Intel Skylake review

Martin Cuma, CHPC

In this article we look at the performance of the Intel Skylake Xeon CPU platform released in July 2017, and compare them to the previous generation Broadwell type CPUs.

The Skylake Xeon (called Xeon Scalable processor) introduced a number of innovations, notably AVX-512 vectorization instructions capable of 8-wide double precision vectors (previous AVX2 had 4-wide DP vectors). This change in itself has a potential of doubling performance of floating point codes. Other changes include CPU core optimizations, rearchitecture of the caches, and new, mesh-based topology of the cores, which allows for higher bandwidth and lower latency between the cores. All is well described in Tom's Hardware article,
http://www.tomshardware.com/reviews/intel-xeon-platinum-8176-scalable-cpu,5120.html

The Skylake Xeons are divided into four categories based on performance of the CPUs. The highest, and most expensive tier is called "Platinum", the topmost one's list price is ~$13,000, the lowest one list price is $3115 and one may roughly compare them to the previous E7 CPUs. These are designed for large SMP servers. The second tier, "Gold", is what we are after for the HPC centers, having up to 22 cores, and costing ~$1,200-~$3,500. There are also "Silver" and "Bronze" tiers which have lower core counts, memory speeds and Ultra Path Interconnect (UPI) transfer rates. A good comparison of all the tiers is here:
https://www.microway.com/knowledge-center-articles/detailed-specifications-of-the-skylake-sp-intel-xeon-processor-scalable-family-cpus/

Dell got us access to Xeon Gold 6130 dual-CPU nodes. These CPUs have 16 cores each (32 cores per node) at 2.1 GHz base frequency. MSRP of this CPU is ~$1,900 and gives one of the best dollars/flops ratio based on the Microway article above, and will probably be the recommended CPU to get for our researchers.

We compare this performance to our stock Broadwell Xeon E5-2680 v4, which has 14 cores (28 cores per node) at 2.4 GHz and lists at $1,745.

External benchmarks

There have not been too many articles online showing Skylake benchmarks. Two of these, Tom's Hardware, http://www.tomshardware.com/reviews/intel-xeon-platinum-8176-scalable-cpu,5120.html, and Anandtech,
http://www.anandtech.com/show/11544/intel-skylake-ep-vs-amd-epyc-7000-cpu-battle-of-the-decade, look at the high end Platinum models, though the Xeon Platinum 8176 runs at 2.1 GHz, the same frequency as our Gold 6130 CPU, so, single core CPU performance should be comparable (though the boost frequency is 3.8 vs. 3.7 GHz).
The Anandtech article has good discussion on the memory and integer performance, while the Tom's Hardware covers better floating point performance.
Dell also published a short LAMMPS benchmark result,
http://en.community.dell.com/techcenter/high-performance-computing/b/general_hpc/archive/2017/08/04/lammps-four-node-comparative-performance-analysis-on-skylake-processors. However, this article compares a 2.7 GHz Skylake to 2.3 GHz Broadwell which gives the Skylake higher boost. Dell also recently published a comparison of the 4-way SMP nodes with the Skylake Platinum CPU,

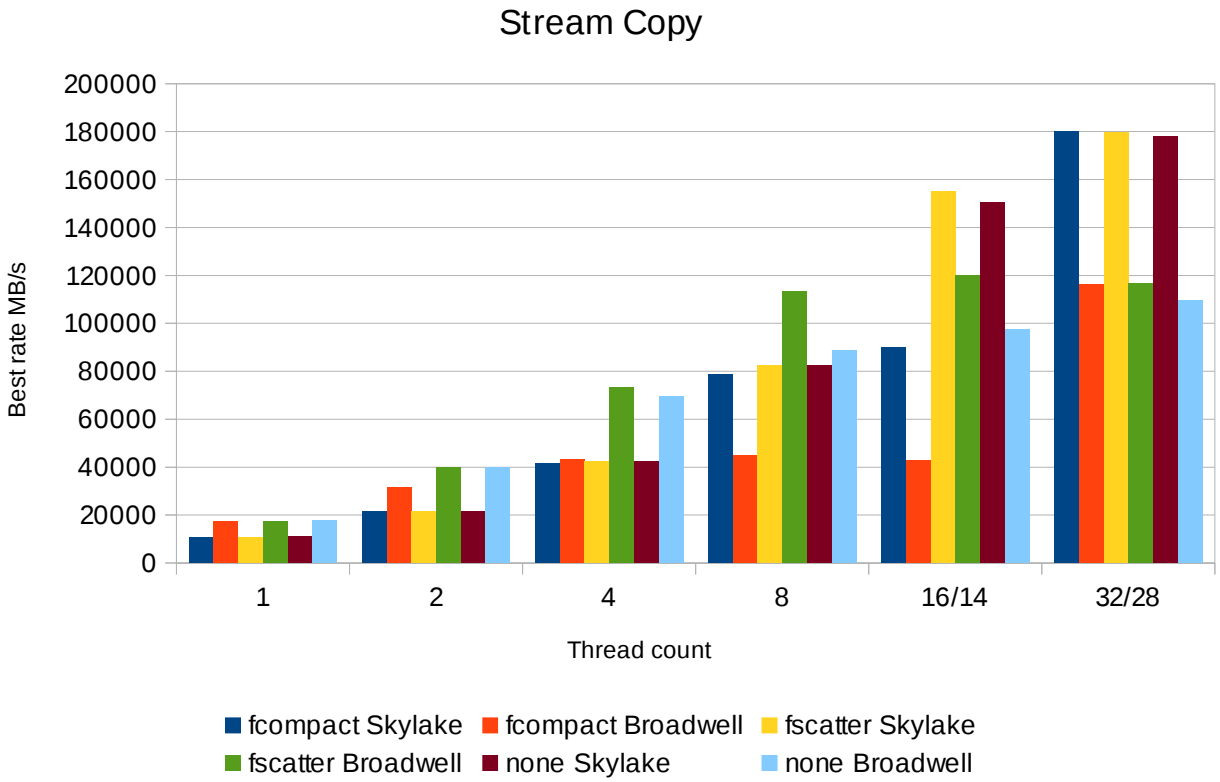
http://en.community.dell.com/techcenter/high-performance-computing/b/general_hpc/archive/2017/08/21/performance-study-of-four-socket-poweredge-r940-server-with-intel-skylake-processors, which shows similar trends as our results below, but on a hardware that most of our researches would not buy.

Raw and synthetic performance benchmarks

STREAM benchmark

The STREAM benchmark tests the bandwidth from CPU to the main memory by performing four different operations on large sequential data arrays. We have compiled STREAM using the Intel 2017.4 on both the Skylake and Broadwell with the host based optimizations. STREAM is thread parallelized using OpenMP and we look at the throughput from one thread to the number of threads equal to the number of the physical cores. As both the Skylake and the Broadwell machines have two NUMA CPUs, we also look at the effect of the thread locality to the CPU core, examining three ways to pack the threads to the cores, *compact* - where first all the cores on CPU 0 get filled, followed by CPU 1, *scatter*, where the threads get packed on the two CPUs in a round robin fashion, and *none*, where we let the OS to float the treads on the CPU cores.

Stream consists of four benchmarks, but all of them show similar trends so we only show result for the Copy in Figure 1. There are a few points to be made from this graph:
- The Skylake's single core memory bandwith of ~ 13 MB/s is lower than Broadwell's ~18 MB/s.
- The Skylake's total memory bandwidth of ~ 180 MB/s is considerably larger than Broadwell's ~125 MB/s.
- Scatter thread packing provides higher memory bandwidth than compact - which makes sense since it's using memory channels from both CPUs.
- no thread affinity (none), apart from lower memory bandwidth, also exhibits large variability from run to run.



High Performance Computing Challenge (HPCC) benchmark

HPCC benchmark is a synthetic benchmark suite geared at assessing HPC performance from different angles. It consists of seven main benchmarks that stress various computer subsystems, such as raw performance, memory access and communication. For detailed description of the benchmark see http://icl.cs.utk.edu/hpcc/.

For the Skylake and Broadwell, we have built HPCC 1.5.0 with Intel 2017.4 compilers and corresponding Intel MKL and MPI using the following compiler optimization flags:
-O3 -ansi-alias -ip -axCORE-AVX512,CORE-AVX2,AVX,SSE4.2 -restrict.
Older benchmark results have used similar flags with previous versions of Intel compiler.

Year	2017	2016	2014	2012	2010
CPU generation	Skylake	Broadwell	Haswell	SandyBridge	Westmere
Core count	32	28	24	16	12
Frequency_GHz	2.1	2.4	2.5	2.2	2.8
HPL_Tflops	1.64	0.85	0.73	0.27	0.12
StarDGEMM_Gflops	54.04	31.98	31.83	17.08	10.46
SingleDGEMM_Gflops	56.09	41.41	41.72	20.30	10.71
PTRANS_GBs	13.94	10.84	7.39	4.62	3.05
MPIRandomAccess_GUPs	0.0026	0.0037	0.0266	0.0171	0.0427
StarRandomAccess_GUPs	0.0397	0.0304	0.0256	0.0292	0.0196
SingleRandomAccess_GUPs	0.0787	0.0825	0.0778	0.0611	0.0366
StarSTREAM_Triad	4.55	3.26	2.55	3.42	2.48
SingleSTREAM_Triad	12.57	10.55	12.93	12.50	10.25
StarFFT_Gflops	2.06	1.67	1.53	1.51	1.22
SingleFFT_Gflops	2.75	2.31	2.38	2.03	1.95
MPIFFT_Gflops	29.88	11.93	8.53	7.90	4.64

Table 1. HPCC results, the higher the value the better.

In Table 1 we show the result of select HPCC metrics for select fully loaded nodes Intel Xeon CPUs since 2010. Focusing on the Skylake vs. Broadwell, we see a significant performance increase for most benchmarks. The Single benchmarks are run on one core so their improvement is not that significant, since the Skylake clock speed is lower.

To visualize the improvement in floating point performance, in Figure 1 we show the High Performance Linpack (HPL) performance of the different Xeon generations, which is examplifies the change in the floating point (FP) vectorization units. The 2010 Westmere CPU had SSE4.2 vectorization set capable of doing 2 double precision operations (DPO) per cycle. This has doubled to 4 DPO/cycle in 2012 SandyBridge with the AVX instruction set. The 2014 Haswell's AVX2 added Fused Multiply Add (FMA) instruction, which, along with the increase in core count and clock speed as compared to our benchmarked SandyBridge more than doubled the floating point output. Broadwell CPU was a process shrink of Haswell so the extra performance was added mainly by the increased core count. Going to Skylake, we are seeing another doubling of FP performance with the 8 DP long AVX512 instruction set.

HPL_Tflops	High Performance Linpack benchmark - the one that's used for Top500 - measures the floating point rate of execution for solving a linear system of equations.
StarDGEMM_Gflops	Parallel DGEMM - measures the floating point rate of execution of double precision real matrix-matrix multiplication.
SingleDGEMM_Gflops	Serial DGEMM - on single processor
PTRANS_GBs	Parallel Matrix Transpose - exercises the communications where pairs of processors communicate with each other simultaneously. It is a useful test of the total communications capacity of the network.
MPIRandomAccess_GUPs	MPI Parallel Random Access

StarRandomAccess_GUPs	UPC Parallel Random Access - measures the rate of integer random updates of memory (GUPS).
SingleRandomAccess_GUPs	Serial Random Access
StarSTREAM_Triad	Parallel STREAM - a simple synthetic benchmark program that measures sustainable memory bandwidth (in GB/s) and the corresponding computation rate for simple vector kernel.
SingleSTREAM_Triad	Serial STREAM
StarFFT_Gflops	Parallel FFT - measures the floating point rate of execution of double precision complex one-dimensional Discrete Fourier Transform (DFT).
SingleFFT_Gflops	Serial FFT
MPIFFT_Gflops	MPI FFT

Table 2. HPCC explanations.

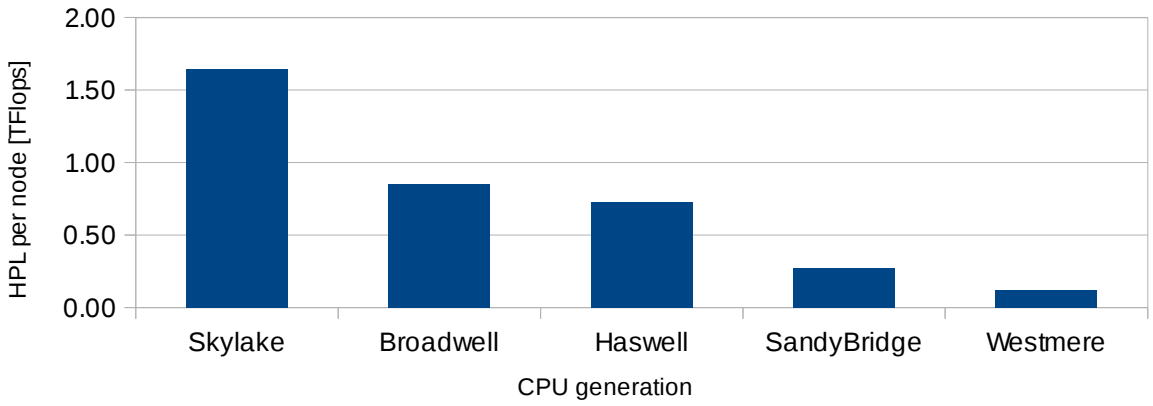


Figure 1. Top HPL performance for the select Intel CPU generations. Higher value is better.

NAS Parallel Benchmarks

NAS Parallel Benchmarks are a set of programs derived from computational fluid dynamics (CFD) applications. Some basic information about the benchmarks is here: https://en.wikipedia.org/wiki/NAS_Parallel_Benchmarks. Each of these benchmarks can be run with different problem sizes. Class A is a small problem, Class B is medium size, Class C is a large problem, and Class D is a very large problem (needing about 12 GB of RAM). There are also even larger classes E and F. We have ran Classes A-D and present results for Class C. We have compiled the codes with Intel 2017 compilers, using "-O3 -ipo -axCORE-AVX512 -qopenmp" option on the Skylake and "-O3 -ipo -axCORE-AVX512 -qopenmp" option on the Haswell.

All the NAS benchmark plots compare the performance in Mops/sec or Mops/sec/thread. As we are looking at comparing maximum performance on the whole multi-core machine, and also evaluating the SMP capabilities, below we look at the Mops/sec. The higher is the Mops/sec count, the better. We present the benchmarks in two graphs broken by the Mops/sec value for better comparison.

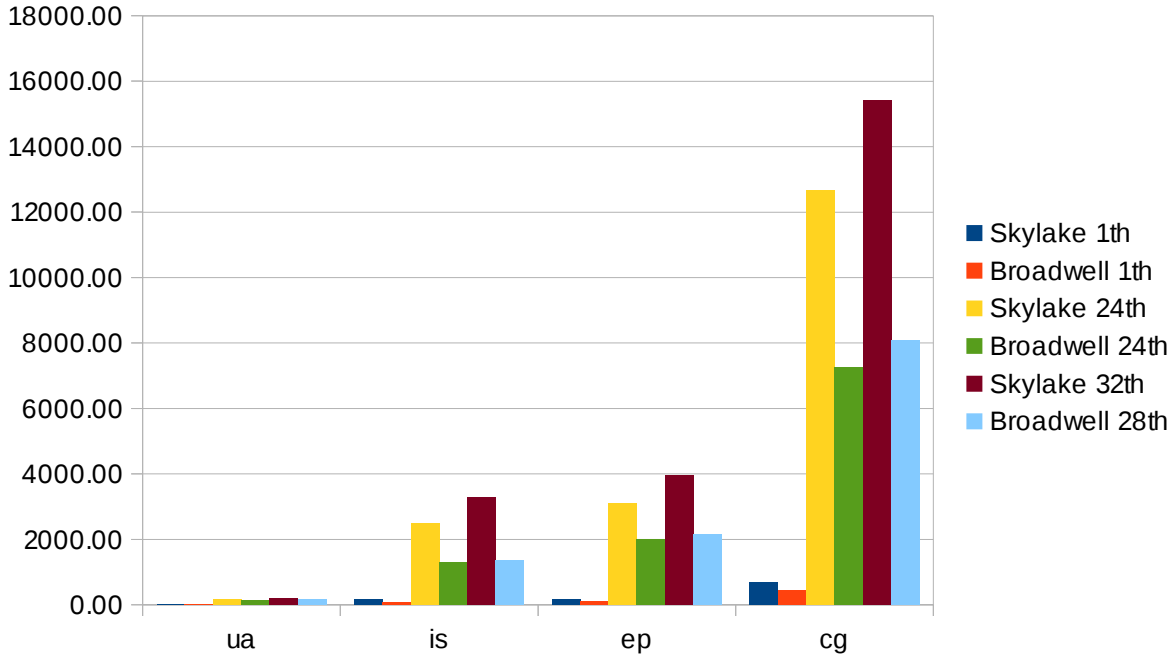


Figure 2a. NAS UA, IS, EP and CG benchmarks for size C

IS, EP, CG, SP and MG shows double the performance on the Skylake, which suggests that these codes are well vectorizable. The other benchmarks performance improvement is not as high, but, with increased core count it's still significant. The only exception is FT, where Broadwell performs slightly better. We will need to spend some time to understand this benchmark and explain this.

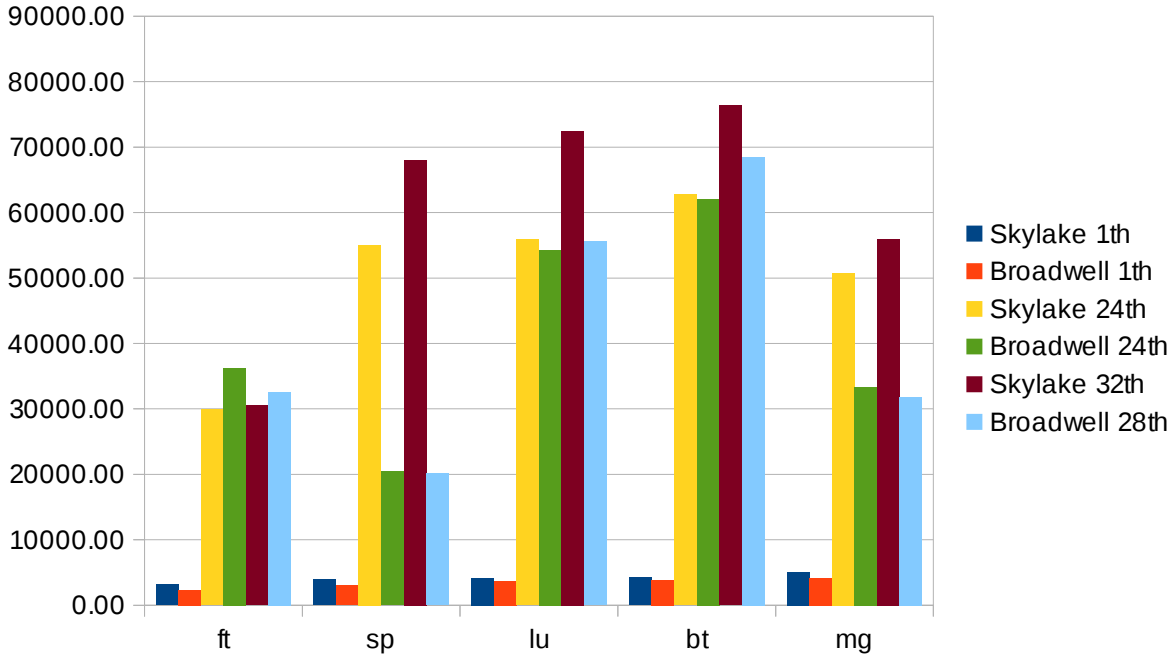


Figure 2b. NAS FT, SP, LU, BT and MG benchmarks for size C

Synthetic benchmarks conclusion

Synthetic benchmarks suggest that one can expect up to 100% application speedup with the new Skylake platform, based both on the floating point throughput and memory bandwidth.

Real applications benchmarks

LAMMPS

LAMMPS is a popular molecular dynamics simulation program developed at Sandia National Laboratory. We have built the 31Mar17 version using Intel 2017 compilers, MPI and MKL (using MKL's FFTW wrappers) and with optimization flags "-axCORE-AVX512,CORE-AVX2,AVX,SSE4.2 -O3 -prec-div -fp-model precise".

We have run three LAMMPS benchmarks from http://lammps.sandia.gov/bench.html:
LJ = atomic fluid, Lennard-Jones potential with 2.5 sigma cutoff (55 neighbors per atom), NVE integration
Chain = bead-spring polymer melt of 100-mer chains, FENE bonds and LJ pairwise interactions with a 2^(1/6) sigma cutoff (5 neighbors per atom), NVE integration
EAM = metallic solid, Cu EAM potential with 4.95 Angstrom cutoff (45 neighbors per atom), NVE integration
Each problem was scaled 2x in each dimension resulting in 256,000 atoms and was run for 1,000 time steps.

In Table 3 we show the benchmark results for Skylake and Broadwell. The Skylake performs consistently 25-30% faster than the Broadwell. One thing to keep in mind, though, is that I built the LAMMPS fairly standardly without additional packages, like Kokkos, which may provide subsequent many-core acceleration in mixed MPI-threading mode. We may want to investigate this in the future.

	chain			eam			lj		
	Skylake	Broadwell	Speedup	Skylake	Broadwell	Speedup	Skylake	Broadwell	Speedup
1	74.26	78.00	1.05	305.00	378.18	1.24	117.24	141.46	1.21
2	35.04	37.99	1.08	155.34	194.64	1.25	59.08	71.51	1.21
4	17.74	20.09	1.13	80.93	103.33	1.28	31.01	37.63	1.21
8	9.37	10.69	1.14	43.73	56.71	1.30	16.76	21.00	1.25
16	4.80	6.00	1.25	22.75	30.44	1.34	8.66	11.24	1.30
24	3.45	4.15	1.20	16.50	20.53	1.24	6.30	7.66	1.22
32/28	**2.89**	**3.66**	**1.27**	**13.87**	**18.34**	**1.32**	**5.22**	**6.68**	**1.28**

Table 3. LAMMPS performance on Skylake and Broadwell (in seconds, lower is better) and the Skylake speedup wrt. Broadwell. Last bold line represents the whole node.

The Dell benchmark presented at http://en.community.dell.com/techcenter/high-performance-computing/b/general_hpc shows a more optimistic doubling of performance with Skylake. However, it compares a higher clock, 2.7 GHz 18 core Xeon Gold 6150 (list price $3358) with a lower clock speed 2.3 GHz 16 core Xeon E5-2697 (list price $2614), which is a less fair comparison for the older CPU.

VASP

VASP is a plane wave electronic structure program that is widely used in solid state physics and materials science. CHPC has several heavy users of VASP. We have compiled VASP 5.4.4 with Intel 2017 compilers, MKL and MPI, and "-O2 -axCORE-AVX512,CORE-AVX2,AVX,SSE4.2" compiler flags.

We present two benchmarks of semiconductor based systems, Si and SiO, the SiO being several times larger, and one even larger chemical system, MoS_2.The smallest system is slowly becoming less relevant as both the hardware and the software improve, so, in our explanations we focus on the larger problems. As with the HPCC, we include results we obtained on previous generation of processors in Table 4, though, beware that the older CPUs were run with older VASP version which was potentially less optimized. The results are runtime in seconds, so, the smaller number the better.

(Si 12 layer, 24 at., 16 kpts, 60 bnds)								
	1 CPU	2 CPU	4 CPU	8 CPU	12 CPU	16 CPU	24 CPU	28/32 CPU
Westmere-EP 2.8 12c	233.49	123.05	68.79	51.73	47.13			
Sandybridge 2.2 16c	195.83	102.24	56.15	36.17	29.66	36.71		
Haswell 2.5 20c	118.02	56.70	34.58	22.13	20.48	15.74	27.06	
Broadwell 2.4 28c	108.46	55.31	30.06	19.25		12.84	13.52	13.85
Skylake 2.1 32c	80.41	41.60	22.78	15.50		11.33	11.08	9.30
Skylake speedup	**1.35**	**1.33**	**1.32**	**1.24**		**1.13**	**1.22**	**1.49**
(Si192+O, 4 kpts, 484 bnds)								
	1 CPU	2 CPU	4 CPU	8 CPU				
Westmere-EP 2.8 12c	999.36	514.66	330.20	210.14	175.22			
Sandybridge 2.2 16c	771.53	396.33	215.07	128.79	97.49	120.68		
Haswell 2.5 20c	424.72	187.93	116.83	76.69	66.32	57.79	41.52	
Broadwell 2.4 28c	395.01	163.62	91.65	55.61		41.63	34.36	35.09
Skylake 2.1 32c	278.25	144.49	75.63	45.29		32.17	26.25	27.92
Skylake speedup	**1.42**	**1.13**	**1.21**	**1.23**		**1.29**	**1.31**	**1.26**
(MoS2 300 atoms, 1 kpt, 1560 bnds)								
Broadwell 2.4 28c	8773.25	4365.48	2343.30	1392.68		861.56	759.81	649.25
Skylake 2.1 32c	6261.62	3292.76	1625.42	991.83		613.28	499.79	446.49
Skylake speedup	**1.40**	**1.33**	**1.44**	**1.40**		**1.40**	**1.52**	**1.45**

Table 4. VASP performance in seconds (lower is better)

Focusing on the comparison between the older Broadwell and newer Skylake CPU, we notice 25-50% improvement in performance, both per core and over the whole node. This is a little better than LAMMPS, presumably because there is more vectorizable operations in the linear algebra heavy code like VASP. Also note that we also ran the AVX2 (not AVX512) VASP binary on the Skylake, and the performance was about the same. This is because most of the computation is in the BLAS and LAPACK MKL routines which automatically pick the right vectorization for the given CPU.

	CPUs/node	Cores/CPU	Si	Rel. perf.	SiO	Rel. perf.	MoS2	Rel. perf.
Westmere-EP 2.8	2	6	47.13	0.78	175.22	0.69		
Sandybridge 2.2 16c	2	8	36.71	1.00	120.68	1.00		
Haswell 2.5	2	12	27.06	1.36	41.52	2.91		
Broadwell 2.4	2	14	12.84	2.86	35.09	3.44	649.25	1.00
Skylake 2.1	2	16	9.30	3.95	27.92	4.32	446.49	1.45
K80 GPU	4	a lot			54.60	2.21	318.08	2.04

Table 5. Best VASP performance for each CPU, and relative to SandyBridge (Si, SiO) and Broadwell (MoS2).

In Table 5 we compare the best performance per node for each of the CPUs we look at, and add a benchmark result from two NVidia K80 GPU cards (4 GPUs total). For the two smaller benchmarks, we also list relative performance to SandyBridge node; for the MoS2 the reference is a Broadwell node. The SiO GPU result is skewed as the benchmark is not big enough to efficiently load up the GPU. We can see that a Skylake node has about 4x the performance of the SandyBridge node, and, the two K80 cards are only about 40% faster than the Skylake. We should eventually benchmark the P100 GPUs to see how far will they get, though the initial attempts segfaulted so we'll have to find some time to explore the cause.

TSEM

TSEM is a geophysical electromagnetic inversion code developed at CHPC in collaboration with Consortium for Electromagnetic Modeling and Inversion (CEMI) at the University of Utah's Department of Geology and Geophysics. This particular benchmark inverts electromagnetic data from towed streamer used in marine hydrocarbon exploration.

The code consists of several modules that are called sequentially in an iterative fashion. In particular, there is an inversion module, which mainly does local matrix and vector operations with small amount of communication and some file I/O. Then there is a forward modeling stage, which does a lot of communication, lot of vector-matrix multiplication and some FFT and which takes the bulk of the run time. Finally, there are several precomputational stages which are computationally heavy but embarrassingly parallel. The computational kernel here is a legacy greens functions library which are not fully vectorizable. All portions of the code are parallelized at two levels, on coarse level with MPI and on fine level OpenMP. Both MPI and OpenMP scalability vary widely with the problem that's being computed due to inefficiencies in parallel distribution of work and the amount of work to be distributed.

The main reason we include this program is that most of the data are complex and vectorization of complex operations has not been very efficient until the AVX instructions. We are hoping that the AVX512 will have a noticeable impact on the code performance.

Also, pinning both MPI tasks and OpenMP threads can have some effect on the performance, which is why we pinned both MPI tasks (which is done automatically by Intel MPI), and OpenMP threads with the "KMP_AFFINITY granularity=fine,compact,1,0" flag on the Broadwell and "KMP_AFFINITY granularity=fine,compact" on the Skylake (as the Skylake nodes had the Hyperthreading disabled).

MPI tasks		32/28	16/14	8/7	4
OpenMP threads		1	2	4	8/7
Domain Greens tensors	Broadwell	440.38	437.72	**437.08**	440.85
	Skylake	344.48	342.18	**343.00**	344.11
	Skylake speedup	1.28	1.28	1.27	1.28
Forward Modeling	Broadwell	3041.56	2343.43	1411.69	**1274.24**
	Skylake	1709.86	1226.91	**986.48**	1061.32
	Skylake speedup	1.78	1.91	1.43	1.20
Total runtime	Broadwell	3628.68	2944.78	2062.44	**1997.58**
	Skylake	2187.67	1696.65	**1486.39**	1638.71
	Skylake speedup	1.66	1.74	1.39	1.22

Table 6. TSEM inversion performance on Broadwell and Skylake, runtimes in seconds, lower is better.

In Table 6. we show select performance characteristics of the TSEM code for different MPI task and OpenMP thread count. The Greens tensors scale linearly and as such they perform about the same for all task and thread count, with the Skylake providing about 28% speedup over the Broadwell. The forward modeling brings about more dramatic increase in speed, partly due to the improved vectorization, and party to better mapping of tasks and threads on the 32 Skylake cores. Notice that too much threading decreases the performance as the OpenMP loops become more granular.Similarly, no threading causes significant overhead over the many MPI tasks. The sweet spot is somewhere in the middle, with 8 (Skylake) or 4 (Broadwell) MPI tasks. Overall, we see about 35% performance increase in the Skylake as compared to the Broadwell.

Conclusions

The Skylake architecture brings a significant improvement in performance. Programs that consist mostly of dense linear algebra operations can expect up to 100% speedup as compared to the previous Broadwell CPUs in the similar price range. Realistic speedup for real applications can be expected between 30% and 50% per dual CPU node.