An integrated workflow for pre-processing and domain decomposition to finite-element reservoir simulations

Kleiton Andre Schneider[1,2], Yuri Shalom de Freitas Bezerra[2], João Vítor Venceslau Coelho[2], Ronaldo Tavares e Silva[2], Samuel Xavier de Souza[2]*, Sidarta Araújo de Lima[2], Adriano dos Santos[2]. [1]UFMS, [2]UFRN.

Abstract

The injection of polymers in porous media is an area of great interest in the academic and industrial community, due to the many relevant applications for different physical processes, mainly in petroleum engineering. We propose to develop a petroleum reservoir simulator program implemented in Python/Fortran, designed to simulate the injection of polymers in hydrocarbon reservoirs with efficient coupling of the process in the well-reservoir region. Equations arising from physical/mathematical modeling for pressure, flow and transport are discretized using the finite element method. To solve the resulting system of equations, which is linearized using the Newton-Raphson method, a predictor-corrector strategy based on time and space domain decomposition is used. It has already been demonstrated in the literature that the numerical simulations for polymer injection in porous media using this same methodology provide accurate solutions, with low computational cost, in several scenarios.

The main problem with some similar software is its usage workflow, as it requires extensive human interaction and does not allow complete automation of the process. Mesh generation is done by the GMSH API for Python. In general, there was a lot of interaction with its graphical interface for mesh generation. We simplify and automate the process of defining the physical domain and the desired parameters for the mesh, such as number of nodes/elements per entity, mesh type (regular/irregular), number of partitions, etc. We highlight the possibility of using adaptive meshes, with different local refinements for each region or partition, which allows for greater computational efficiency, in association with the space-time domain decomposition.

We use an optimized computational implementation for mesh pre-processing, where information about boundary conditions is computed. By writing the preprocessing entirely in Python, we were able to establish a direct connection between the generated mesh and the processing in Fortran, eliminating the need to save large input files to solve the problem. In this way, we simplified the workflow and extended the program to use automated domain decomposition and meshes with different refinement subdomains in order to leverage the code for parallelization, using MPI and OpenMP.

Tavares, R. S., Santos, R. B., Lima, S. A., Santos, A. dos, & Klein, V. (2022). A multi-timestep Dirichlet-Neumann domain decomposition method applied to the polymer injection in porous media. In Computational Geosciences (Vol. 26, Issue 4, pp. 757–778). Springer Science and Business Media LLC. https://doi.org/10.1007/s10596-022-10128-8

Geuzaine, C., & Remacle, J.-F. (2009). Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. In International Journal for Numerical Methods in Engineering (Vol. 79, Issue 11, pp. 1309–1331). Wiley. https://doi.org/10.1002/nme.2579