

Lab: Symmetric Computing MPI on MIC and CPU

Lab: Symmetric Computing

- What you will learn:
 - How to compile MPI code with Intel MPI library.
 - Launch MPI code on MIC only (native), and on both CPUs and MIC (symmetric mode)
- What you will do:
 - Run in MPI symmetric mode using `mpiexec.hydra`
 - Run in MPI symmetric mode using `ibrun.symm`
 - Run an MPI code with offload

Part 0 – Grab the Lab Files

- Login to Stampede:

```
$ ssh <username>@stampede.tacc.utexas.edu
```

- Untar the mic_mpi.tar file (in ~train00) into your directory:

```
$ tar xvf ~train00/mic_mpi.tar
```

- Use idev to launch a 1-node interactive session:

```
$ idev
```

Part 1 – Native MPI Execution (mpiexec.hydra)

- Move into the mpi/symmetric dir. Change to the impi environment:

```
$ cd mpi/symmetric
$ module swap mvapich2 impi
```

- Create a MIC and CPU binary for pi_hybrid:

```
$ mpicc -openmp pi_hybrid.c -mmic -o pi_hybrid.mic
$ mpicc -openmp pi_hybrid.c -o pi_hybrid.cpu
```

- Source the setup file to add IMPI environment variables:

```
$ source ./setup_mic_env.sh
```

- Execute on MIC only (Native): 8 tasks and 30 threads/task:

```
$ mpiexec.hydra \
  -env OMP_NUM_THREADS 30 \
  -env LD_LIBRARY_PATH $MIC_LD_LIBRARY_PATH \
  -env I_MPI_PIN_MODE mpd \
  -n 8 -host mic0 ./pi_hybrid.mic          #This is rather tedious!
                                           # use ./do_native script
```

Part 1 – Native MPI Execution (mpiexec.hydra)

- Execute on MIC+16 tasks on the host (CPU):

```
$ mpiexec.hydra \  
-n 16 -host localhost ./pi_hybrid.cpu : \  
-env OMP_NUM_THREADS 30 \  
-env LD_LIBRARY_PATH $MIC_LD_LIBRARY_PATH \  
-env I_MPI_PIN_MODE mpd \  
-n 8 -host mic0 ./pi_hybrid.mic
```

#This is even more complicated!

#You can use ./do_sym script instead

- Hmm, why do they call this symmetric 😊?

Part 2 – Symmetric MPI (ibrun.symm)

- Let's try this with TACC's ibrun.symm launcher:

1. Create environment variables for MIC:

```
$ export MIC_PPN=8
```

```
$ export MIC_OMP_NUM_THREADS=30
```

```
$ export MIC_MY_NSLOTS=8
```

2. Now use ibrun.symm to launcher a MIC (-m) binary:

```
$ ibrun.symm -m pi_hybrid.mic
```

3. Now launcher MIC (-m) and CPU (-c) binaries:

```
$ ibrun.symm -m pi_hybrid.mic -c pi_hybrid.cpu
```

Part 3 – MPI with Offload

- Change back to mvapich2 MPI. Swap and look for mvapich2 in module list:

```
$ module swap impi mvapich2
$ module list
```

- Move into the offload directory:

```
$ cd $HOME/mpi/offload
```

- Create an offload binary for pi_offload:

```
$ mpicc -O3 -openmp pi_offload.c -o pi_offload
```

- Set offload reporting on, # of threads, etc.:

```
$ export OFFLOAD_REPORT=2
$ export MIC_ENV_PREFIX=MIC
$ export MIC_OMP_NUM_THREADS=120
$ export MIC_KMP_AFFINITY=scatter
```

- Launch MPI-offload code with ibrun, using 2 MPI tasks:

```
$ ibrun -n 2 -o 0 ./pi_offload
```

- Compare pi_offload.c and pi_hybrid.c codes in the offload and symmetric directories.