

# **Spectral Element Method**

*Support Modules*

Quick Reference Manual

Joseph Schoonover



# Contents

<b>Contents</b>	<b>1</b>
<b>1 Purpose and Overview</b>	<b>3</b>
<b>2 Modules</b>	<b>5</b>
2.1 Interpolation (interp) . . . . .	5
2.2 Spectral Operator Storage (nodal) . . . . .	5
2.3 Solution Storage (dgsem/cgsem) . . . . .	5
2.4 Iterative Solvers (iterativesolve) . . . . .	5
2.5 Geometry (geom) . . . . .	5
2.6 Filters (filters) . . . . .	5
2.7 Provided High-End Solvers (highend) . . . . .	5



# 1 Software Overview

The Spectral Element Libraries in Fortran (SELF) provide supporting data-structures for implementing spectral element methods (SEMs) to solve partial differential equations (PDEs) in multiple dimensions. The focus of the libraries is particularly on Legendre-Galerkin type methods. Legendre-Galerkin methods solve PDEs by approximating the solution and geometry by interpolating polynomials. Discrete equations are formed by solving the weak form of the PDE in which the integrals are approximated by discrete Gauss or Gauss-Lobatto quadrature. Underneath the umbrella of “Legendre-Galerkin” are Continuous Galerkin (CG) and Discontinuous Galerkin (DG) methods. CG methods are primarily used for elliptic or parabolic PDEs while DG methods are focused on hyperbolic systems.

The software is broken into the following components :

1. Commonly used routines and dictionaries,
2. Interpolation and Quadrature,
3. Spectral Operator storage structures,
4. Spectral filters,
5. Solution storage data structures,
6. Geometry,
7. High end solvers for particular PDEs

This first chapter is focused on getting you working with the high end solvers. The high end solvers follow two generic structures, one for CG methods and another for DG methods.

Currently, the DG method templates are available for one, two, and three dimensions while CG is only available in one and two dimensions. Underneath the `src/highend/` directory, you will see a set of subdirectories for each of the provided high-end solvers. The `examples/` directory is where you will find test-cases of each of the high-end solvers that are “ready-to-go”.

Underneath a given example, you will find a `build/` directory, `localmods/` directory, and a `run/` directory.

## 2 Modules

This chapter describes in more detail the modules that are provided in each sub-directory, including a description of the data-structures and associated procedures. This provides the necessary information for developing your own “highend” solvers from scratch. Coincidentally, a few of the highend solvers are described here to provide examples of how the support libraries can be used to solve PDE’s on complex geometry.

### 2.1 Interpolation (interp)

### 2.2 Spectral Operator Storage (nodal)

### 2.3 Solution Storage (dgsem/cgsem)

### 2.4 Iterative Solvers (iterativesolve)

### 2.5 Geometry (geom)

### 2.6 Filters (filters)

### 2.7 Provided High-End Solvers (highend)