Programming with MPI

Compiling and running MPI programs

Jan Thorbecke

Compiling MPI Programs

Compiling and Starting MPI Jobs

- Compiling:
  - Need to link with appropriate MPI and communication subsystem libraries and set path to MPI Include files
  - Most vendors provide scripts or wrappers for this (mpif90, mpicc, ftn, etc)
- Starting jobs:
  - Most implementations use a special loader named mpirun
    - **`mpirun -np <no_of_processors> <prog_name>`**
  - Sometimes you can also use (Intel MPI specific)
    - **`mpiexec.hydra -n <no_of_processors> <prog_name>`**

Compilation and Parallel Start

- Compilation in C: mpicc -o prog prog.c
  mpiicc -o prog prog.cpp (Intel)
- Compilation in C++: mpiCC -o prpg prog.c
  mpicxx -o prog prog.cpp
  mpiicpc -o prof prog.C
  (Intel)
- Compilation in Fortran: mpif77 -o prog prog.f
  mpif90 -o prog prog.f90
- Executing program with num processes:
  mpirun –n num ./pra
  mpiexec -n num ./prg

MPICH: a Portable MPI Environment

- MPICH is a high-performance portable implementation of MPI (both 1, 2 and 3).
- It runs on MPP's, clusters, and heterogeneous networks of workstations.
- The CH comes from Chameleon, the portability layer used in the original MPICH to provide portability to the existing message-passing systems.
- http://www.mcs.anl.gov/research/projects/mpich2/

Compiling MPI with MPICH

- From a command line:
  `mpicc -o prog prog.c`
- Use profiling options
  - **--help** Find list of available options
- The environment variables `MPICH_CC`, `MPICH_CXX`, `MPICH_F77`, and `MPICH_F90` may be used to specify alternate C, C++, Fortran 77, and Fortran 90 compilers, respectively.

Example: MPI program hello_world.c

```
#include <mpi.h>
#include <stdio.h>

int main(int argc ,char *argv[])
{
  MPI_Init(&argc, &argv);

  fprintf(stdout, "Hello World!\n);

  MPI_Finalize();

  return 0;
}
```


Example: compile with MPI

```
-bash-3.1$ mpicc -o hello_world hello_world.c
-bash-4.2$ mpirun -np 4 ./hello_world
Hello world!
Hello world!
Hello world!
Hello world!
```

Access and Exercises

- Open a terminal (ssh) to

```
ameland.ta.tudelft.nl
```

- Login with your username (mpilrn#) and password

```
ssh mpilrn1@ameland.ta.tudelft.nl
```

- Copy exercises from:

```
tar xvfz /vardim/home/thorbcke/MPIcourse.tgz
```


Exercise HelloWorld

- cd to HelloWorld
- browse to the README, do not yet try to answer the questions
- compile and run the program Hello_world
- This is to make sure everybody can compile and run MPI programs.

Other MPI implementations

- OpenMPI:
  - https://www.open-mpi.org
  - don't confuse with OpenMP
  - good implementation for IB networks
- Intel MPI
  - https://software.intel.com/en-us/intel-mpi-library
  - bundled with compiler, tuned for Intel OmniPath Architecture
- Vendor specific MPI from
  - Cray
  - IBM
  - HP/SGI

Books about MPI

- **Reference MPI standard**
  http://mpi-forum.org/docs/mpi-3.1/mpi31-report.pdf
- **Parallel Programming with MPI**, Peter S. Pacheco, Morgen Kaufmann Publishers, 1997 - very good introduction.
- **MPI: The Complete Reference**, Marc Snir and William Gropp et al, The MIT Press, 1998 (2-volume set)
- **Using MPI: Portable Parallel Programming With the Message-Passing Interface and Using MPI-2**: Advanced Features of the Message-Passing Interface. William Gropp, Ewing Lusk and Rajeev Thakur, MIT Press, 1999

Web-tutorials about MPI

- http://mpitutorial.com
- http://www.nccs.nasa.gov/tutorials/mpi_tutorial2/mpi_II_tutorial.html
- https://fs.hlrs.de/projects/par/par_prog_ws/
- https://computing.llnl.gov/tutorials/mpi/
- http://mpi.deino.net/mpi_functions/index.htm
- https://www.mpich.org

MPI Forum

- MPI-1 Forum
  - First message-passing interface standard.
  - Sixty people from forty different organizations.
  - Users and vendors represented, from US and Europe.
  - Two-year process of proposals, meetings and review.
  - MPI 1.0 — June,1994.
  - MPI 1.1 — June 12, 1995.
- MPI-2 Forum July 18, 1997
- MPI-3 Forum 21 September 2012
- The Standard (3.0) itself:
  - at http://www.mpi-forum.org
  - All MPI official releases, in both PDF and HTML

MPI - Message Passing Interface

- MPI or MPI-1 is a library specification for message-passing.
- MPI-2: Adds in Parallel I/O, Dynamic Process management, Remote Memory Operation, C++ & F90 extension …
- MPI Standard:
  - http://www-unix.mcs.anl.gov/mpi/standard.html
- MPI Standard 1.1 Index:
  - http://www.mpi-forum.org/docs/mpi-11-html/node182.html
- MPI-2 Standard Index:
  - http://www.mcs.anl.gov/research/projects/mpi/mpi-standard/mpi-report-2.0/mpi2-report.htm#Node0
- MPI Forum Home Page:
  - http://www.mpi-forum.org/index.html

Job Management and queuing

Job Management and queuing

- On a large system many users are running simultaneously.
- What to do when:
  - The system is full and you want to run your 512 CPU job?
  - You want to run 16 jobs, should others wait on that?
  - Have bigger jobs priority over smaller jobs?
  - Have longer jobs lower/higher priority?
- The job manager and queue system takes care of it.

Job Management and queuing

- OpenPBS (torque/moab):
  - TORQUE: resource manager
  - http://www.adaptivecomputing.com/products/open-source/torque/
  - MOAB: workload manager
  - http://www.adaptivecomputing.com/products/hpc-products/moab-hpc-basic-edition/
- SLURM
  - http://slurm.schedmd.com
- Sun Grid Engine
  - http://gridengine.sunsource.net/

queuing commands torque/moab

- qsub
  - submit a job to a queue
- qstat
  - status of queued jobs: options -a, -Q
- qdel
  - delete job from queue
- showq
  - display jobs in the moab job queue

job script example torque/moab

```
#!/bin/bash
#PBS -N Name-of-job
#PBS -l nodes=2:ppn=4
#PBS -q verylong
#PBS -l walltime=1:00:00
#PBS -j eo
```

submit:

```
qsub -q normal job.scr
```

output:

```
jobname.ejobid
jobname.ojobid
```

SLURM

- sbatch
  - submit a job to a queue
- squeue
  - status of queued jobs: options -l
- scancel
  - delete job from queue
- sinfo
- sview
  - GUI

job script example slurm

```
#!/bin/bash
#SBATCH -J Name-of-job
#SBATCH --cpus-per-task=2
#SBATCH --ntasks=4
#SBATCH --nodes=1
#SBATCH -o snap46-%A.out
#SBATCH --time=1:00:00
```

submit:

```
sbatch job.scr
```

output:

```
snap46-jobid.out
```


Exercise: HelloWorld

- Write a basic queue script for the hello_world program.
- SLURM is installed as workload manager.
- run on multiple nodes and verify which nodes you have run
  - `mpirun --help` shows options you can use to get node information
  - are you running parallel, on multiple nodes, which nodes?