Symmetric Computing

John Cazes

Texas Advanced Computing Center
Symmetric Computing

Run MPI tasks on both MIC and host

• Also called “heterogeneous computing”

• Two executables are required:
  – CPU
  – MIC

• Currently only works with Intel MPI

• MVAPICH2 support coming
Balance

• How to balance the code?

<table>
<thead>
<tr>
<th></th>
<th>Sandy Bridge</th>
<th>Xeon Phi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory</td>
<td>32 GB</td>
<td>8 GB</td>
</tr>
<tr>
<td>Cores</td>
<td>16</td>
<td>61</td>
</tr>
<tr>
<td>Clock Speed</td>
<td>2.7 GHz</td>
<td>1.1 GHz</td>
</tr>
<tr>
<td>Memory Bandwidth</td>
<td>51.2 GB/s(x2)</td>
<td>352 GB/s</td>
</tr>
<tr>
<td>Vector Length</td>
<td>4 DP words</td>
<td>8 DP words</td>
</tr>
</tbody>
</table>
Balance

Example: Memory balance
Balance memory use and performance by using a different # of tasks/threads on host and MIC

Host
16 tasks/1 thread/task
2GB/task

Xeon PHI
4 tasks/? threads/task
2GB/task
Balance

Example: Performance balance
Balance performance by tuning the # of tasks and threads on host and MIC

- **Host**
  - 16 tasks/1 thread/task
  - 2GB/task

- **Xeon PHI**
  - 4 tasks/30 threads/task
  - 2GB/task
Symmetric run on 1 Node

mpiexec.hydra \ 
–n 16 –host localhost ./host.exe \ 
:–env OMP_NUM_THREADS 30 \ 
–env LD_LIBRARY_PATH $MIC_LD_LIBRARY_PATH \ 
–env I_MPI_PIN_MODE mpd \ 
–env KMP_AFFINITY balanced \ 
–n 4 –host mic0 ./mic.exe

16 tasks on host

4 tasks on mic0

Environment variables for MIC tasks
Environment variables for MIC

By default, environment variables are “inherited” by all MPI tasks.

Since the MIC has a different architecture, several environment variables must be modified:

- **OMP_NUM_THREADS** – # of threads on MIC
- **LD_LIBRARY_PATH** – must point to MIC libraries
- **I_MPI_PIN_MODE** – controls the placement of tasks
- **KMP_AFFINITY** – controls thread binding
Task Binding

When using IMPI, process binding may be controlled with the following environment variable:

- `I_MPI_PIN_MODE=<pinmode>`

<table>
<thead>
<tr>
<th>pin mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mpd</td>
<td>mpd daemon pins MPI processes at startup (Best performance for MIC)</td>
</tr>
<tr>
<td>pm</td>
<td>Hydra launcher pins MPI processes at startup (Doesn’t appear to work on MIC)</td>
</tr>
<tr>
<td>lib</td>
<td>MPI library pins processes BUT this does not guarantee colocation of CPU and memory (Default)</td>
</tr>
</tbody>
</table>
Thread Placement

Thread placement may be controlled with the following environment variable

- `KMP_AFFINITY=<type>`

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
<th>Diagram</th>
</tr>
</thead>
<tbody>
<tr>
<td>compact</td>
<td>pack threads close to each other</td>
<td><img src="image" alt="compact diagram" /></td>
</tr>
<tr>
<td>scatter</td>
<td>Round-Robin threads to cores</td>
<td><img src="image" alt="scatter diagram" /></td>
</tr>
<tr>
<td>balanced</td>
<td>keep OMP thread ids consecutive (MIC only)</td>
<td><img src="image" alt="balanced diagram" /></td>
</tr>
<tr>
<td>explicit</td>
<td>use the proclist modifier to pin threads</td>
<td></td>
</tr>
<tr>
<td>none</td>
<td>does not pin threads</td>
<td></td>
</tr>
</tbody>
</table>
Symmetric Launcher

Analog of ibrun for launching symmetric applications

ibrun.symm –m <MIC executable> –c <CPU executable>

• Requires both host and MIC executable
• Environment variables for number of tasks and threads on MIC
  – MIC_MY_NSLOTS(Default=4)
  – MIC_OMP_NUM_THREADS(Default=30)
• Will work on multiple nodes
Symmetric Launcher Example

mpif90 -openmp -mmic prog.f90 -o a.out.mic
mpif90 -openmp prog.f90 -o a.out.cpu
export OMP_NUM_THREADS=1
export MIC_MY_NSLOTS=2
export MIC_OMP_NUM_THREADS=2
ibrun.symm -m ./a.out.mic -c ./a.out.cpu

- a.out.mic will run with 2 tasks and 2 threads/task per MIC card (1 card/node)
- a.out.cpu will run with the SLURM provided number of tasks and 1 thread/task