in progress; system looks promising but too early to summarize

References: http://colfaxresearch.com/get-ready-for-intel-knights-landing-3-papers/

Parallelization

COLSPOT: reads several degrees of data; for speedup parallelized on two levels (threads + shell-level) i.e. can use many cores / several computers

INTEGRATE: as COLSPOT but reads all data

INIT: reads (typically) only the first 5 degrees of data. Used to be **the only serial part of XDS data processing**. For big machines/clusters this meant that up to 1/3 of wallclock time was spent in INIT.

INIT was rewritten; it is now parallel (OpenMP threads) - the speedup depends on the number of threads. For typical 180° data sets of which 5° are used for INIT, the wallclock time spent in INIT can now be considered insignificant.

Native reading of data files using a dynamically (i.e. at runtime) loaded library

(this is old news; for more information see talks at HDRMX meeting in May 2016)

A generic interface for this purpose was implemented

Currently, it is only used by the Dectris HDF5 Plug-In
https://github.com/dectris/dectris-xds-plugin

Removes the overhead of H5ToXds (or similar programs)

Can be used to read arbitrary files, if the Producer provides a library

XDS version

- for KNL
- with INIT parallelization
- native HDF5 reading

planned to be released soon, i.e. in 2016.