SAND2017-5701C

Exceptional service in the national interest

Sandia National Laboratories



Next Generation Science Applications for the Next Generation of Supercomputing

C.T. Vaughan, S.D. Hammond, D.C. Dinge, P.T. Lin, D.M. Pase,
C.R. Trott, J. Cook, C. Hughes, and R.J. Hoekstra

Sandia National Laboratories

CUG 2017, Redmond, WA

CCR Center for Computing Research

U.S. DEPARTMENT OF ENERGY | NNSA National Nuclear Security Administration

Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Trinity

Sandia National Laboratories

- Cray XC40
- Total of about 19000 nodes
  - About half are Intel Haswell with 2 processors per node and 16 cores per processor running at 2.3 GHz and 128 GB memory per node
  - About half are 68 core Intel Knights Landing processors – most are currently in a separate test system and the two halves will be merged this summer
  - Cray Aries Dragonfly interconnect

SPARC

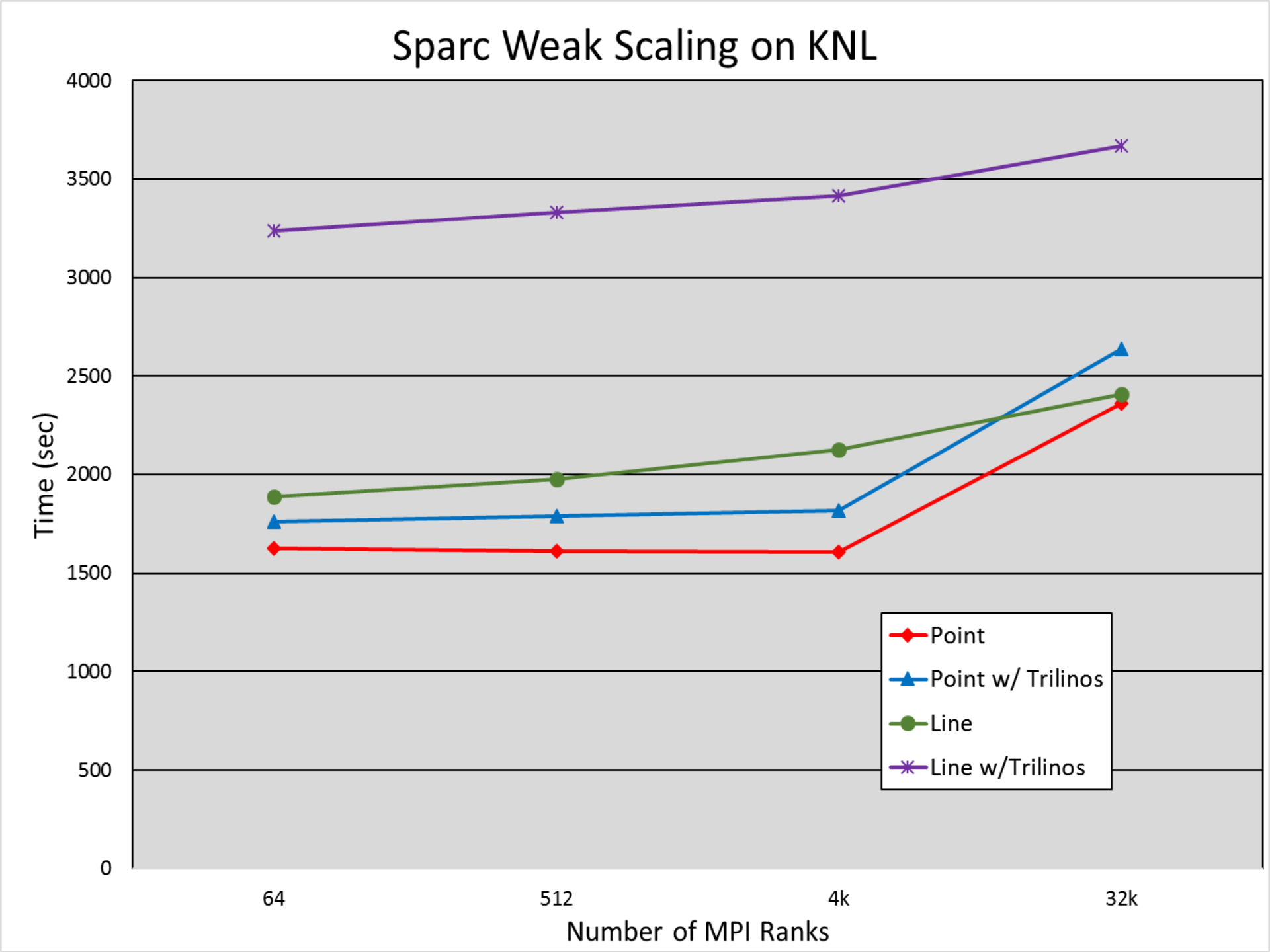
Sandia National Laboratories

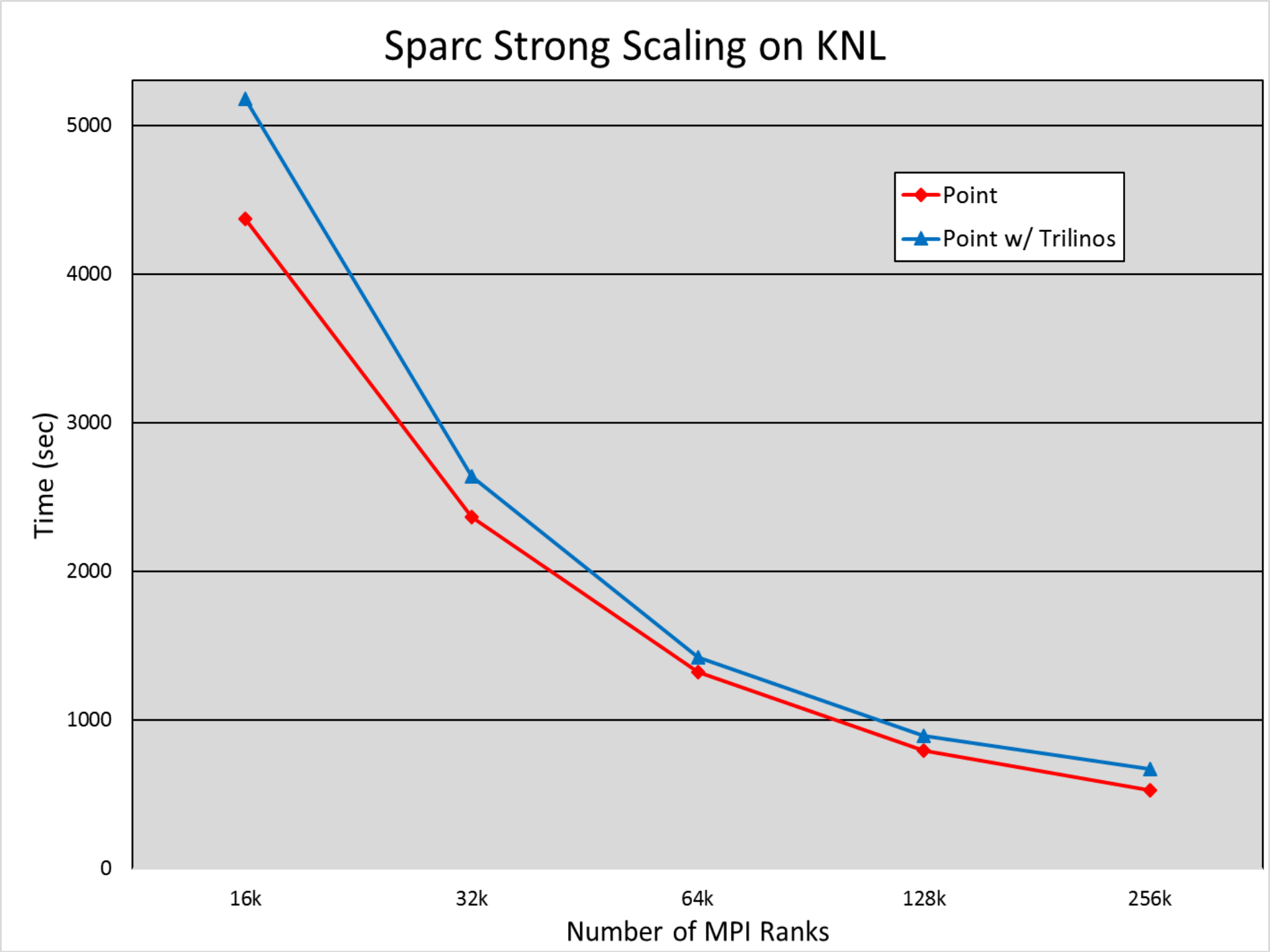
- Sandia's Parallel Aerosciences Research code
  - Computational Fluid Dynamics (CFD) code
- Written in C++
- Written as a parallel 3D code
- Uses MPI with OpenMP
  - Uses Sandia's Kokkos C++ parallel pattern abstractions
- Uses cell-centered finite volume methods for CFD and Galerkin finite element methods for ablation and thermal analysis
- Use packages from the Trilinos framework
  - Has options to use either a native solver or a Trilinos solver

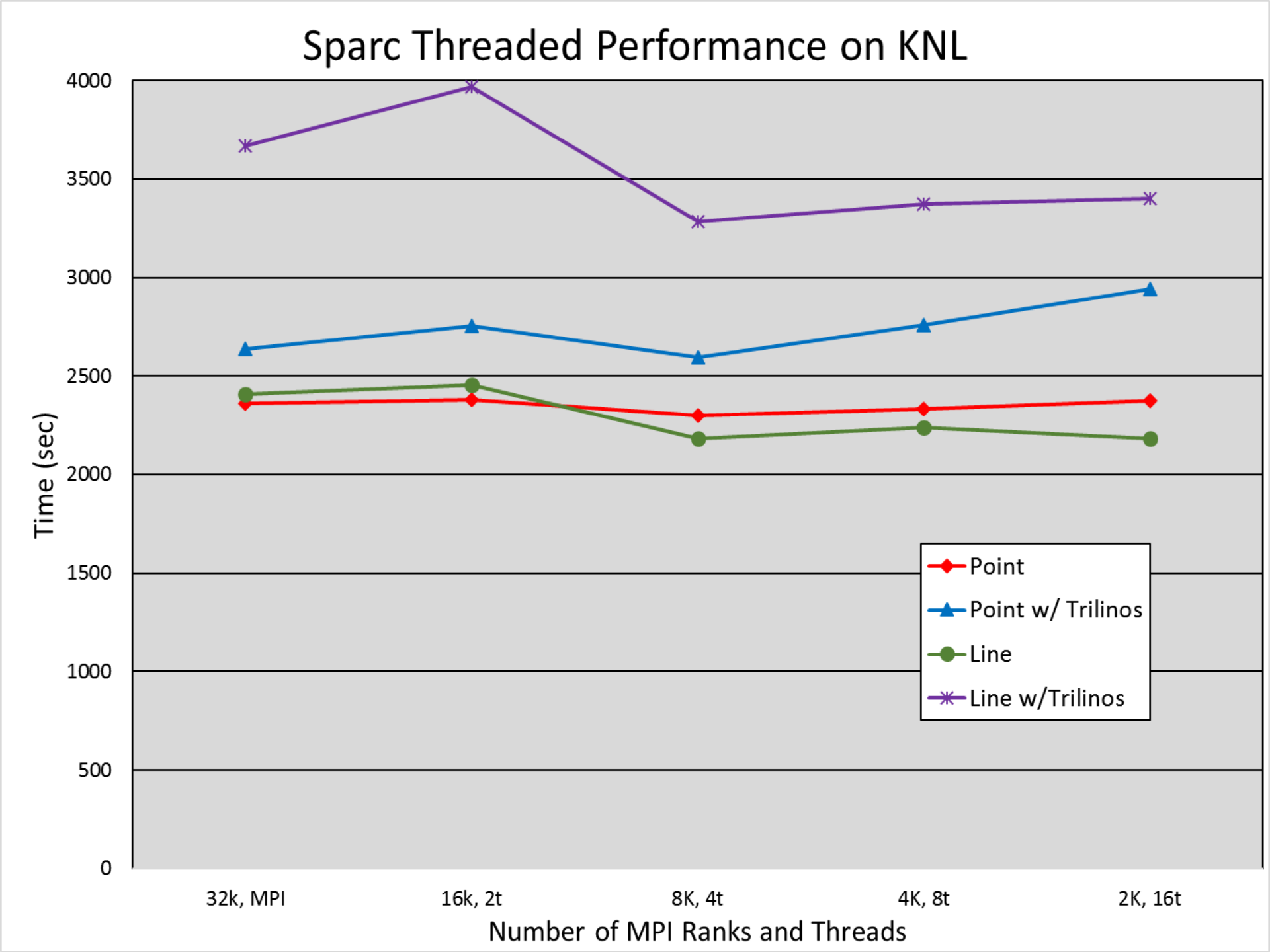
Benchmarking Sparc on Trinity

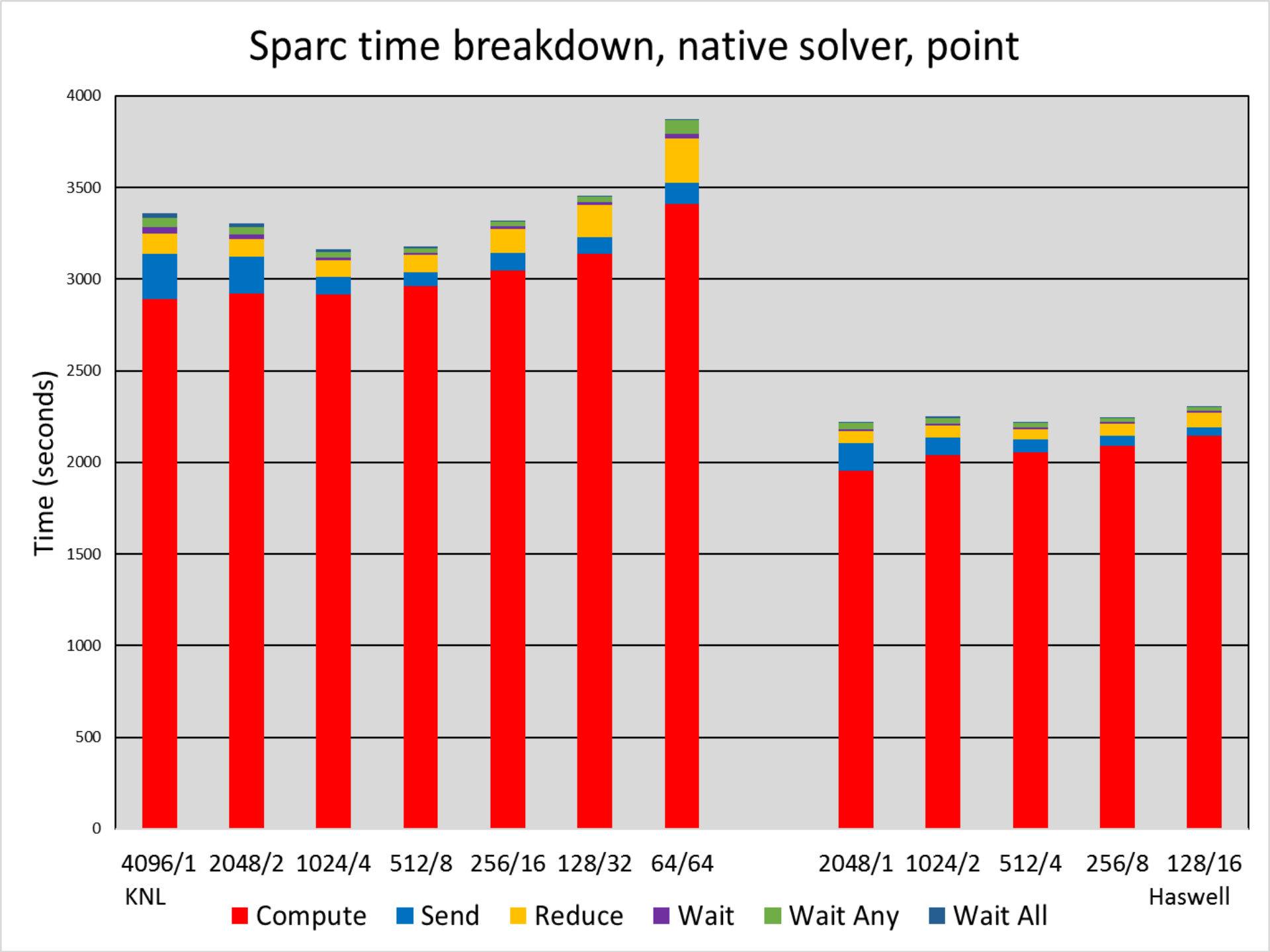
Sandia National Laboratories

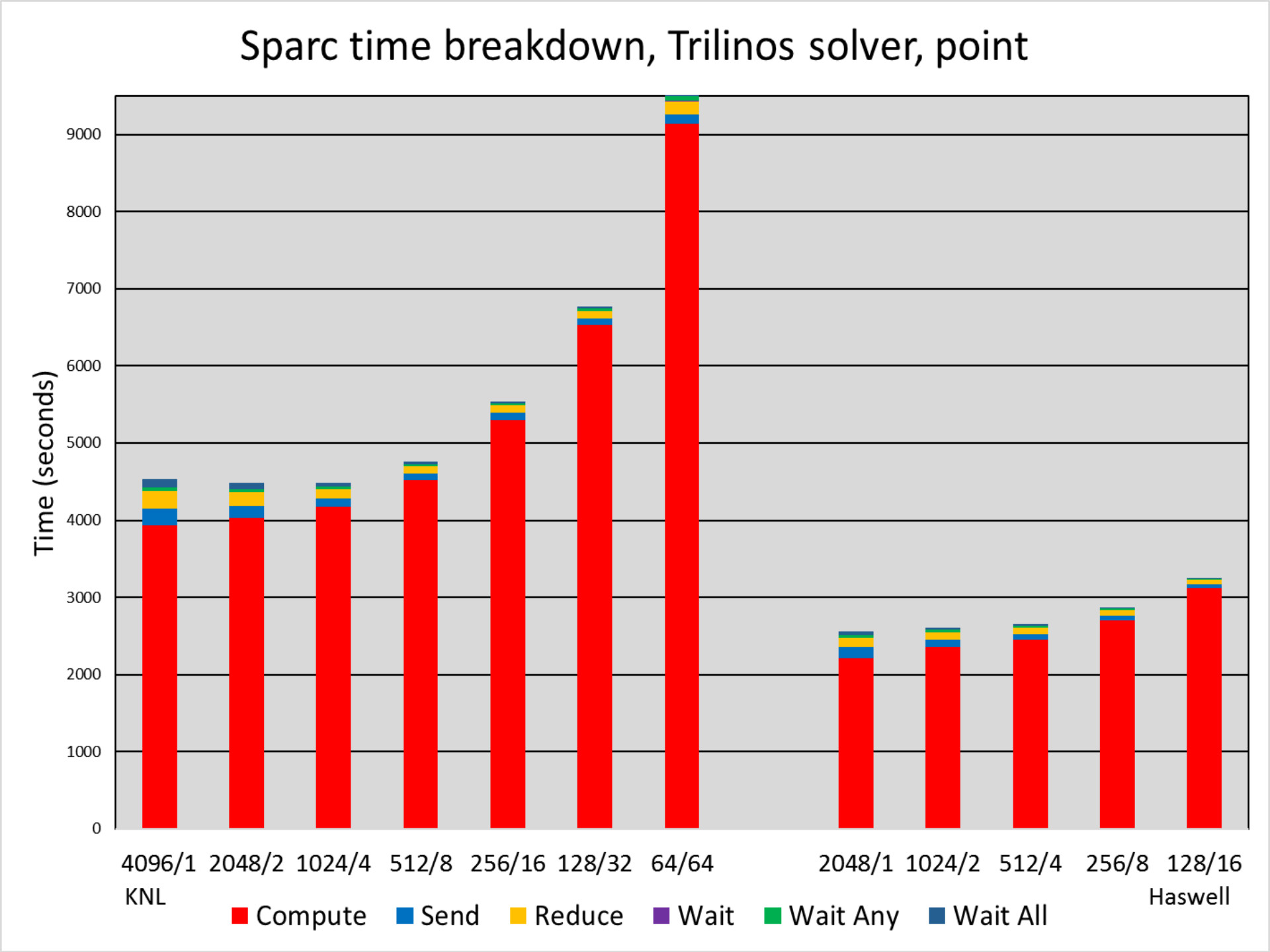
- For KNL, we ran in Quad-Flat mode
  - MCDRAM available as allocatable space
  - Initial experience is that Quad-Cache mode shows little performance gain
- Runs on KNL used 64 cores per node with no hyperthreading
  - Our experiments shows a small performance gain by using two hyperthreads and a small performance loss from using four hyperthreads
- Compiled with Intel 2017 Update 2 compiler


Sparc Weak Scaling on KNL
4000
3500
3000
2500
2000
1500
1000
500
0
Time (sec)
64
512
4k
32k
Number of MPI Ranks
Point
Point w/ Trilinos
Line
Line w/Trilinos


Sparc Strong Scaling on KNL
5000
4000
3000
2000
1000
0
Time (sec)
Point
Point w/ Trilinos
16k
32k
64k
128k
256k
Number of MPI Ranks


Sparc Threaded Performance on KNL
4000
3500
3000
2500
2000
1500
1000
500
0
Time (sec)
Point
Point w/ Trilinos
Line
Line w/Trilinos
32k, MPI
16k, 2t
8K, 4t
4K, 8t
2K, 16t
Number of MPI Ranks and Threads

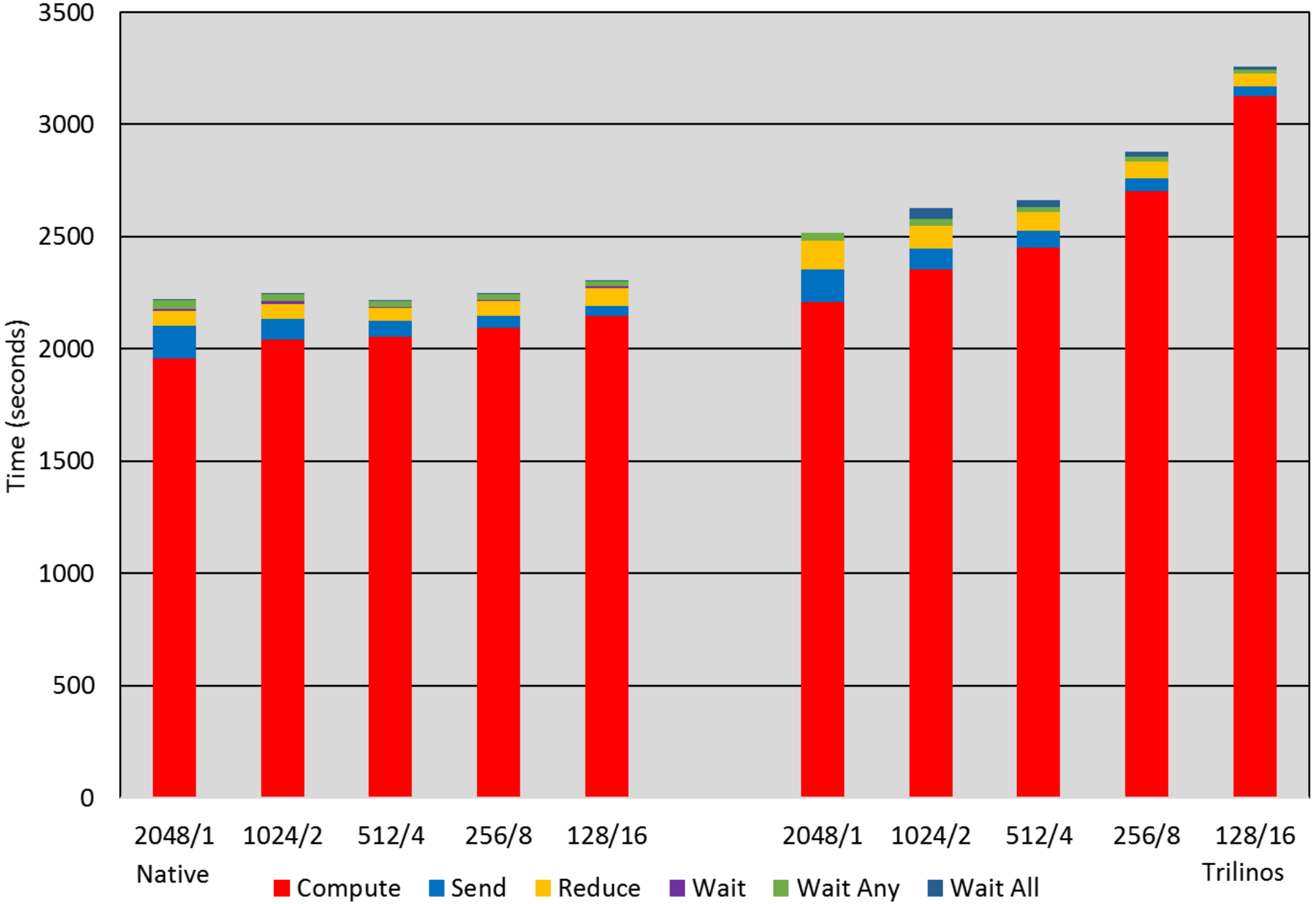

Sparc time breakdown, native solver, point
Time (seconds)
4000
3500
3000
2500
2000
1500
1000
500
0
4096/1
2048/2
1024/4
512/8
256/16
128/32
64/64
2048/1
1024/2
512/4
256/8
128/16
KNL
Haswell
Compute
Send
Reduce
Wait
Wait Any
Wait All


Sparc time breakdown, Trilinos solver, point
Time (seconds)
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
4096/1
2048/2
1024/4
512/8
256/16
128/32
64/64
2048/1
1024/2
512/4
256/8
128/16
KNL
Haswell
Compute
Send
Reduce
Wait
Wait Any
Wait All

Comments about threading on KNL

Sandia National Laboratories

- Number of messages exchanged decreases from 130000 to 103000 per MPI rank when going from MPI only to 64 threads
  - So the total number of messages decreases substantially
- The bytes exchanged grows from 1.2 E10 to 1.46 E11 bytes per MPI rank when going from MPI only to 64 threads
  - The total message traffic drops from 4.5 E13 to 9.33 E12 bytes
  - The average message size goes from 82 KB to 1.3 MB
- For the native solver, the number of reduces stays constant at 47500 calls and 3.6 MB
- For the Trilinos solver, the number of reduces stays constant at 30000 calls and 2.4 MB


Sparc time breakdown, point
Time (seconds)
3500
3000
2500
2000
1500
1000
500
0
2048/1
1024/2
512/4
256/8
128/16
2048/1
1024/2
512/4
256/8
128/16
Native
Trilinos
Compute
Send
Reduce
Wait
Wait Any
Wait All

KNL node modes

Sandia National Laboratories

- Previous results were run in Quad Flat mode
- We also ran in Quad Cache mode
- For problems that used a significant portion of memory, Quad Cache mode was less than 2% faster than Quad Flat
- Likewise, for problems that used very little memory, Quad Cache mode was less than 2% faster than Quad Flat
- For problems that used a reasonable amount of memory, Quad Cache mode was up to 50% faster

Conclusions

Sandia National Laboratories

- Sparc scales fairly well both in a strong mode and a weak mode on both the Haswell and KNL nodes
- The native solver does well as MPI ranks are exchanged for threads
- The Trilinos solver does not scale well as MPI ranks are exchanged for threads
- On the KNL nodes, Sparc does not make good use of the wide vectors – less than 5% of the code vectorizes

Sandia
National
Laboratories

Exceptional service in the national interest