Support for the concurrent solution of PDE systems and generalization of specialist boundary conditions in Nektar++

Duration: 3 months, Researcher: Gabriele Rocco

Outline: Nektar++ is a high-order spectral/hp element toolkit primarily developed for fluid mechanics simulations. The code has an incompressible flow solver based around a semi-implicit velocity-correction algorithm as well as a compressible flow solver based on explicit discontinuous Galerkin and more recently Flux Reconstruction methods. The motivation of this project is the extension of the incompressible solver, but the developments will apply equally well to other solvers in the framework.

Project objectives: The project is to help capitalize on the expertise I gained, having recently defended his PhD thesis on stability analysis of flows using the Nektar++ incompressible flow solver. During my previous research assistant project I have implemented two features within the current framework, specifically a parallelization of the Arnoldi solver and extended the incompressible solver to support cylindrical coordinates. Several features, directly related to the algorithms which I implemented during my PhD, are desirable for the current research on vortical flows and my knowledge of the framework can therefore enable rapid development of the following two objectives in the 3 month project.

Solution of concurrent PDE systems: During my PhD I implemented an extension to support transient growth analysis, which is able to expose the presence of convective instabilities by means of an adjoint loop optimization. This approach consists of sequentially solving two different PDE systems, each with appropriate boundary conditions, repeatedly until convergence is reached. The current approach hard-codes the solver configuration for the case of transient growth analysis, which limits the scope and flexibility of the code to address more complex problems. In particular, the boundary conditions and parameters for both PDE systems are identical in the existing transient growth analysis, enabling the single configuration to be reused. The objective of this part of the project is to extend the capabilities of the code to support the concurrent use of multiple PDE systems, each with an arbitrary configuration. The task will be addressed specifically for the Navier-Stokes equations although it should transparently support other solvers.

Generalisation of specialist boundary conditions: Three main types of boundary conditions are present in Nektar++ (Dirichlet, Neumann, Robin). Other specialisations can be specified, which are typically solver-dependent, and may be related to time-dependent expressions or particular conditions for the flow variables. At present, these specialist conditions are hard-coded in the core libraries. This objective is to generalize the framework to encapsulate these boundary conditions and move them out of the low-level discretisation libraries and into the solver-level code where they most naturally live.

Alignment with Prism strategy:

- *Development of key staff*: Although I ultimately wishes to gain experience outside of my PhD work, I am in an position to aid a development of the Nektar++ framework through this short project. This necessarily will not only improve the framework through utilizing the previous knowledge gained during my PhD studies but will also allow me to promote these techniques to the broader PRISM group and thereby potentially help promote wider collaboration.
- *Collaboration other PRISM projects:* Research with Colin Cotter on a joint project of evaluating steady state solutions for stability analysis will benefit from the possibility to run simulations using different boundary conditions for each PDE.
- *Longer-term research*: the possibility of imposing different boundary conditions and parameters when two different solvers are run sequentially on the same geometry provides the possibility to address a range of scientific problems. This leads to a generalization of the transient growth approach and, the extension to other types of equations represent a desirable feature, which is currently object of investigation within Nektar++ research group.

Brief Work Plan: There is no need for any specific background work in the project. Most of the complexity will be involved in code debugging and validation testing. The initial stage of the project will require the extension of the current approach of treating the boundary conditions. The remainder of the project will be the restructuring and modification of the input format used for specifying the problem. The bulk of the time will be required for testing and validation and a project report will be provided detailing the work undertaken with all user operation being placed within the Nektar++ documentation.