

Implementation of mapping strategies in a distributed memory environment

Jordi Cotela-Dalmau^{*a}, Philipp Bucher^b, Aditya Ghantasala^b, Michael Andre^b,
Andreas Winterstein^b, Riccardo Rossi^a and Roland Wüchner^b

^a Centre Internacional de Mètodes Numèrics en Enginyeria (CIMNE)
Universitat Politècnica de Catalunya
Campus Nord UPC, 08034 Barcelona, Spain
e-mail: jcotela@cimne.upc.edu, rrossi@cimne.upc.edu

^bChair of Structural Analysis
Technische Universität München (TUM)
Arcisstr. 21, D-80333 München, Germany
e-mail: philipp.bucher@tum.de, aditya.ghantasala@tum.de, michael.andre@tum.de,
a.winterstein@tum.de, wuechner@tum.de

ABSTRACT

The simulation of coupled problems requires frequent exchange of data among the different domains. This can be achieved through the use of a mapping algorithm, which will ensure that mapped quantities have corresponding values on either side of the coupling interface, even when the interfaces themselves are moving or do not precisely match each other.

There is a clear trend in high performance computing (HPC) towards calculation servers composed of many individual processors. Computation on such clusters requires the use of a distributed memory programming model, in which not all data is readily available during the simulation. This requires a careful design of the code to avoid introducing bottlenecks where calculations must wait for data to be exchanged between the parallel subdomains.

In the context of mapping algorithms, distributed memory programs represent an additional challenge, since the data to be exchanged not only lies on different domains but is also distributed among multiple physical processors. We will address the practical aspects of the implementation of interface mapping functions to work in conjunction with MPI parallel finite element solvers for each coupled domain. The focus will be in the aspects that are specific to the parallel performance of the algorithm, such as identifying the origin–destination pairs for the interpolation when those lie on different processes, constructing a communication strategy to efficiently exchange data in distributed memory and the limitations that the MPI context imposes on the mapping algorithm itself. Different mapping algorithms, namely nearest node, element interpolation and mortar-type methods, will be considered.

The proposed approach is being implemented within the Kratos MultiPhysics finite element framework and validated through the solution of model problems, allowing the quantification of both the accuracy and the parallel performance of each algorithm, with the end goal of using the resulting implementation for the simulation of FSI problems in an MPI context.

REFERENCES

- [1] Bernardi, C., Maday, Y. and Patera, A. T. A new nonconforming approach to domain decomposition: the mortar element method. *Nonlinear Partial Differential Equations and Their Applications*, 1994. for the Lagrange multiplier. *SIAM Journal on Numerical Analysis*, 38(3):989–1012, 2000.
- [2] Dadvand, P., Rossi, R., Gil, M., Martorell, X., Cotela-Dalmau, J., Juanpere, E., Idelsohn, S. R., and Oñate, E. Migration of a generic multi-physics framework to HPC environments, *Computers & Fluids* 80 (2013) 301–309.