MPI

What is MPI?

- MPI (Message Passing Interface) is a specification for message passing libraries
  - *More specifically it addresses how data is moved from one process to another*
- MPI is standardized but can vary in implementations
  - Standard versions for *C, C++,* and *Fortran*
  - Language bindings exist for *Python, Java, MATLAB, R,* as well as a few other languages
- MPI Implementations on Proteus
  - *OpenMPI (recommended)*
  - *MVAPICH2*

https://computing.llnl.gov/tutorials/mpi/

Some MPI terminology

- **process** – MPI object that uses a core to execute instructions
- **communicator** – a collection of MPI processes that can send and receive information from each other
- **rank** – unique identifier for each process
- **size** – total number of processes in a communicator

Some Common MPI Functions

- MPI_Init(&argc, &argv)
- MPI_Comm_size(*comm,* &numprocs)
- MPI_Comm_rank(*comm,* &id)
- MPI_Get_processor_name(processor_name, &name_len)
- MPI_Send(buff, BUFSIZE, *MPI_Datatype*, source, TAG, *comm*)
- MPI_Recv(buff, BUFSIZE, *MPI_Datatype*, source, TAG, *comm*, *status*)
- MPI_Finalize()

Let's break down MPI_Recv's inputs

- MPI_Recv(buff, BUFSIZE, *MPI_Datatype*, source, TAG, *comm*, *status*)
  - **buff** - initial address of buffer
  - **BUFSIZE** – maximum number of elements in buffer
  - **MPI_Datatype** – datatype of each buffer element
  - **source** – rank of the source
  - **TAG** - message tag
  - **comm** – MPI communicator
  - **status** – an MPI_Status object

How to load MPI

- module avail proteus

```
[cwf25@proteusi01 OpenMPI]$ module avail proteus

------------------------------------------------------------------------------------------------------------------------------------------------------ /cm/shared/modulefiles ------------------------------------------------------------------------------------------------------------------------------------------------------
proteus      proteus66    proteus-gpu  proteus-new  proteus-rh68

------------------------------------------------------------------------------------------------------------------------------------------------------ /mnt/HA/opt/modulefiles ------------------------------------------------------------------------------------------------------------------------------------------------------
proteus-blas/gcc/64/20110419                        proteus-fftw3/intel/gcc/64/3.3.3                    proteus-mvapich2/intel/64/1.9-mlnx-ofed             proteus-openmpi/gcc/64/1.8.1-mlnx-ofed
proteus-fftw2/amd/gcc/2.1.5                         proteus-fftw3/open64/64/3.3.3                       proteus-mvapich2/intel/64/2.0.1-mlnx-ofed-2.1-dont-use proteus-openmpi/gcc-cuda/64/1.8.1-mlnx-ofed
proteus-fftw2/gcc/64/double/2.1.5                   proteus-gsl/gcc/64/1.16                             proteus-mvapich2/intel-cuda/64/2.1rc1-mlnx-ofed-2.1 proteus-openmpi/intel/2015/1.8.1-mlnx-ofed
proteus-fftw2/gcc/64/float/2.1.5                    proteus-hdf5_18/gcc/1.8.14-mpi                      proteus-mvapich2/open64/64/1.9-mlnx-ofed            proteus-openmpi/intel/2015/cuda/6.0/1.8.1-mlnx-ofed
proteus-fftw2/mvapich2/open64/64/double/2.1.5       proteus-hdf5_18/gcc/1.8.14-serial                   proteus-netcdf/intel/2015/4.4.1                     proteus-openmpi/intel/64/1.6.5-mlnx-ofed
proteus-fftw2/mvapich2/open64/64/float/2.1.5        proteus-hdf5_18/intel/2015/1.8.14-serial            proteus-netcdf/intel/2015/4.5.0                     proteus-openmpi/intel/64/1.8.1-mlnx-ofed
proteus-fftw2/open64/64/double/2.1.5                proteus-hdf5_18/intel/2015/1.8.17-serial            proteus-netcdf-fortran/intel/2015/4.4.4             proteus-openmpi/open64/64/1.6.5-mlnx-ofed
proteus-fftw2/open64/64/float/2.1.5                 proteus-lapack/gcc/64/3.5.0                         proteus-openblas/ivybridge/0.3.0                    proteus-openmpi/open64/64/1.8.1-mlnx-ofed
proteus-fftw3/amd/gcc/64/3.3.3                      proteus-mvapich2/gcc/64/1.9-mlnx-ofed               proteus-openblas/sandybridge/0.2.19
proteus-fftw3/gcc/64/3.3.3                          proteus-mvapich2/gcc/64/2.0.1-mlnx-ofed-2.1-dont-use proteus-openblas/sandybridge/0.3.0
proteus-fftw3/intel/2015/3.3.7                      proteus-mvapich2/gcc-cuda/64/1.9-mlnx-ofed          proteus-openmpi/gcc/64/1.6.5-mlnx-ofed
```

- Then load the MPI you want:
  - *module load proteus-openmpi/gcc/64/1.8.1-mlnx-ofed*

How to load MPI

- Load the modules into your code
  - *For C this line would be #include <mpi.h>*
- Compile if necessary
  - *For C use mpicc*
- In your job script make sure MPI modules are loaded and then run MPI
  - *Set OMP_NUM_THREADS if necessary*
  - *$MPI_RUN myprogram.exe*

Some things to consider

- No. of slots
- No. of threads
- Implementation of MPI
- What your job does
- Other jobs on the cluster
- Hybrid?
- New nodes?

Results measured in Gflops

Flops = Floating Point Operations per second

Multi-Node Benchmarks (HPL)

AMD	1-node, 64-core	4-node, 256-core	16-node, 1024-core
mvapich2	446.9 GFLOPS	1708 GFLOPS	7084 GFLOPS
openmpi	426.4 GFLOPS	1003 GFLOPS	

INTEL	1-node, 16-core	4-node, 64-core	16-node, 256-core
mvapich2	281.8 GFLOPS	954.2 GFLOPS	
openmpi	281.0 GFLOPS	992.2 GFLOPS	959.8 GFLOPS
intel	288.6 GFLOPS	934.5 GFLOPS	

*From Wiki

Results measured in Gflops

Flops = Floating Point Operations per second

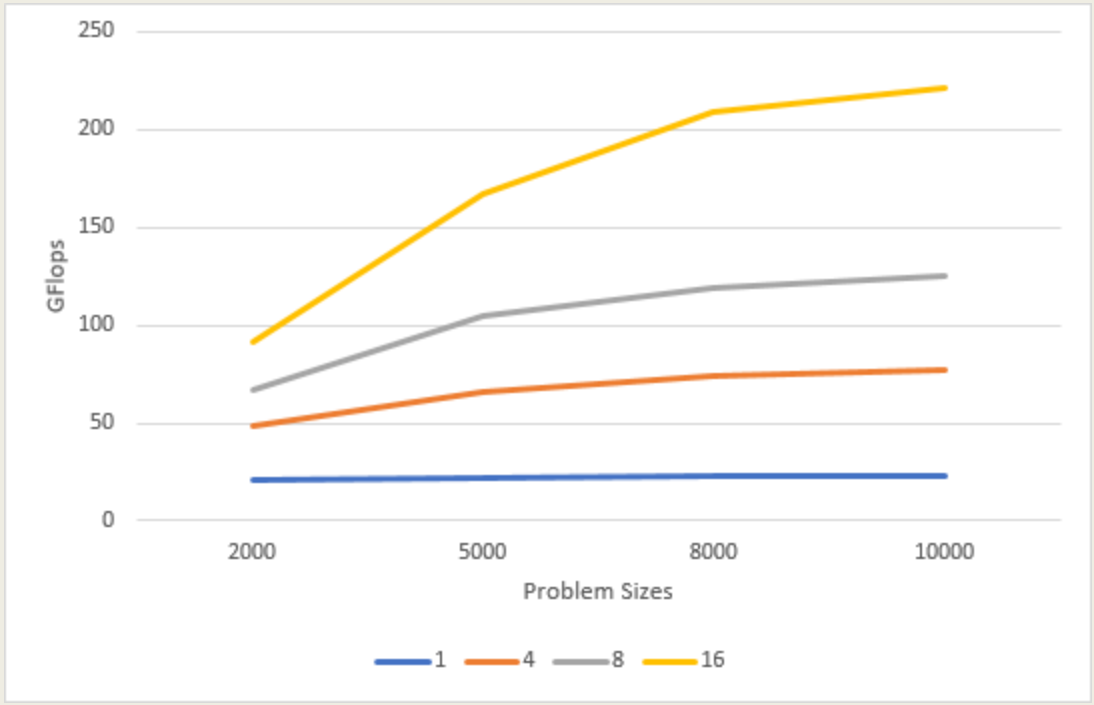
Single Node Benchmarks (HPL)

No. of Slots

Problem Sizes

	1	4	8	16
2000	20.70997222	48.21189815	67.15039352	91.2644213
5000	22.24525	66.27152315	105.3168333	167.774537
8000	22.52801064	73.65669444	119.3998611	208.7412903
10000	22.92831944	77.02326852	125.407963	221.9562963

OpenMPI on Intel Nodes



LET’S TRY IT!