IMPLEMENTATION OF LOW AND HIGH ORDER PRECONDITIONING STRATEGIES IN NEKTAR++

Duration: 6 months, Researcher: Andrew Comerford

OUTLINE:

Nektar++ is a high order spectral/hp element code with solvers primarily developed for fluid mechanics simulations. Typically, discretisation of their mathematical description leads to linear algebra systems that are only realisable using iterative methods. In such problems, the conditioning of the resulting matrix system necessitates the use of preconditioning techniques in order to accelerate the rate of convergence. However, unlike conventional linear finite element methods, high-order discretisations require specialised preconditioning methods; in particular care must be taken with respect to the scalability of h and p. The project focuses on extending the preconditioning strategies already available in Nektar++ to utilise the PETSc library as well as the implementation of preconditioning for pyramidal elements.

PROJECT OBJECTIVES:

Scalable low order preconditioning: Currently, it is known that a bottleneck in the preconditioning for large-scale applications (e.g. Flow over the front wing of a formula one car) is the treatment of the linear space preconditioning; in particular it does not perform suitably well for a large number of cores. PETSc is a library of scalable routines intended for large-scale applications. The library has many advantageous features that can be potentially exploited by Nektar++ for preconditioning the low order space in a manner that scales well on massively parallel systems.

High order preconditioning for pyramidal elements: Pyramidal elements have the ability to make meshing strategies simpler. Although they typically represent a small proportion of the final mesh they allow a key change in the design of the mesh, that is, the use of hexahedral elements together with prismatic boundary layers. Hexahedral elements are desirable for the final mesh structure as the reduced connectivity and hence matrix assembly is simplified. Additionally, in Formula One applications it is attractive to have regions of structured elements i.e. hexahedral elements that blend out to unstructured elements. This transition is only possible with pyramidal elements. To facilitate the above, high preconditioning of the already existing pyramidal element in Nektar++ will be implemented.

Variable polynomial order preconditioning: Variable polynomial order preconditioning is necessary in order to exploit features of the Nektar++ framework. For example in high Reynolds number applications (e.g. Formula

One) the boundary layer is incredibly small. The ability to vary the polynomial order to capture this very fine layer is often numerically advantageous. Adapting the existing high-order preconditioning will be undertaken to realise the above.

The project will benefit from my expertise as I implemented the existing preconditioning strategies in Nektar++. During my postdoctoral fellowship I have worked extensively on the code. Therefore I am in an excellent position to achieve the above outline project goals in a fast and efficient manner. This will be a very valuable contribution to the code and will open up the opportunity to obtain solutions on meshes optimised for a specific application on massively parallel systems.

ALIGNMENT WITH PRISM STRATEGY:

Development of key staff: A core part of my research interests lie within the realm of Nektar++. Currently I am in active pursuit of an academic position and Nektar++ forms a central part of my research plan and direction. This will be beneficial for the dissemination of the Nektar++ framework into other fields and institutions. My expertise in the area of preconditioning in the Nektar++ framework will allow me to achieve the desired goals of the proposal in the 6-month timeframe.

Collaboration other PRISM projects: The PETSc library is utilised widely by researchers in the PRISM project. Bringing this library into Nektar++ means it will be possible to share expertise and experience with this library among the different PRISM projects.

Longer-term research: The implementation of pyramid preconditioning will allow meshes of better design (e.g. Hexahedral dominant structure) to be utilised and solved for in a computationally efficient manner. The PETSc preconditioning will allow the low order preconditioning to scale well onto massively parallel systems.

BRIEF WORKPLAN: As I am already familiar with Nektar++, I will be able to begin implementation of the specific tasks immediately. The first phase will involve adapting the linear space preconditioner to use the AMG-preconditioning framework available in PETSc – this library is already linked into Nektar++. The second phase, will see the inclusion of pyramid element preconditioning into the Nektar++ and tested on some large-scale geometries (e.g. formula front wings) in which the mesh is comprised on hexahedral, prismatic and pyramidal elements. The scalability of the low order AMG-preconditioning with the high order preconditioning will be tested on this mesh The final phase will see the implementation of variable polynomial order preconditioning and testing of this implementation for a large Reynolds number application.