KNL Performance Comparison: *OpenSBLI*

March 2017

1. Compilation, Setup and Input

Compilation

OpenSBLI (https://opensbli.github.io/download) has been compiled on both the ARCHER KNL system and ARCHER Xeon system using the Cray compiler environment:

Compiler/Library	ARCHER Xeon	ARCHER KNL
Cray Compiler	cce/8.4.1	cce/8.5.4
Cray Parallel HDF5	Parallel HDF5 1.8.14	Parallel HDF5 1.10.0

Instructions on how to install the application on both systems can be found at https://opensbli.readthedocs.io/en/latest/getting_started.html - installing-opensbli

Here we have built the `OpenSBLI_mpi_openmp` version.

Setup

ARCHER KNL nodes in "quad_100" mode were used in all cases.

Input

We have used a strong scaling (SS algorithm) benchmark test case with a 512^3 grid points, 10 iterations and disabled I/O.

```
$cat input
ss 512 10 0 False
```

2. Performance Data

Figure 1 represents the performance comparison of the two best combination of processes per node (PPN), hyperthreads(HT) and OpenMP threads (T) configuration on ARCHER Xeon and ARCHER KNL. This corresponds to 24 processes per node and 1 or 2 hyperthreads for Xeon and 64 or 128 processes per node and two hyperthreads for KNL.

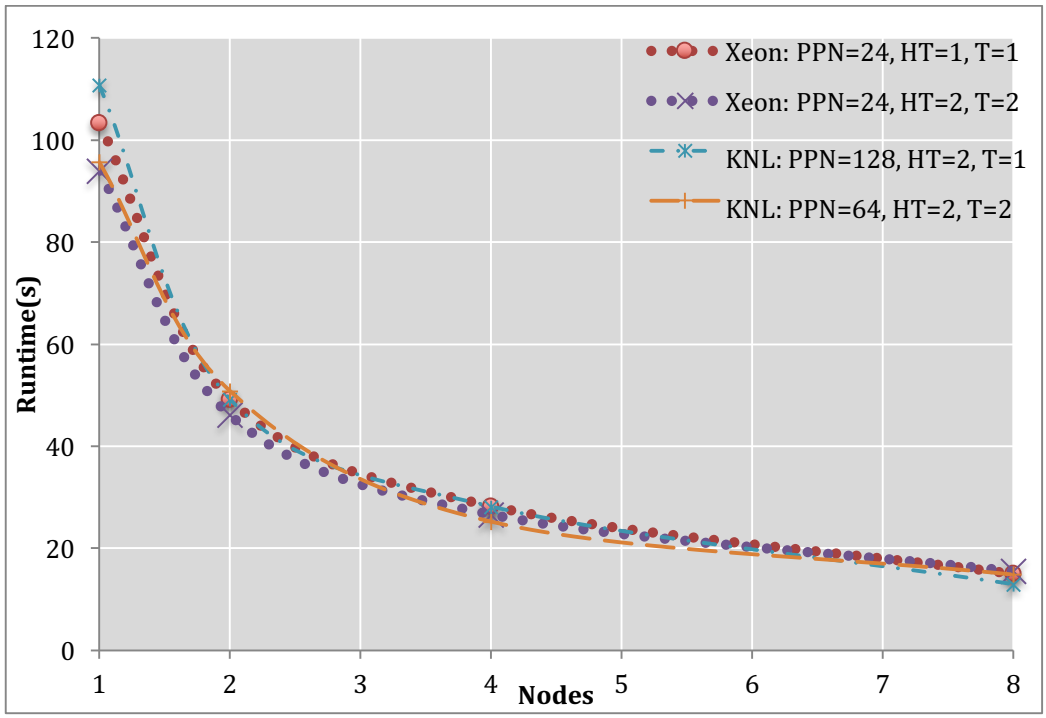

Figure 1 - Best performance configurations comparison between KNL and Xeon.

OpenSBLI reports very similar performance on KNL and Xeon nodes. For less than 4 nodes, ARCHER Xeon seems to provide better performance (PPN=25,HT=2,T=1). However it is KNL who gives better results for than 4 nodes.

Figure 2 (note the log scale) shows the performance comparison of running OpenSBLI on Xeon and KNL fully populated nodes using 1, 2 and 4 hyperthreads and only one OpenMP thread per process. These results indicate that OpenSBLI running on 4 hyperthreads on KNL is not a good option and Xeon and KNL perform better with 1 hyperthread if no other threads are used.

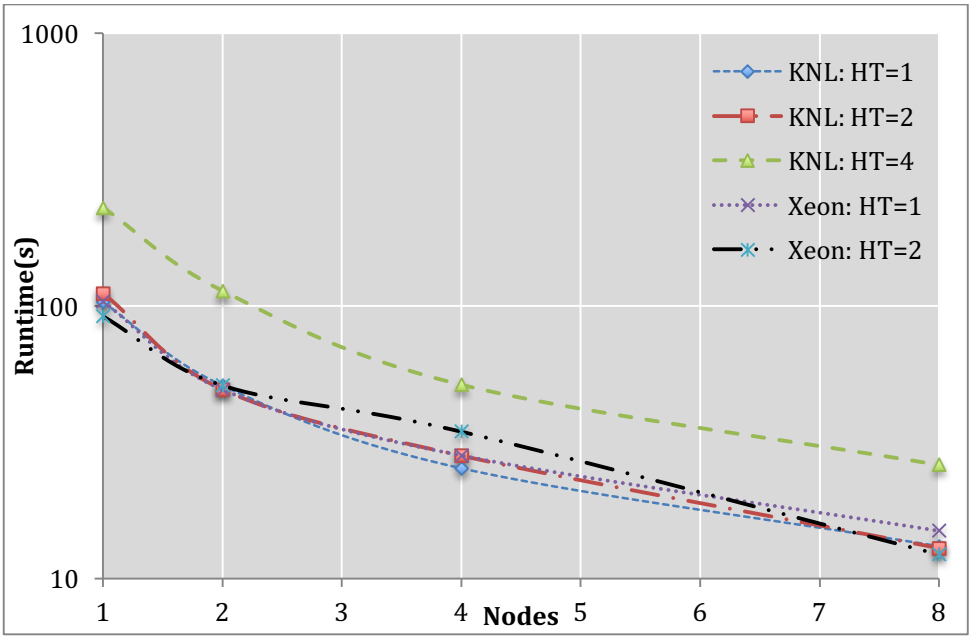

Figure 2 – OpenSBLI performance on fully populated nodes with one thread per process.

In Figure 3 (note the log scale) we have plotted the best performance of OpenSBLI running on 2 and 4 threads (2 and 4 hyperthreads) per process. As seen in Figure 2, Xeon and KNL show similar performance however once again KNL does not perform well on 4 threads per process.

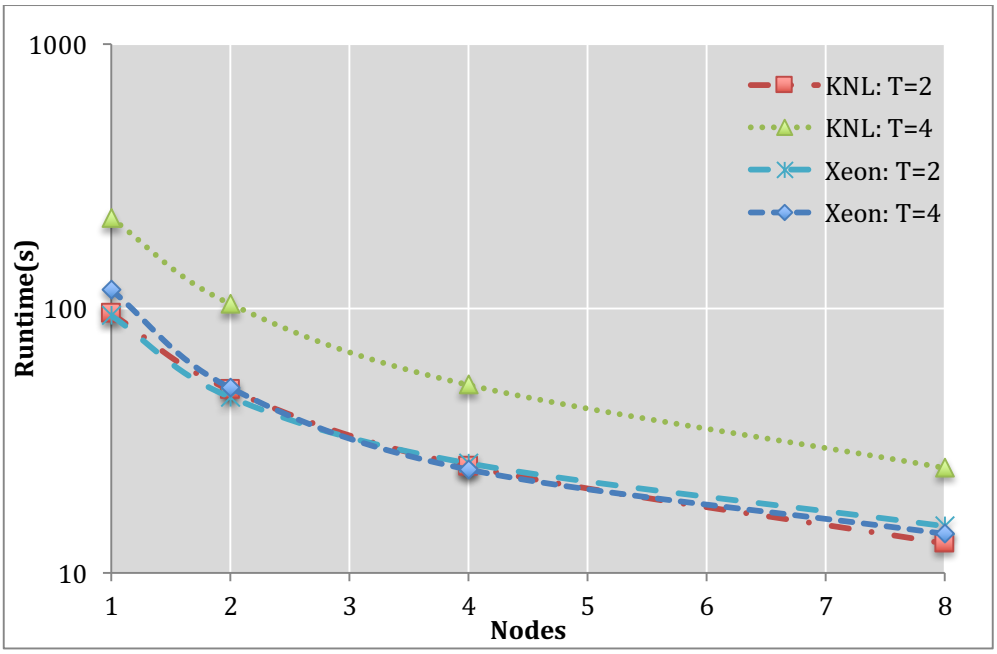

Figure 3 - OpenSBLI best performance with 2 and 4 threads per process.

3. Summary and Conclusions

- In general, both Xeon and KNL display very much alike performance and although the best running configuration changes from single node to multiple nodes, it seems that both systems perform best using 2 hyperthreads and 2 threads per process.
- KNL performance is normally better with 2 hyperthreads.
- The best performance for Xeon varies between 1 or 2 hyperthreads

4. Full Performance Results

KNL

Cray compiler

Processes	nodes	T	HT	runtime(AVG)
32	1	2	1	105.95
32	1	4	2	104.806
32	1	8	4	177.098
64	1	1	1	102.89
64	1	2	2	95.7017
64	1	4	4	233.331
128	1	1	2	110.745
128	1	2	4	219.299
256	1	1	4	228.598
128	2	1	1	50.9131
128	2	2	2	50.8485
128	2	4	4	107.06
256	2	1	2	49.1892
256	2	2	4	103.896
512	2	1	4	113.292
1024	4	1	4	51.3059
256	4	1	1	25.3055
256	4	2	2	25.2389
256	4	4	4	52.5429
512	4	1	2	28.1337
512	4	2	4	53.4812
1536	6	1	4	39.0665
384	6	1	1	23.3561
384	6	2	2	17.1667
384	6	4	4	34.9631
768	6	1	2	18.7249
768	6	2	4	34.8913
1024	8	1	2	12.8888
1024	8	2	4	24.8943
2048	8	1	4	26.0986
512	8	1	1	13.094
512	8	2	2	14.8553
512	8	4	4	27.6953

Intel compiler

Processes	nodes	T	HT	runtime(AVG)
64	1	1	1	173.159
64	1	2	2	145.911
64	1	4	4	253.308
128	1	1	2	143.358
128	1	2	4	241.347
256	1	1	4	277.226
128	2	1	1	84.9706
128	2	2	2	70.4527
128	2	4	4	116.614

256	2	1	2	85.7368
256	2	2	4	124.671
512	2	1	4	130.814
256	4	1	1	43.5282
256	4	2	2	41.3605
256	4	4	4	58.8586
512	4	1	2	35.8643
512	4	2	4	62.1206
1024	4	1	4	57.5134
384	6	1	1	28.6502
384	6	2	2	23.6951
384	6	4	4	38.0835
768	6	1	2	32.6641
768	6	2	4	44.8569
1536	6	1	4	36.4924
512	8	1	1	22.004
512	8	2	2	18.4118
512	8	4	4	29.8768
1024	8	1	2	18.2406
1024	8	2	4	28.0735
2048	8	1	4	31.6593

Xeon

Cray compiler

nodes	Processes	PPN	HT	T	runtime(AVG)
1	1	1	1	24	193.304
1	2	2	1	12	124.78
1	6	6	1	4	117.1
1	12	12	1	2	148.396
1	24	24	1	1	103.411
2	2	1	1	24	103.571
2	4	2	1	12	62.2173
2	12	6	1	4	80.0678
2	24	12	1	2	52.7123
2	48	24	1	1	49.212
4	4	1	1	24	52.4502
4	8	2	1	12	33.1208
4	24	6	1	4	30.3452
4	48	12	1	2	25.8931
4	96	24	1	1	28.2506
6	6	1	1	24	43.9994
6	12	2	1	12	32.9676
6	36	6	1	4	34.0257
6	72	12	1	2	19.8361
6	144	24	1	1	41.3774
8	8	1	1	24	26.9665
8	16	2	1	12	19.554
8	48	6	1	4	14.0358
8	96	12	1	2	14.9586
8	192	24	1	1	14.0043

1	2	2	2	24	120.545
1	4	4	2	12	122.298
1	12	12	2	4	180.997
1	24	24	2	2	94.0206
1	48	48	2	1	91.3788
2	4	2	2	24	72.9312
2	8	4	2	12	69.7387
2	24	12	2	4	49.9517
2	48	24	2	2	46.091
2	96	48	2	1	50.9915
4	8	2	2	24	37.7122
4	16	4	2	12	31.7479
4	48	12	2	4	24.5494
4	96	24	2	2	26.5981
4	192	48	2	1	34.5488
6	12	2	2	24	39.1879
6	24	4	2	12	22.2502
6	72	12	2	4	20.3102
6	144	24	2	2	45.1532
6	288	48	2	1	21.9102
8	16	2	2	24	19.7887
8	32	4	2	12	14.8319
8	96	12	2	4	14.2065
8	192	24	2	2	15.4624
8	384	48	2	1	12.2562

Intel compiler

nodes	Processes	PPN	HT	T	runtime(AVG)
1	1	1	1	24	2605.514547
1	2	2	1	12	1276.99
1	6	6	1	4	599.316
1	12	12	1	2	287.909
1	24	24	1	1	150.781
2	2	1	1	24	1276.51
2	4	2	1	12	640.366
2	12	6	1	4	259.247
2	24	12	1	2	138.916
2	48	24	1	1	73.3449
4	4	1	1	24	644.319
4	8	2	1	12	323.985
4	24	6	1	4	127.031
4	48	12	1	2	68.1564
4	96	24	1	1	40.7647
6	6	1	1	24	564.892
6	12	2	1	12	243.104
6	36	6	1	4	86.3182
6	72	12	1	2	51.604
6	144	24	1	1	28.2363
8	8	1	1	24	322.684
8	16	2	1	12	163.218

8	48	6	1	4	60.4122
8	96	12	1	2	37.4354
8	192	24	1	1	19.8343
1	2	2	2	24	1275.76
1	4	4	2	12	663.894
1	12	12	2	4	287.135
1	24	24	2	2	151.557
1	48	48	2	1	127.467
2	4	2	2	24	641.401
2	8	4	2	12	333.453
2	24	12	2	4	138.57
2	48	24	2	2	73.9371
2	96	48	2	1	67.5597
4	8	2	2	24	322.669
4	16	4	2	12	170.035
4	48	12	2	4	68.0828
4	96	24	2	2	41.1308
4	192	48	2	1	33.4978
6	12	2	2	24	243.215
6	24	4	2	12	122.321
6	72	12	2	4	51.4803
6	144	24	2	2	28.4909
6	288	48	2	1	23.968
8	16	2	2	24	163.223
8	32	4	2	12	83.7793
8	96	12	2	4	37.4409
8	192	24	2	2	19.9833
8	384	48	2	1	16.5254