***Index***

[1 User defined domain decomposition 1](#_Toc516566007)

[1.1 Introduction 1](#_Toc516566008)

[1.2 Model setup 1](#_Toc516566009)

[1.3 Automatic generation of a user defined domain decomposition configuration file 3](#_Toc516566010)

[1.4 Linear sub-grid decomposition 5](#_Toc516566011)

[1.5 Center of mass sub-grid decomposition 5](#_Toc516566012)

[1.6 Horizontal/Vertical mass based decomposition 5](#_Toc516566013)

# User defined domain decomposition

## Introduction

MOHID allows for domain decomposition. This domain partitioning in sub-domains can be easily done in MOHID by automatically subdividing the domain in groups of rows of grid cells. The number of subdomains is set as an input parameter when launching model execution. This method is typically inefficient, as some subdomains will run much faster than others (e.g. sub-areas with high number of land points), thus hindering overall performance.

A new and more versatile method was developed allowing the user to control the configuration of the model domain subdivision. The configuration can be done manually or using a pre-processing software that facilitates some of the tasks. A detailed description of the new method and the pre-processing tool can be found below.

## Model setup

In order to setup user defined domain decomposition, it is required to set a keyword (**D\_DECOMP)** in the bathymetry file of the corresponding domain, defining the path to the domain decomposition configuration file. In this file a set of keywords and blocks allows this configuration and are presented below with the matching description.

|  |  |  |
| --- | --- | --- |
| **Keyword** | Block | Description |
| **MPI\_ID** | <BeginSubDD> <EndSubDD> | Unique number identifying the sub-domain |
| **ILB\_IUB** | <BeginSubDD> <EndSubDD> | Minimum and maximum I indexes of the subdomain (e.g. 1  132) |
| **JLB\_JUB** | <BeginSubDD> <EndSubDD> | Minimum and maximum J indexes of the subdomain (e.g. 1 108) |
| **INTERFACES\_NUMBER** | - | Number of adjacent interfaces between the sub-domains |

The interfaces between sub-domains must be set in the input data file by defining 2 blocks:

* <BeginInterfaceSN> … <EndInterfaceSN> (South-North interfaces)
* <BeginInterfaceWE>…<EndInterfaceWE> (West-East interfaces)

In each block a set of pairs of sub-domain IDs sharing interfaces must be defined. Below an example (Figure 1) of the full definition of the domain decomposition file dividing a domain in 4 sub-domains (Figure 2):

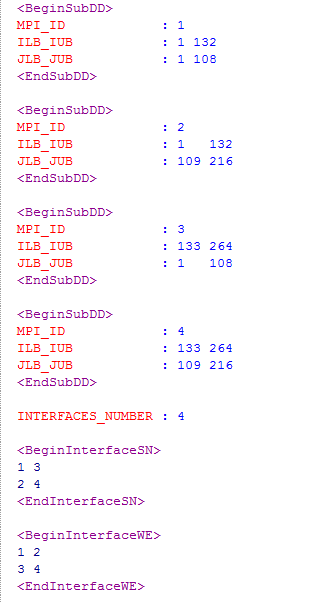


Figure 1 - Example of a domain decomposition configuration file

As shown in Figure 2, the model uses the concept of “halo” area, i.e. areas shared by each sub-domain over their adjacent interfaces. The size of these areas can be set, in number of grid cells, in the bathymetry file, using the keyword HALOPOINTS. By default, the number of grid cells in the “halo” area is 3.

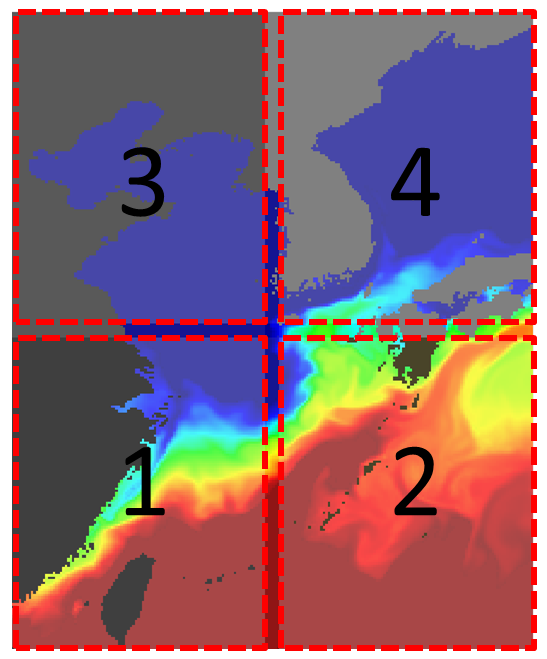
* 

Figure 2 – KOOS Level 2 domain partition into 4 subdomains (areas not covered by dotted lines are “halo” areas, i.e. shared by 2 domains)

## Automatic generation of a user defined domain decomposition configuration file

Generating a user defined domain decomposition configuration file can be relatively simple when splitting the domain up to four sub-domains. Doing it for a higher number of sub-domains can be an arduous task and prone to errors.

A software was developed (written in FORTRAN and to be included in the MOHID Framework), that can automatically create this file with minimal configuration effort. The program can be configured in similar fashion as any other program in the MOHID software family, via an input data file with keyword like commands. The file is mandatorily named **“InputDomainDecomposition.dat”**. In the table below, a full list of configuration options available for this input file is presented.

|  |  |  |
| --- | --- | --- |
| **Keyword** | Type | Description |
| **BATHYMETRY\_FILE** | Character string | Path to the bathymetry file relative to the domain to be partitioned |
| **GEOMETRY\_FILE** | Character string | Path to the vertical geometry file relative to the domain to be partitioned |
| **OUTPUT\_FILE** | Character string | Path to the domain decomposition configuration file used by MOHID |
| **DIVISION\_METHOD** | Integer (1, 2 or 3) | Method for automatic domain decomposition (1 – Grid, 2 – Horizontal Stripes, 3 – Vertical Stripes) |
| **N\_DOMAINS\_X** | Integer | Number of subdomains in the X axis |
| **N\_DOMAINS\_Y** | Integer | Number of subdomains in the Y axis |
| **HAS\_MASTER** | Logical | Indicates if the domain to be partitioned has a master domain. If true, the ID attribute of each subdomain considers its existence |
| **USE\_CENTER\_OF\_MASS** | Logical | If domain decomposition method is 1 – Grid, then the decomposition is based on the center of mass of the 3D computational grid |

Although the **DIVISION\_METHOD** option only allows for 3 different options, there are in fact 4 methods for domain decomposition, as described below.

## Linear sub-grid decomposition

This method (default, **DIVISION\_METHOD : 1 + USE\_CENTER\_OF\_MASS : 0**) divides the domain computational grid in subdomains by dividing the number of grid cells in each grid axis by a user defined value (**N\_DOMAINS\_X** and **N\_DOMAINS\_Y**). The number of subdomains in each axis does not require being the same. This allows for configurations such as 2x2, 3x2, 4x6, etc.

## Center of mass sub-grid decomposition

This method (**DIVISION\_METHOD : 1 + USE\_CENTER\_OF\_MASS : 1**) divides the domain computational grid in subdomains by dividing the number of grid cells in each grid axis by a user defined value (**N\_DOMAINS\_X** and **N\_DOMAINS\_Y**). In this option, the number of subdomains in each axis needs to be the same. The program then computes a center of mass of the computational grid, based on the horizontal and vertical discretization and land masking, considering the actual number of active computational points. Based on this center of mass, the program then subdivides the grid according to the defined number of subdomains.

## Horizontal/Vertical mass based decomposition

These 2 options (Horizontal stripes: **DIVISION\_METHOD : 2** and Vertical stripes: **DIVISION\_METHOD : 3)** consider a similar approach to option described in 1.5, where the subdivision is made by computing subdomains with a similar number of active computational points, but allowing only for the subdivision to be made along one of the grid axis. This approach reduces the number of subdomain interfaces, thus reducing to an optimal minimum the communication computational loads between the subdomains. This approach produces also a more uniform distribution of the work load associated to each domain, thus reducing differential computation times between domains, and optimizing the use of resources. The possibility of subdivision into horizontal or vertical stripes, allows for optimization related with the main grid size and possibly to adjustments in compilation settings and specific aspects of the hardware used to run the model.