

Programming with MPI

content and definitions

Jan Thorbecke, Section of Applied Geophysics

Motivation and Goal

- Understand the basic functionality of MPI.
- Compile and run MPI programs.
- Write MPI programs with point to point communication and collective operations.
- Learn how to set-up domain decomposition and parallel IO.
- Combine MPI with OpenMP and vectorised kernels to match modern compute architectures.

Learning Outcomes

- Understand how and when to use MPI in an application.
- Analyse an existing program for OpenMP and MPI parallelisation possibilities.
- Knowledge how to write MPI code that can scale to 1000's of processors.

Organization

Dates: 7, 12, 21 May and 11, 18 June 2017

Time: 13:30 ~ 17:30

Points: 2

Examination: attendance, exercises and discussions during the course.

Schedule and rooms

CITG Building:

Stevinweg 1

2628 CN Delft

- 11 May room 2.110
- 18 May room 2.99
- 23 May room 2.99
- 1 June room 2.110

Contents

- Introduction (1)
- Parallel Programming (1)
- Basic send and receive (1)
- Parallel Hardware (2)
- Advanced point to point (2,3)
- Collectives (3,4)
- Topology (4)
- Hybrid MPI + OpenMP (4)
- MPI-IO (4, optional)

Day 1

- How to run and compile MPI programs (1 hour)
 - mpich
 - work load manager: slurm moab/torque
 - exercise: HelloWorld
- Parallel programming (1 hour)
 - Programming models
 - OpenMP
 - MPI communication concepts
- Basic Send & Recv (1+ hour)
 - exercise: SendRecv
 - exercise: ParallelSum
 - exercise: PingPong (optional advanced 1+2)

Day 2

- Parallel Hardware (45 minutes)
 - Memory organisation
 - topology
- Basic Send & Recv (continued)
 - exercise: SendRecv
 - exercise: PingPong (optional advanced 1+2)
 - exercise: Ring (optional)
- Collectives
 - exercise: BroadcastBarrier
 - exercise: GatherScatter
 - exercise: Reduce
 - exercises: Collectives (MatrixVector, Pi, others are optional)

Day 3

- Collectives (continued)
 - MPI + OpenMP
 - exercise: Matrix-Vector MPI + OpenMP
- Topology in MPI
 - communication groups
 - virtual topologies
 - Cartesian domain decomposition
 - exercise: MPI_Cart
 - MPI data-types
 - exercise: DataTypes

Day 4

- Topology in MPI (continued)
 - exercise: MPI_Cart (jacobi.f90 optional)
 - MPI data-types
 - exercise: DataTypes
- MPI-IO (optional)
 - Independent
 - exercise: exa1 MPI_File_write_at
 - exercise: exa2 MPI_File_set_view
 - collective
 - exercise: mpiio_2D MPI_File_write_all

The Exercises

- Exercises are an important part to learn MPI. If you already have enough experience with some parts in the exercise you can skip those parts.
- Exercises sometimes require small changes to be made or insert the MPI calls. Every exercise comes with a step by step procedure to follow.
- The solution is more important than the arguments of the MPI functions.
- The exercises are obligatory to pass the course.

Feedback

Feedback about the course is highly appreciated:

- mix of exercises with lectures?
- exercises too easy / complex / time consuming / useful ?

You can always send me an e-mail or visit me at my desk in room 3.12 (CITG Building on Tuesday's and Thursday's only).