

Running HARMONIE on Xeon Phi Coprocessors

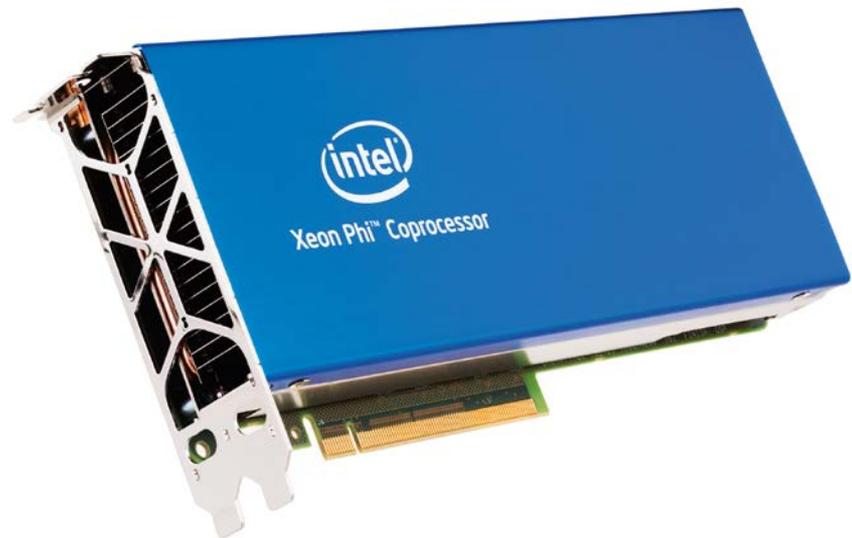
Enda O'Brien

Irish Centre for High-End Computing



Disclosure

ICHEC is grateful to Intel for supporting the porting & optimization of several applications, including HARMONIE, to Xeon Phi coprocessors.



Motivating Questions

Hypothetical:

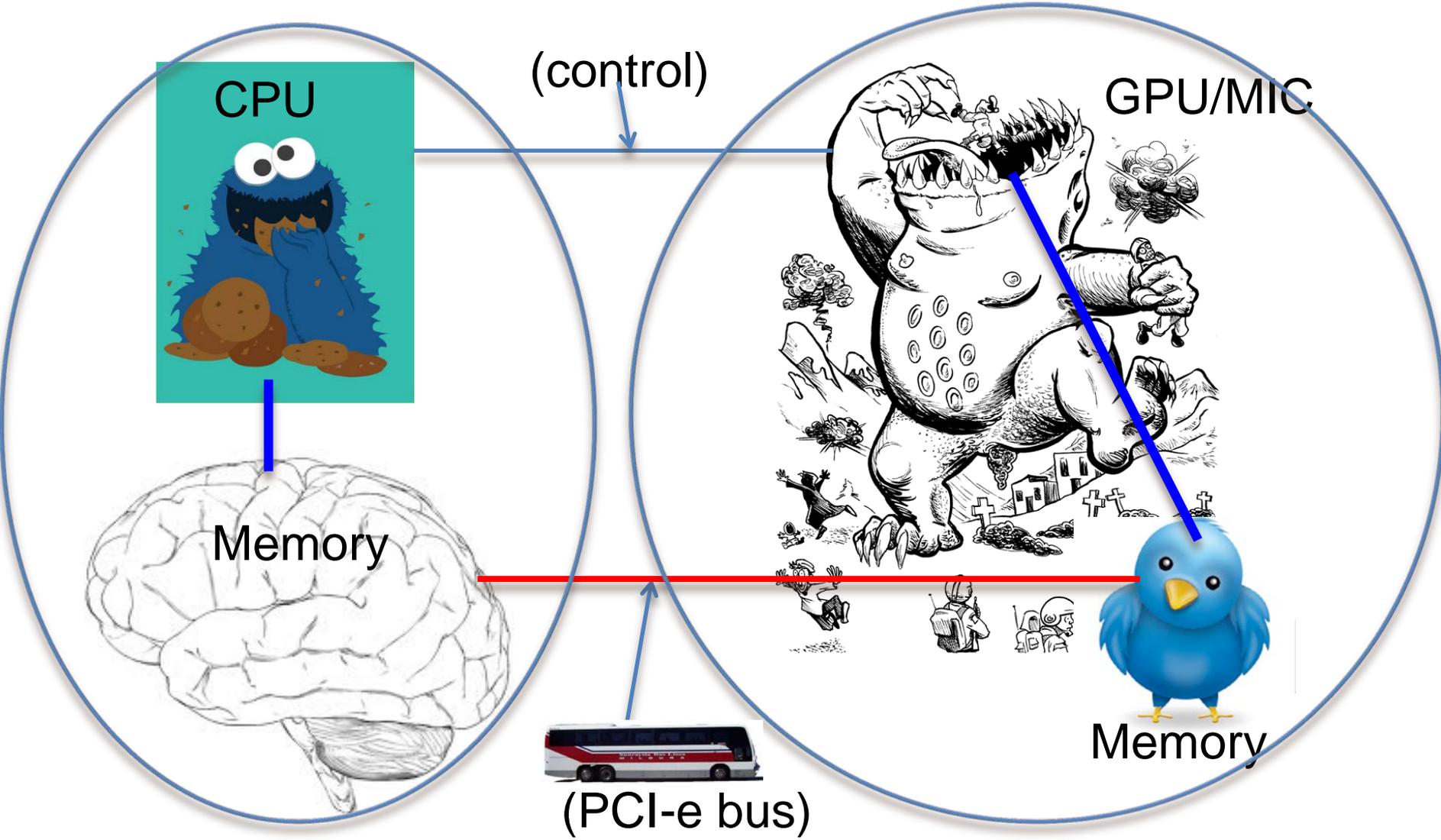
- How much (human) effort is worth investing to obtain a **10 x** performance speedup, *if available*, from hardware accelerators?
- How about **2 x** speedup?
- Or **20%** speedup?

Practical:

- Which provides more value: an extra compute node, or an accelerator?

Host Node

GPU/MIC Device



Ways to use Accelerators

Xeon Phi

GPU

Offload mode:

- Uses directives in source
- Many programming constraints
- All processes run on hosts, with parallel sections offloaded to accelerator



(possible, but hard)



(possible, but hard)

Native mode:

- no source changes required
- Cluster of MIC nodes



(easy)



Symmetric mode:

- No source changes required
- MICs & hosts each a separate node in a cluster



(should be easy,
but isn't)



Offload of Main OpenMP loop Fails

```
cpg.F90(570): error #8545:
```

```
A variable used in an OFFLOAD region must not be of  
derived type with pointer or allocatable components.
```

```
[YDSL]
```

```
!dir$ omp offload target(mic)
```

```
in(ydsl, CDCONF, LDRETCFOU, LDWRTCFOU0, LDCPG_SPLIT)
```

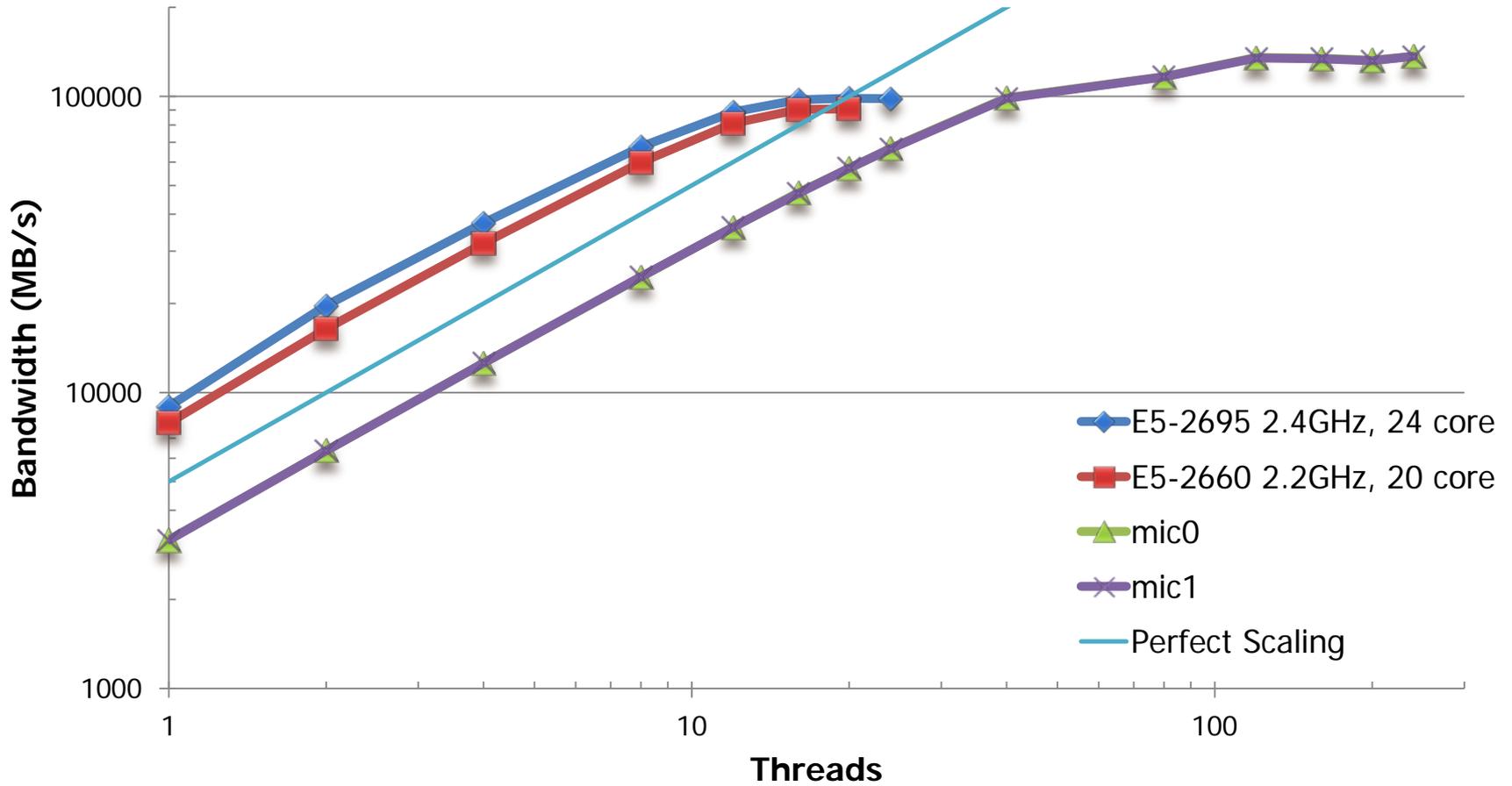
That is a show-stopper.

Xeon vs. Xeon Phi: Vital Stats

	E5-2660 2.2 GHz	Xeon Phi 5110P
Cores (pre node)	20	61
Threads (per node)	40	240
Clock Freq.	2.2 GHz	1.053 GHz
Memory	64 GB/node	8 GB x 2 cards = 16 GB
Max. Stream Triad	91 GB/s	137 GB/s
Linpak	316 Gflop/s (max) 288 Gflop/s (16 GB)	720 Gflop/s
IMB PingPong latency	< 2 usec	5 - 12 usec
IMB PingPong B/w	> 4 GB/s	0.22 - 4 GB/s

*Phi performance is contingent on using **all** cores or threads!*

Stream Triad Performance on Xeon systems



IMB Ping-Pong 0-byte Message Latency (usec)

t[usec]	host0	host0-mic0	host0-mic1	host1	host1-mic0	host1-mic1
host0	0.36	5.24	6.40	1.96	6.43	7.05
host0-mic0	5.24	2.28	9.08	6.43	8.96	9.71
host0-mic1	6.40	9.08	2.37	7.05	9.71	10.99

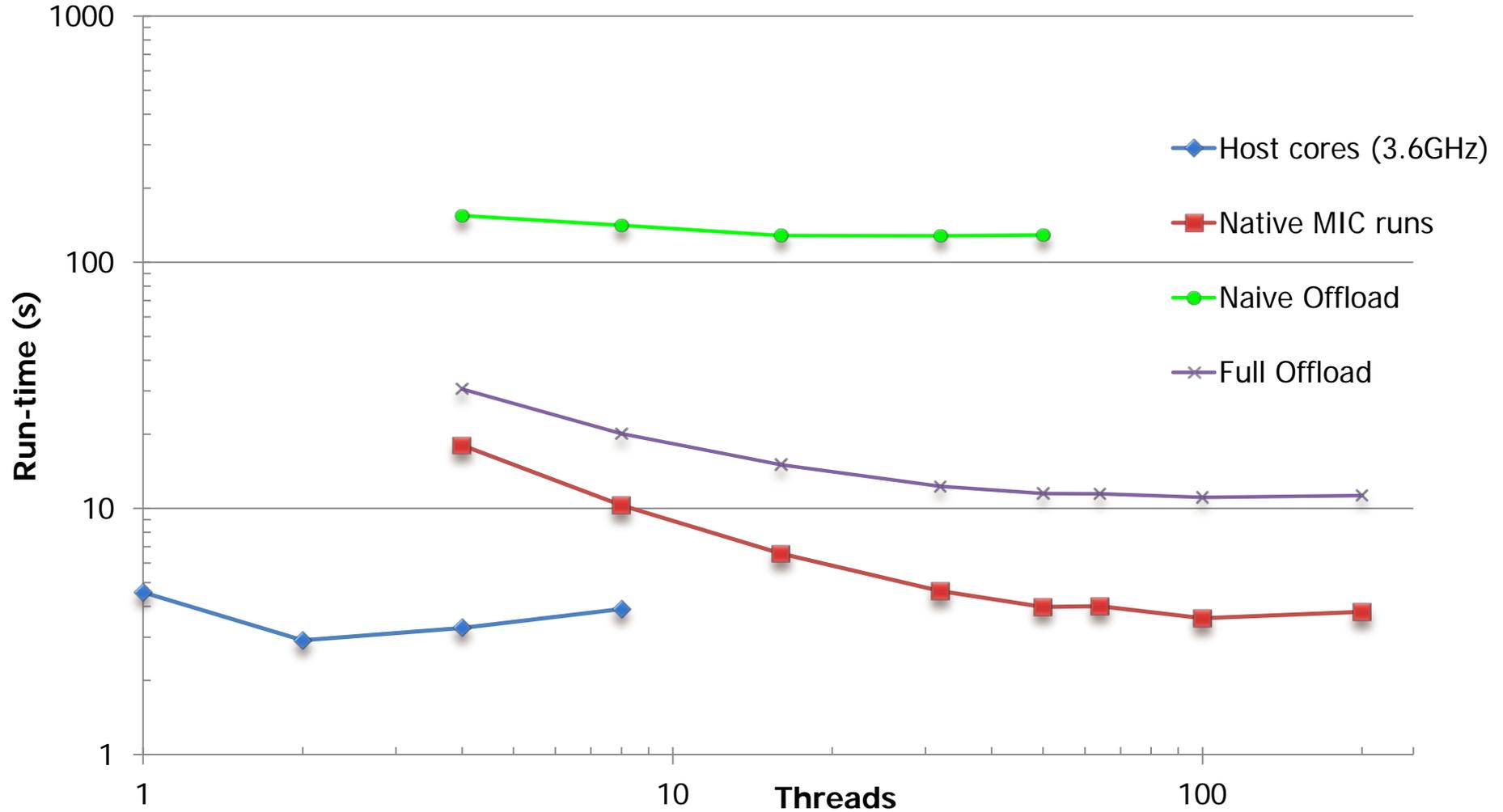
IMB Ping-Pong 4-MB Message Bandwidth (MB/s)

MB/s	host0	host0-mic0	host0-mic1	host1	host1-mic0	host1-mic1
host0	4067	4923	5193	5870	4156	505
host0-mic0	4923	2020	1269	4156	3539	494
host0-mic1	5193	1269	1951	505	494	266

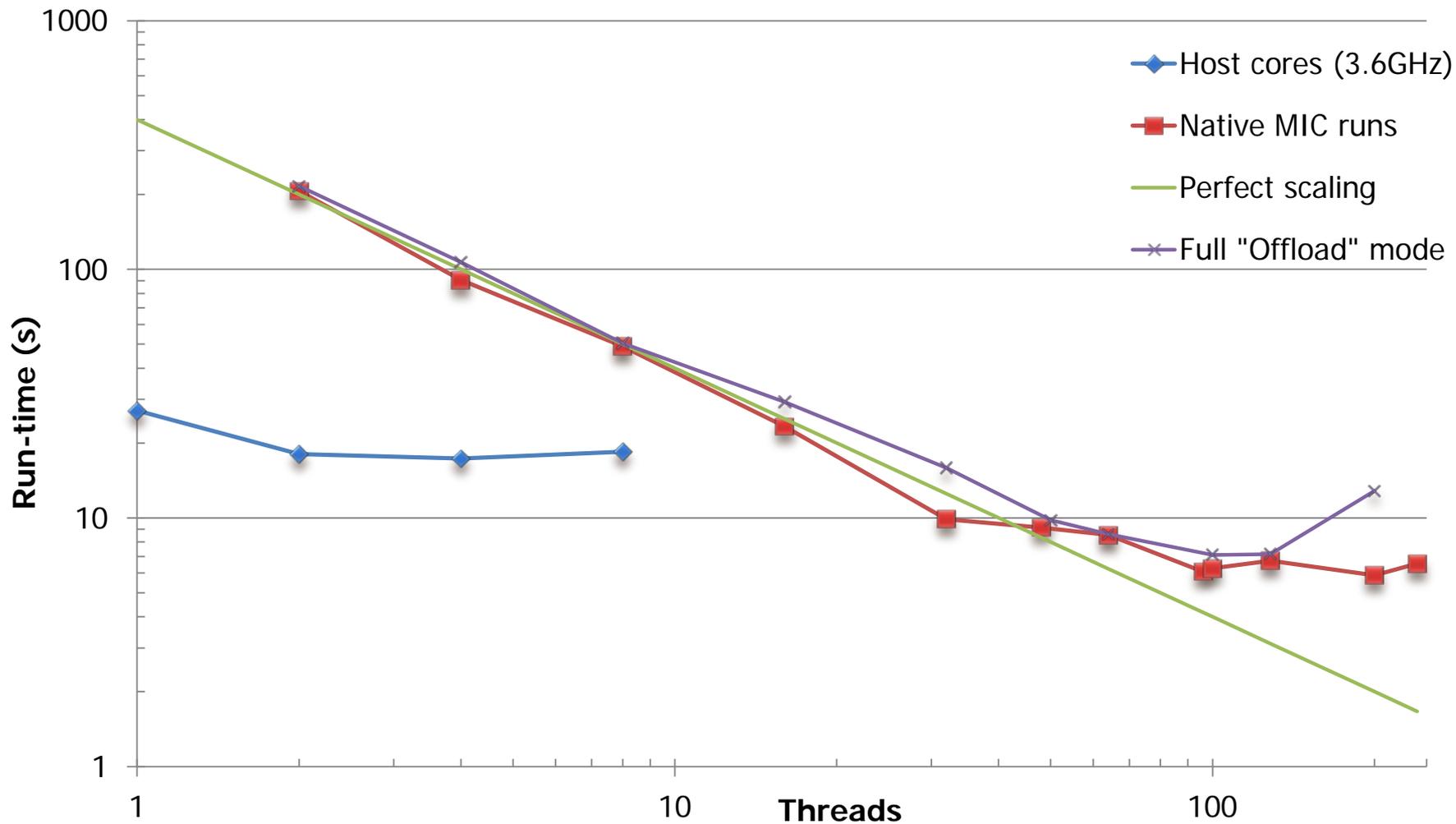
Test Code (Fortran)

```
!$OMP PARALLEL DO PRIVATE(i,j,k)
  do k=2,nz-1
    do j=2,ny-1
      do i=2,nx-1
        arr_out(i,j,k) = wgght1*arr_in(i,j,k) + wgght2*(
&          arr_in(i-1,j,k) + arr_in(i+1,j,k) +
&          arr_in(i,j-1,k) + arr_in(i,j+1,k) +
&          arr_in(i,j,k-1) + arr_in(i,j,k+1) )
      enddo
    enddo
  enddo
!$OMP END PARALLEL DO
```

"Stencil test" OpenMP Performance on Phi, 1GB case



"Stencil test" OpenMP Performance on Phi, 6GB case



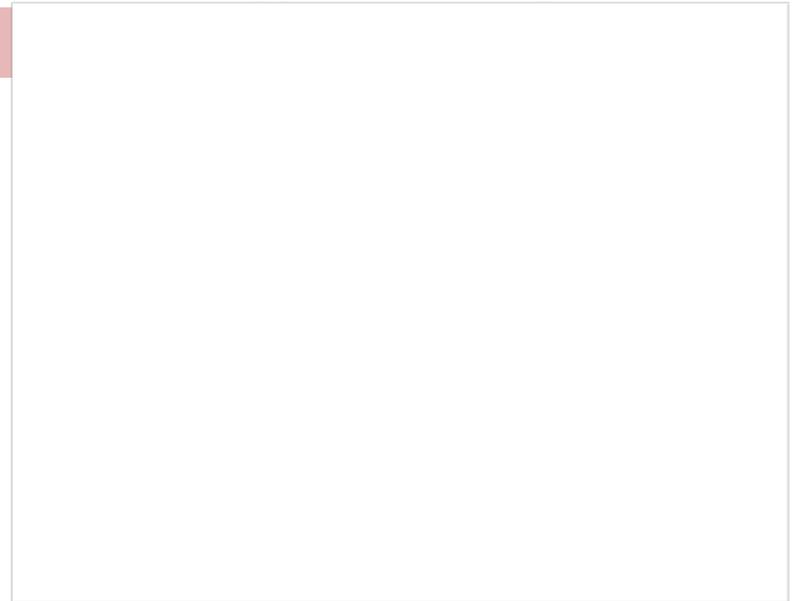
Best case: 3x speedup on Phi

HARMONIE on Xeon Phi

- HARMONIE builds ~cleanly with “-openmp -mmic”, runs natively on Phi
 - No source code changes (in principle)
 - Must use Intel compilers, Intel MPI
 - Must re-build `zlib`, `hdf5`, `netcdf`, & `grib_api` with “-mmic”
- Builds completed:
 - HARMONIE cycle37h1.1 and cycle38h1.1;
 - MPI-only and MPI/OpenMP
 - Host and Phi.
- Main executable from “Phi” build copied to “standard” installation
 - for use in “Forecast” phase only.
- Test case, IRELAND55: `300 x 300 x 65`-point domain, 5.5 km resolution:
Memory needed: ~20GB minimum (depends on run-time config.)

Host: 20 physical cores; 40 logical cores (with Hyperthreading)

	No HyperThreads		
<i>MPI Processes</i>	<i>OMP_NUM_THREADS</i>	<i>Total Threads</i>	<i>Forecast Time (s)</i>
2	10	20	1570
5	4	20	1445
10	2	20	1384
20	1	20	769
40			



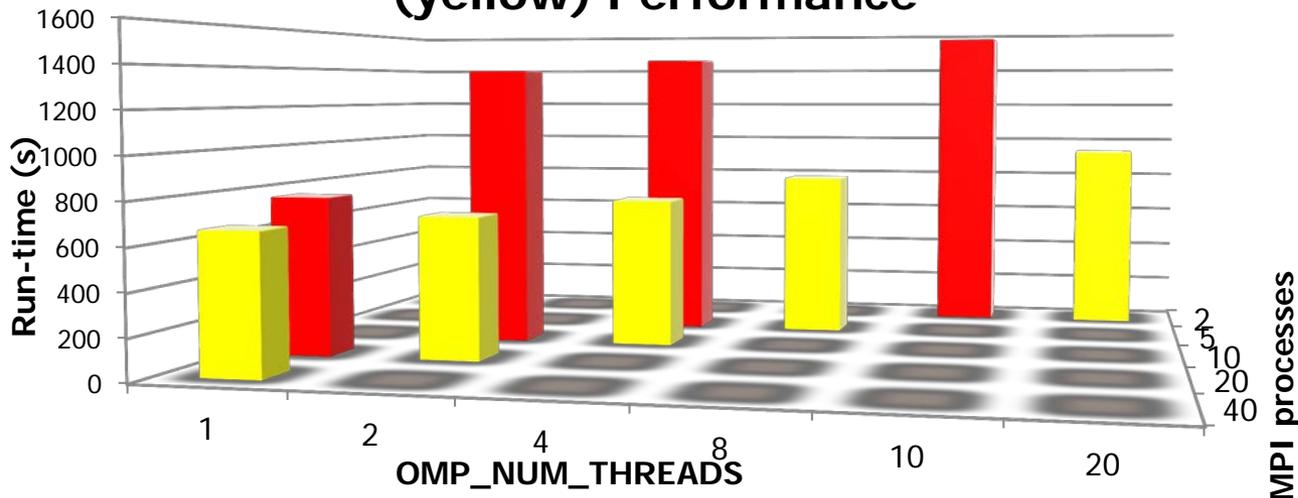
Host: 20 physical cores; 40 logical cores (with Hyperthreading)

<i>MPI Processes</i>	No HyperThreads			Using HyperThreads		
	<i>OMP_NUM_THREADS</i>	<i>Total Threads</i>	<i>Forecast Time (s)</i>	<i>OMP_NUM_THREADS</i>	<i>Total Threads</i>	<i>Forecast Time (s)</i>
2	10	20	1570	20	40	940
5	4	20	1445	8	40	814
10	2	20	1384	4	40	727
20	1	20	769	2	40	687
40				1	40	668

Host: 20 physical cores; 40 logical cores (with Hyperthreading)

<i>MPI Processes</i>	No HyperThreads			Using HyperThreads		
	<i>OMP_NUM_THREADS</i>	<i>Total Threads</i>	<i>Forecast Time (s)</i>	<i>OMP_NUM_THREADS</i>	<i>Total Threads</i>	<i>Forecast Time (s)</i>
2	10	20	1570	20	40	940
5	4	20	1445	8	40	814
10	2	20	1384	4	40	727
20	1	20	769	2	40	687
40				1	40	668

20-Thread (red) and 40 Hyperthread (yellow) Performance

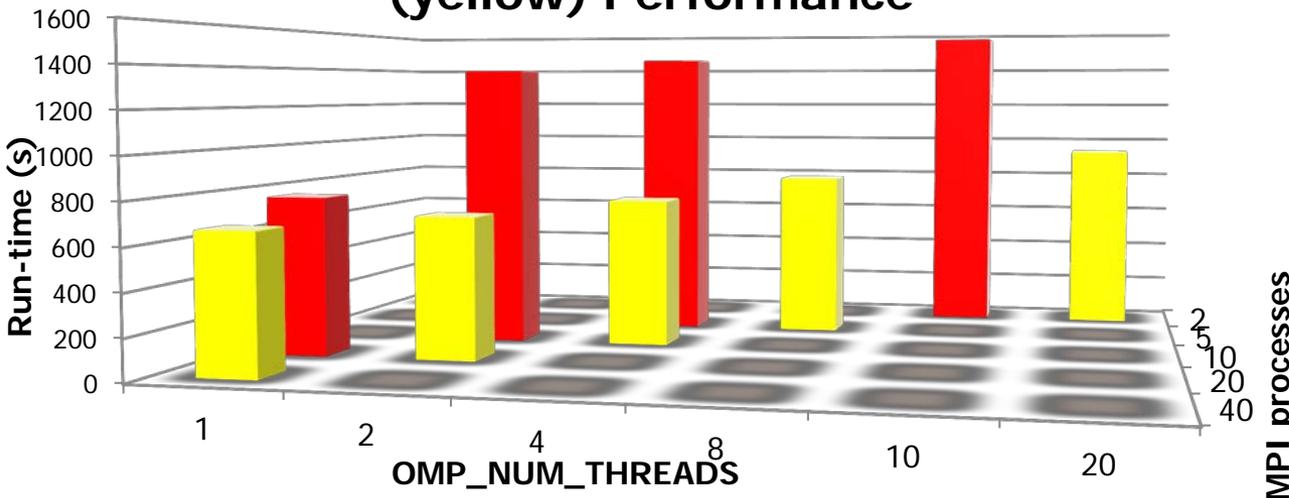


MPI vs. OpenMP on Host nodes

Host: 20 physical cores; 40 logical cores (with Hyperthreading)

<i>MPI Processes</i>	No HyperThreads			Using HyperThreads		
	<i>OMP_NUM_THREADS</i>	<i>Total Threads</i>	<i>Forecast Time (s)</i>	<i>OMP_NUM_THREADS</i>	<i>Total Threads</i>	<i>Forecast Time (s)</i>
2	10	20	1570	20	40	940
5	4	20	1445	8	40	814
10	2	20	1384	4	40	727
20	1	20	769	2	40	687
40				1	40	668

20-Thread (red) and 40 Hyperthread (yellow) Performance



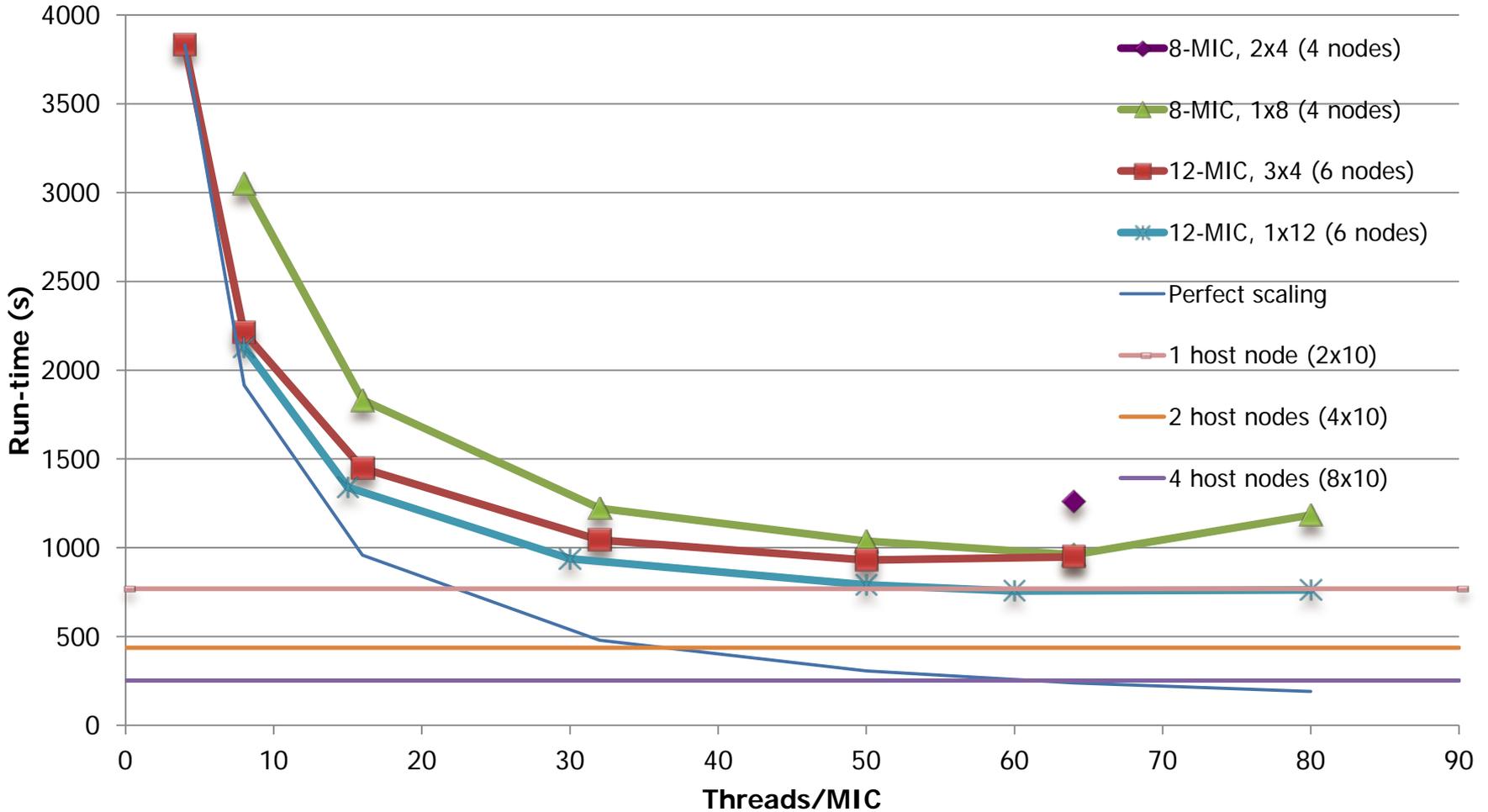
On Host: Use MPI in preference to OpenMP - (after using OpenMP to soak up the "HyperThreads")

MPI vs. OpenMP on MIC cards

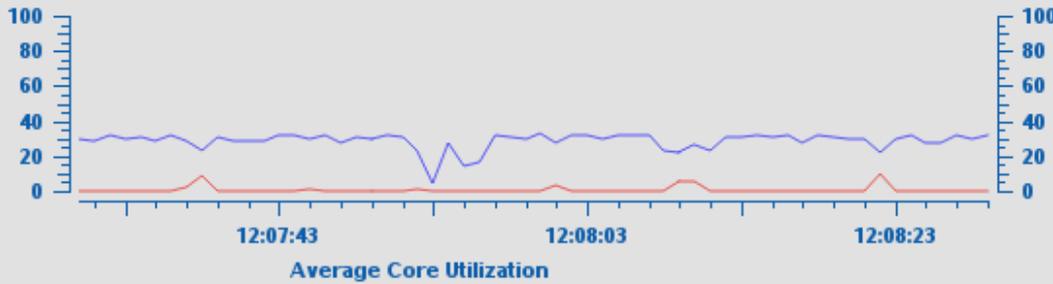
MPI Only	192 MPI tasks (12 MICs, 16 MPI tasks /MIC)	3779s
MPI/OpenMP	12 MPI tasks (12 MICs, 16 OMP threads /task)	1448s
MPI/OpenMP	12 MPI tasks (12 MICs, 50 OMP threads/task)	931s

On MICs: Use OpenMP threads in preference to MPI processes

Harmonie Scalability (Ireland 5.5km, 6-hr Forecasts) Using 8 or 12 MICs, 1 MPI task/MIC



Utilization View (All Devices)



44

Average Core Temperature °C



14.534 GB
Total Memory Usage

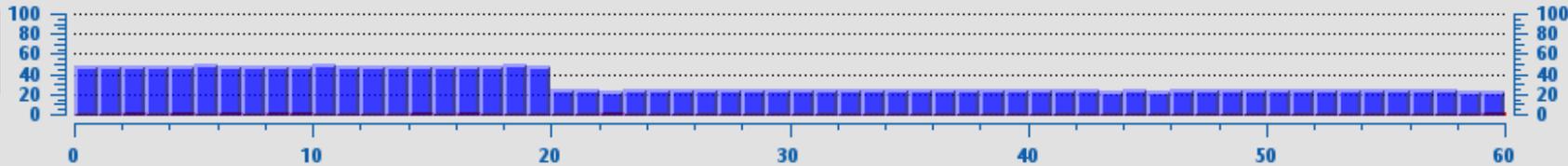
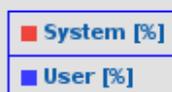


260 Watts
Total Power Usage

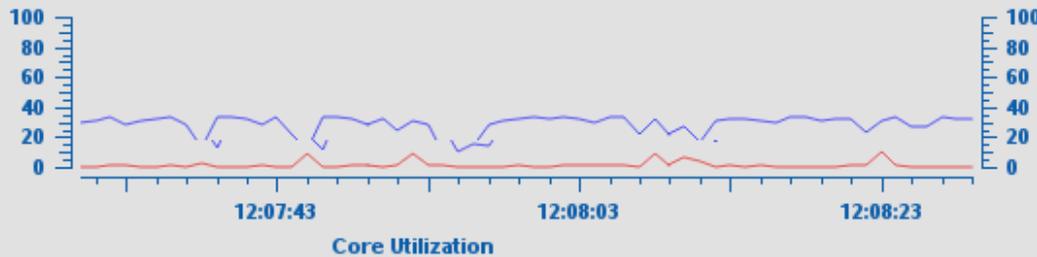
mic0: Core Histogram View



Individual Core Usage



mic1: Utilization View



41

Processor Core Temperature °C



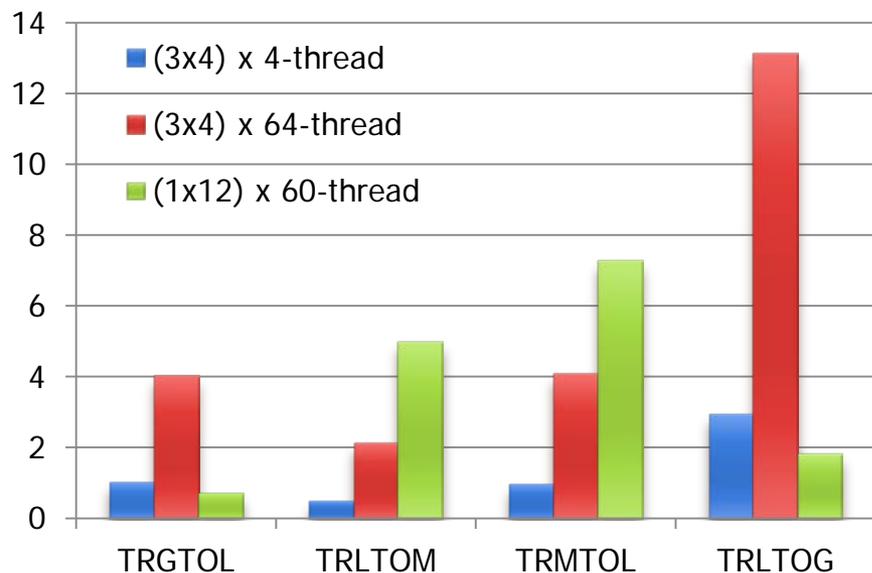
7212.2 MB
Memory Usage



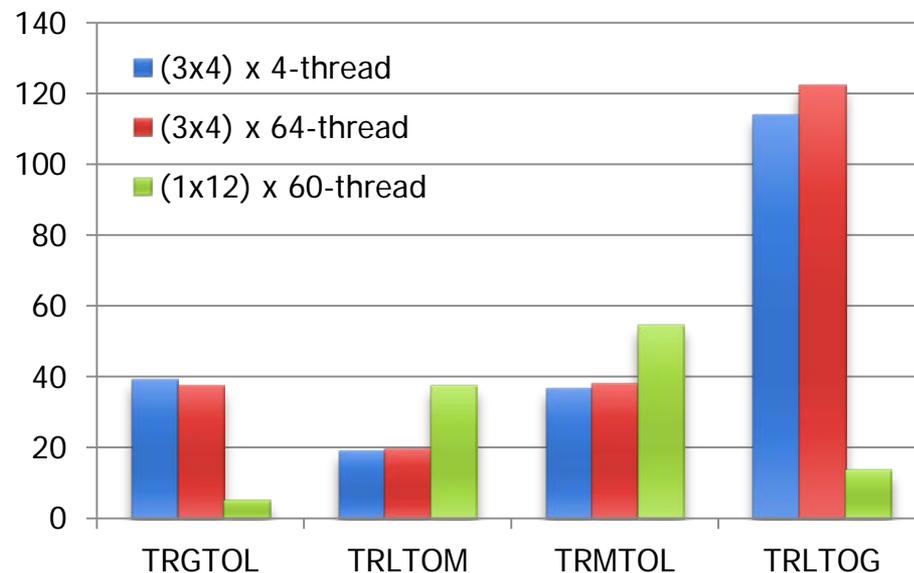
132 Watts
Power Usage

Non-threaded routines dominate at large thread-counts

Transforms (% of total time)



Transforms (wall-time in s)



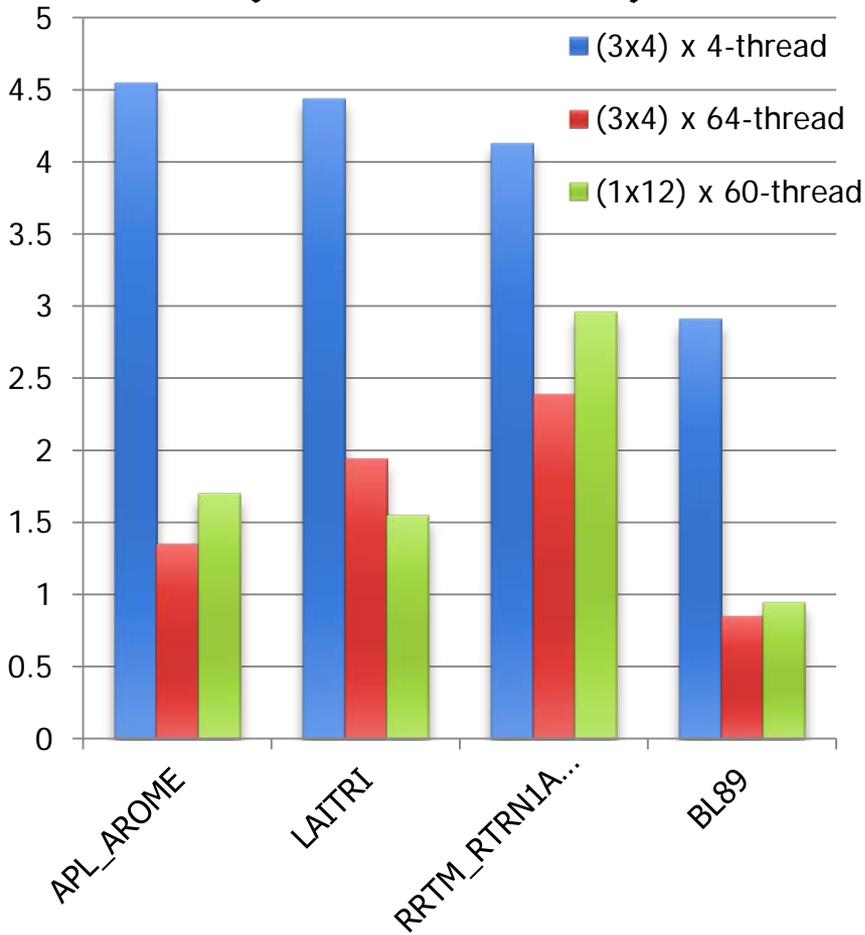
TRGTOL = Transform Grid TO Latitude decomposition

TRLTOM = Transform Latitude TO M (zonal) decomposition

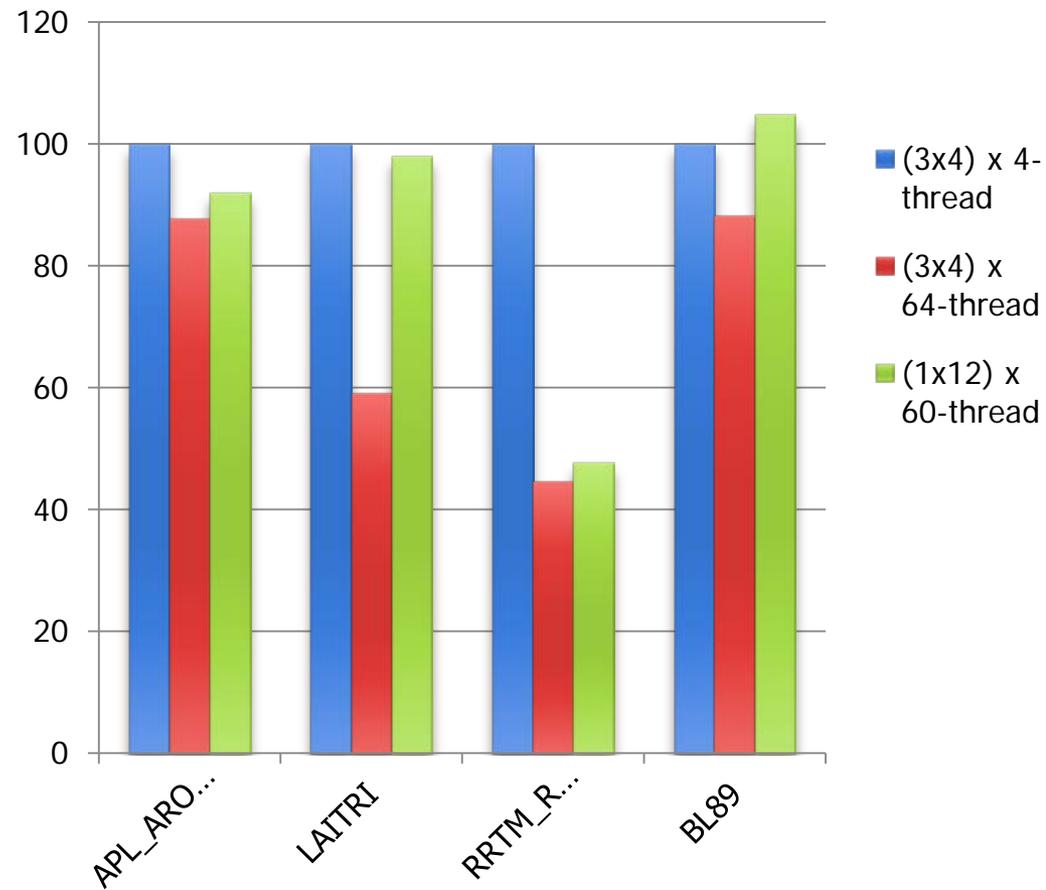
TRMTOL = Transform M (zonal) TO Latitude decomposition

TRLTOG = Transform Latitude decomposition TO Grid

Main Threaded Routines (% of total time)



Parallel Efficiency (% of ideal)



Issues

- Much performance (cores, threads) left unused because of memory limits.
- To run Harmonie efficiently on the Xeon Phi coprocessors, need a problem size big enough to scale to ~100+ threads, yet small enough to fit in < 8GB memory.
 - Next-generation 7000-series MIC processors have 16 GB memory.
- Could OMP_NUM_THREADS be increased without increasing memory usage?
 - Reduce number of “private” OMP variables?
 - Use more MPI tasks/MIC, fewer OMP-threads/MPI-task?
 - Find “optimal” KMP_STACKSIZE?
- **Symmetric mode** (HARMONIE running on both host and MIC processors simultaneously) currently “hangs” in first MPI collective.
 - *Still, most promising prospect...*
- **Offload mode** has many issues with pointers in derived data-types, which will require many source-code changes.
 - Is that even worthwhile?