

# 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:** 25 April and 2, 9, 16 May 2019

**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

- 25 April room 4.99
- 02 May room 1.21
- 09 May room 1.21
- 16 May room 3.02

## 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 (this can be extended to 2+hours)
  - MPI communication concepts
- Basic Send & Recv (1+ hour)
  - exercise: SendRecv
  - exercise: ParallelSum
  - exercise: PingPong/Basic (optional Advanced 1+2)

# 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 2

- Parallel Hardware (45 minutes)
  - Memory organisation
  - topology
- Basic Send & Recv (continued)
  - exercise: SendRecv
  - exercise: PingPong/Basic (optional Advanced 1+2)
- Collectives
  - exercise: SendRecvCollectives (a:BroadcastBarrier)
  - exercise: SendRecvCollectives (b,c: GatherScatter)
  - exercise: Reduce
  - exercises: Collectives (MatrixVector)

## 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 3.03 (CITG Building on Tuesday's and Thursday's only).