Computer Physics Communications 192 (2015) 205–219

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc

Nektar++: An open-source spectral/*hp* element framework☆

C.D. Cantwell[a,*], D. Moxey[a], A. Comerford[a], A. Bolis[a], G. Rocco[a], G. Mengaldo[a], D. De Grazia[a], S. Yakovlev[b], J.-E. Lombard[a], D. Ekelschot[a], B. Jordi[a], H. Xu[a], Y. Mohamied[a], C. Eskilsson[c], B. Nelson[b], P. Vos[a], C. Biotto[a], R.M. Kirby[b], S.J. Sherwin[a]

[a] *Department of Aeronautics, Imperial College London, London, UK*

[b] *School of Computing and Scientific Computing and Imaging (SCI) Institute, University of Utah, Salt Lake City, UT, USA*

[c] *Department of Shipping and Marine Technology, Chalmers University of Technology, Gothenburg, Sweden*

ARTICLE INFO

Article history:
Received 22 September 2014
Received in revised form
23 January 2015
Accepted 13 February 2015
Available online 24 February 2015

Keywords:
High-order finite elements
Spectral/*hp* elements
Continuous Galerkin method
Discontinuous Galerkin method
FEM

ABSTRACT

Nektar++ is an open-source software framework designed to support the development of high-performance scalable solvers for partial differential equations using the spectral/*hp* element method. High-order methods are gaining prominence in several engineering and biomedical applications due to their improved accuracy over low-order techniques at reduced computational cost for a given number of degrees of freedom. However, their proliferation is often limited by their complexity, which makes these methods challenging to implement and use. *Nektar++* is an initiative to overcome this limitation by encapsulating the mathematical complexities of the underlying method within an efficient C++ framework, making the techniques more accessible to the broader scientific and industrial communities. The software supports a variety of discretisation techniques and implementation strategies, supporting methods research as well as application-focused computation, and the multi-layered structure of the framework allows the user to embrace as much or as little of the complexity as they need. The libraries capture the mathematical constructs of spectral/*hp* element methods, while the associated collection of pre-written PDE solvers provides out-of-the-box application-level functionality and a template for users who wish to develop solutions for addressing questions in their own scientific domains.

Program summary

Program title: Nektar++

Catalogue identifier: AEVV_v1_0

Program summary URL: http://cpc.cs.qub.ac.uk/summaries/AEVV_v1_0.html

Program obtainable from: CPC Program Library, Queen's University, Belfast, N. Ireland

Licensing provisions: MIT

No. of lines in distributed program, including test data, etc.: 1052456

No. of bytes in distributed program, including test data, etc.: 42851367

Distribution format: tar.gz

Programming language: C++.

Computer: Any PC workstation or cluster.

Operating system: Linux/UNIX, OS X, Microsoft Windows.

RAM: 512 MB

Classification: 12.

External routines: Boost, FFTW, MPI, BLAS, LAPACK and METIS (www.cs.umn.edu)

☆ This paper and its associated computer program are available via the Computer Physics Communication homepage on ScienceDirect (http://www.sciencedirect.com/science/journal/00104655).

* Corresponding author.
E-mail address: c.cantwell@imperial.ac.uk (C.D. Cantwell).

http://dx.doi.org/10.1016/j.cpc.2015.02.008

Nature of problem:
The Nektar++ framework is designed to enable the discretisation and solution of time-independent or time-dependent partial differential equations.

Solution method:
Spectral/*hp* element method

Running time:
The tests provided take a few minutes to run. Runtime in general depends on mesh size and total integration time.

1. Introduction

Finite element methods (FEM) are commonplace among a wide range of engineering and biomedical disciplines for the solution of partial differential equations (PDEs) on complex geometries. However, low-order formulations often struggle to capture certain complex solution characteristics without the use of excessive mesh refinement due to numerical dissipation. In contrast, spectral techniques offer improved numerical characteristics, but are typically restricted to relatively simple regular domains.

High-order finite element methods, such as the traditional spectral element method [1], the p-type method [2] and the more recent spectral/*hp* element method [3], exhibit the convergence properties of spectral methods while retaining the geometric flexibility of traditional linear FEM. They potentially offer greater efficiency on modern CPU architectures as well as more exotic platforms such as many-core general-purpose graphics processing units (GPGPUs). The data structures which arise from using high-order methods are more compact and localised than their linear finite element counterparts, for a fixed number of degrees of freedom, providing increased cache coherency and reduced memory accesses, which is increasingly the primary bottleneck of modern computer systems.

The methods have had greatest prominence in the structural mechanics community and subsequently the academic fluid dynamics community. They are also showing promise in other areas of engineering, biomedical and environmental research. The most common concern cited with respect to using high-order finite element techniques outside of academia is the implementational complexity, stemming from the complex data structures, necessary to produce a computationally efficient implementation. This is a considerable hurdle which has limited their widespread uptake in many application domains and industries.

Nektar++ is a cross-platform spectral/*hp* element framework which aims to make high-order finite element methods accessible to the broader community. This is achieved by providing a structured hierarchy of C++ components, encapsulating the complexities of these methods, which can be readily applied to a range of application areas. These components are distributed in the form of cross-platform software libraries, enabling rapid development of solvers for use in a wide variety of computing environments. The code accommodates both small research problems, suitable for desktop computers, and large-scale industrial simulations, requiring modern HPC infrastructure, where there is a need to maintain efficiency and scalability up to many thousands of processor cores.

A number of software packages already exist for fluid dynamics which implement high-order finite element methods, although these packages are typically targeted at a specific domain or provide limited high-order capabilities as an extension. The *Nektar flow solver* is the predecessor to Nektar++ and implements the spectral/*hp* element method for solving the incompressible and compressible Navier–Stokes equations in both 2D and 3D. While it is widely used and the implementation is computationally efficient on small parallel problems, achieving scaling on large HPC clusters is challenging. Semtex [4] implements the 2D spectral element method coupled with a Fourier expansion in the third direction. The implementation is highly efficient, but can only be parallelised through Fourier-mode decomposition. Nek5000 [5] is a 3D spectral element code, based on hexahedral elements, which has been used for massively parallel simulations up to 300,000 cores. Hermes [6] implements hp-FEM for two-dimensional problems and has been used in a number of application areas. Limited high-order finite element capabilities are also included in a number of general purpose PDE frameworks including the DUNE project [7] and deal.II [8]. A number of codes also implement high-order finite element methods on GPGPUs including nudg++, which implements a nodal discontinuous Galerkin scheme [9], and PyFR [10], which supports a range of flux reconstruction techniques.

Nektar++ provides a single codebase with the following key features:

- Arbitrary-order spectral/*hp* element discretisations in one, two and three dimensions;
- Support for variable polynomial order in space and heterogeneous polynomial order within two- and three-dimensional elements;
- High-order representation of the geometry;
- Continuous Galerkin, discontinuous Galerkin and hybridised discontinuous Galerkin projections;
- Support for a Fourier extension of the spectral element mesh;
- Support for a range of linear solvers and preconditioners;
- Multiple implementation strategies for achieving linear algebra performance on a range of platforms;
- Efficient parallel communication using MPI showing strong scaling up to 2048-cores on Archer, the UK national HPC system;
- A range of time integration schemes implemented using generalised linear methods; and
- Cross-platform support for Linux, OS X and Windows operating systems.

In addition to the core functionality listed above, Nektar++ includes a number of solvers covering a range of application areas. A range of pre-processing and post-processing utilities are also included with support for popular mesh and visualisation formats, and an extensive test suite ensures the robustness of the core functionality.

The purpose of this paper is to expose the novel aspects of the code and document the structure of the library. We illustrate its use through a broad range of example applications which should enable other scientists to build on and extend *Nektar++* for use in their own applications. We begin by outlining the mathematical formulation of the spectral/*hp* element method and discuss the implementation of the framework. We then present a number