High Performance Computing
USN-HCMV, Paris 13
Joint Master 2022

———-

Worksheet 4

This workksheet consists in solving with a Schwarz Domain Decomposition technique the following 2D classical problem :

$$\begin{aligned} -\triangle u &= f \text{ sur } \Omega \\ u &= u_D \text{ sur } \partial\Omega \end{aligned}$$

(f, u_D) chosen so that the exact solution to this problem is the following :
$u(x) = sin(om * x + cx) * sin(om * y + cy) + x * x + y * y$

The Laplace operator is discretized by 2nd order centered Finite Difference on a regular cartesian mesh with step hx , hy.

1 Work plan

- Set up the Schwarz Domain Decomposition technique with Dirichlet interface boundary conditions.
- Study numerically (experimentally) the influence of the size of the overlap and compare with theoretical results
- Do the same with Neuman boundary conditions
- Implement the following Robin interface boundary conditions $\alpha u + \frac{\partial u}{\partial n}$
- Study numerically the influence of the size of the overlap as before and the influence of the α coefficient.
- Write a small report about your results

2 Provided Matlab functions

4 structures are defined to store the different code data.

- Mg : A global structure containing the definition of $\Omega = [a1, a2] \times [b1, b2]$, and the total number per direction Nx, Ny
- T : A structure defining the topology of the geometrical domain split
  T.Ndom = Number of sub-domains
  T.Ndomx = Number of domains in the x direction
  T.Ndomy = Number of domains in the y direction
- P : A "Problem" structure that contains for each sub-domain the computed solution array, the interface Dirichlet boundary array u_D, the right hand side f
- M : A local structure describing the geometry of each sub-domain, and lists of points that list points that will send values to the neighbour domain and points that will receive values from these neighbour domains.

```
M = struct('Nx',0,'Ny',0,'a1',0,'a2',0,'b1',0,'b2',0, ...
'hx',0,'x',0,'hy',0,'y',0, ...   % Geometry
'Ndom',0,'idom',0,  ...  % Number of domains, and domain number
'Ndir',0,'ldir',0,  ...  % Number of Dir. points and list
'Ninte',0,'linte',0, ... % Number of Interface points to be sent and list
'Nintr',0,'lintr',0, ... % Number of Interface points gthat receive and list
'nxe',0,'nye',0,...      % Inner normal for sending points
'nxr',0,'nyr',0,...      % Outer normal for sending points
'Nin',0,'in',0, ...      % Number and list if interior points
'Nvois',0,'novois',0,'lvoise',0,'lvoisr',0, ... % Number of neighbours ,
%  1 <= Neighbour number <= Ndom
'list',0);
% Number of interface neighbour points and concatenated list
```

2.1 para2D

First lines of your main program to be completed. It contains

— Geometry

— Discretisation

— Splitting Topology

local matrices, right hand side, ...

Your work is to add the rest

$$[P] = \text{schwarz}(P,M,T)$$

that returns P.usol

2.2 mcreamesh

Function that fills in the M structure : Number of points per each subdomain, number and list of neighbours, number and list of points that send and receive data from the neighbour domains.

2.3 FaceInt

Function called by mcreamesh to define normal vectors and interface descriptors.

2.4 init

Function that initialises the numerical solution, the exact solution, the Dirichlet boundary condition on $\partial\Omega$, the right hand side (sm0).

2.5 creamat

Function that defines the Laplace operator matrix +Dirichlet boundary conditions (ldir + lintr) (A(i,i)=1, a(i,j)=0 j $\neq$ i point Dirichlet)

2.6 visu

Function that visualises mesh or solution