11th World Congress on Computational Mechanics (WCCM XI)
5th European Conference on Computational Mechanics (ECCM V)
6th European Conference on Computational Fluid Dynamics (ECFD VI)
July 20–25, 2014, Barcelona, Spain

PARALLEL CHIMERA METHOD

Beatriz Eguzkitza[1] and Guillaume Houzeaux[2]

[1] Barcelona Supercomputing Center (BSC-CNS), Dept. Computer Applications in Science and Engineering, Edificio NEXUS I, Campus Nord UPC, Gran Capitán 2-4, 08034 Barcelona, Spain

Key words: *Chimera method, Coupling Meshes; Domain Decomposition Methods, Navier-Stokes equations; Incompressible flow; Parallelization*

The Chimera method was developed three decades ago as a meshing simplification tool. Different components are meshed independently and then glued together using a domain decomposition technique to couple the equations solved on each component. This coupling is achieved via transmission conditions (in the finite element context) or by imposing the continuity of fluxes (in the finite volume context). Historically, the method has then been used extensively to treat moving objects, as the independent meshes are free to move with respect to the others. The method is also useful to treat local refinement or in optimization problems.
At each time step, the main task in Chimera problems consists in recomputing the interpolation of the transmission conditions or fluxes, after a previous hole cutting step to create the interfaces of the subdomains. Our proposed method to solve this kind of problems is based on a geometrical coupling of independent meshes assembled in an implicit way (see [1]). The parallelization of the Chimera method in a distributed memory context is not an easy task, whenever the method is intended to be implicit. In common explicit implementations of the method, the transmission conditions are usually imposed iteratively so that the subdomains can be solved in a staggered way. In this case, the parallelization is more straightforward as one could use as many parallel instances of the code as subdomains and the only difficulty consists in exchanging of the transmission conditions through MPI. Figure 1 illustrates the differences between an explicit and an implicit coupling in parallel and illustrate the idea of the present work. On the one hand, in the case of the explicit coupling, each subdomain is solved independently with different parallel instances of the code, and the coupling is carried out between these instances in an iterative way; for example, solve 1, send transmission conditions to 2, solve 2, send transmission conditions to 1, and so on until convergence. In this case, the coupling involves only the right-hand side of the respective algebraic systems. On the other hand, in the implicit case proposed in this work, the transmission conditions are included in the matrix of the algebraic system and the global coupled solution is solved in parallel, regardless the existence of background and patches meshes typical of the Chimera problems. In reference [3] an implicit strategy is presented in the context of the finite volume

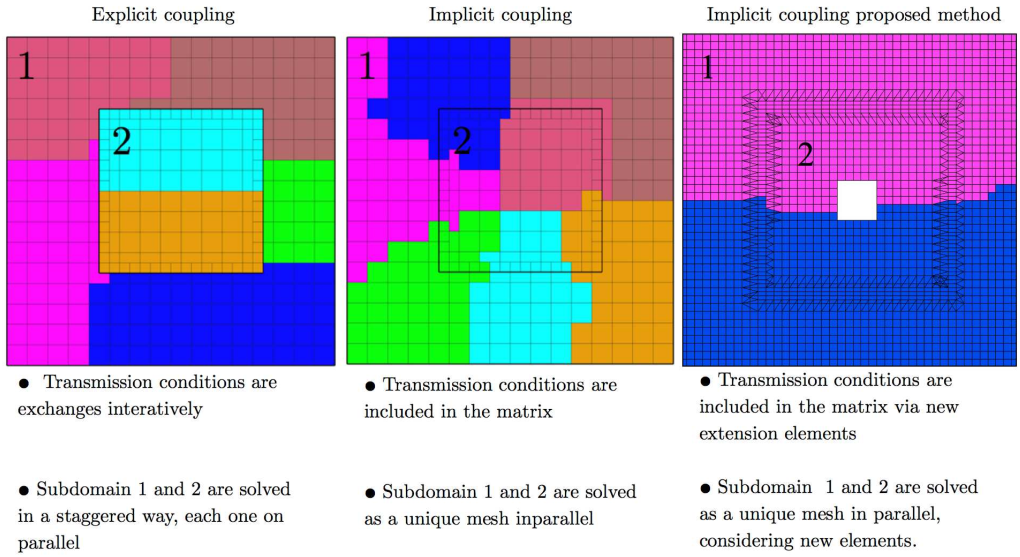

Figure 1: Chimera in parallel. (Left) Explicit coupling. (Middle) Implicit coupling. (Right) Implicit coupling with proposed method.

method. A good review of the parallelization issues related to overset grids is done in [2]. Although it is true that Chimera methods are useful in a variety of the applications it is worth to mention that the hole cutting preprocess also demands a lot of computational work which must be done in an efficient manner and the experience of the user has to be minimized.
All of this issues are the objective of the presented work. We are going to show a novel Chimera method based in a geometrical coupling, called HERMESH method, previously developed for fixed components where the coupling between the meshes is done as a pre-process, before the mesh partitioning is carried out for parallelization purpose. The main properties of the method presented in this work are the following: it is implicit, independent equation, working in a distributed memory context , it is valid to solve problems with components in relative motion and the amount of user inputs zero being the process automatic.

REFERENCES

[1] G. Houzeaux, B.Eguzkitza, R.Aubry, H.Owen and M.Vázquez. A Chimera Method for the Navier–Stokes Equations. *International journal for numerical methods in fluids*, Submitted.

[2] N.C. Prewitt, D. M. Belk and W. Shyy. Parallel computing of overset grids for aerodynamic problems with moving objects. *Progress in Aerospace Sciences*, Vol. **36**, 117–172, 2000.

[3] E. Ahusborde and S. Glockner. An implicit method for the Navier–Stokes equations on overlapping block-structured grids. *International journal for numerical methods in fluids*, Vol. **62**, 784–80, 2010.