Parallel Geometric Multigrid for Poisson problems

Supervisor. Emile Parolin (emile.parolin@sorbonne-universite.fr)

Keywords. Multigrid Method ; Distributed Computing ; Poisson problem.

Context. Multigrid methods are among the fastest numerical algorithms for the solution of large sparse systems of linear equations arising from the discretisation of elliptic problems. A detailed description of such methods can be found for instance in [1, 2]. These methods rely on a hierarchy of grids and their efficiency rests on two main ideas. The first one is to use coarse grids to construct approximate solutions that are then interpolated on finer levels. The second one is to smooth out high frequency components in the error that are present on finer grids before restricting the approximation on coarse grids.

To exploit modern large distributed computing architectures, parallel versions of multigrid methods need to be implemented. In this context a challenge for multigrid schemes is the ratio of communications compared to computations.

Description of the work. For some load f, we consider in this project a Poisson problem in a rectangular domain $\Omega = (0, a) \times (0, b)$, with homogeneous Dirichlet boundary conditions

$$\begin{cases} -\Delta u = f, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega. \end{cases} \tag{1}$$

The work will consists in :

1. writing a sequential algorithm to solve a second order finite-difference discretisation of the above Poisson problem using a multigrid method. Several smoothers will be implemented based on standard iterative methods (Jacobi, Gauss-Seidel, SOR...).
2. implementing a parallel version of the previous algorithm able to run on distributed architectures using message passing (MPI).

The language of implementation can be C/C++ or Julia. The correctness of the implementation will be checked by using a direct solver and considering right-hand-sides f for which analytic solutions are known. The convergence of the multigrid method will be studied and compared to standard general purpose iterative methods. Scaling tests for the parallel code will be performed.

Time permitting, several extensions can be considered such as writing a 3D version or considering the heat equation (which involves time, hence requires time discretisation schemes which can also be parallelized in a multigrid fashion).

References

[1] James W. Demmel. *Applied numerical linear algebra.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.

[2] Yousef Saad. *Iterative methods for sparse linear systems.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 2003.