1. Getting started with MPI

Basic model,

mpiexec supplies a) the number of hosts involved, b) their names, possibly in a hostfile, c) other parameters, such as whether to include the interactive host, d) the name of the program and its parameters.

Mpirun makes an ssh connection to each of the hosts, giving them sufficient information that they can find each other.

Python note load the TACC-provided python:

```
Module load python
```

And run it as:

```
ibrun python-mpi programname.py
```

Fortran, provides a module named mpi_f08.

Read routine prototypes.

C

int MPI_Comm_size(MPI_Comm comm,int *nprocs)

Fortran

CALL MPI_Comm_size(comm, size, ierr)

Python

MPI.Comm.Send(self, buf, int dest, int tag =0)

from mpi4py import MPI

2. MPI topic: Functional parallelism

The SPMD (single program multiple data) model

Different head

include “mpi.h” // for C

include “mpif.h” ! for Fortran

MPI initialization

Python, the import statement performs the initialization.

C: int MPI_Init (int *argc, char ***argv)

Fortran: MPI_Init (ierror)

INTEGER, OPTIONAL, INTENT(OUT) :: ierror

Conclude an MPI program

C: int MPI_Finalize (void)

Fortran: MPI_Finalize (ierror)

INTEGER, OPTIONAL, INTENT(OUT) :: ierror

Testing the initialized/finalized status by flag.

How to distinguish between processors?

Communicator: an abstract description of a group of prcesses.

MPI_Comm_size reports how many processes there are in all.

MPI_Comm_rank states what the number of the process is that calls this routine.

1. Getting started with OpenMP

2. OpenMP topic: Parallel regions

Question

1. what is the difference between the old processors (physically separated) and the modern processors in using MPI program?

2. What is MPI_Abort?

3. What is the key point of functional parallelism?