Using Sum Factorization and Spectral Methods in Finite Element Methods

Ammon Washburn, Andrew Gillette

University of Arizona

05/06/2015
Idea behind Finite Element method (The Galerkin Method)
Transition from $h$ FEM to $hp$ spectral FEM
Number of evaluations increase dramatically with $hp$ methods
Sum factorization and spectral methods improves the constant dramatically
FEMs are a subset of a type of numerical method named Galerkin Method

- We want to find \( u \in V \) satisfying \( a(u, v) = f(v) \forall v \in V \) where \( f \) is a functional and \( a \) is bilinear
- If you have an infinite dimensional Hilbert space \( V \) then consider a finite-dimensional sub-space \( V_n \)
- Find an approximate answer \( u_n \) using the basis of the finite dimensional subspace
- Enough to find \( u_n \) so that \( a(u, e_i) = f(e_i) \) for every basis element \( e_i \in V_n \)
Galerkin Method

Properties of your approximate solution \( u_n \in V_n \)

- Because \( a(\cdot, \cdot) \) is bilinear, we are guaranteed to have a solution and that solution is unique.
- After we find the solution we are guaranteed this will be the "best" solution in the subspace.
- Is your subspace the best?

Figure 1: How close is your solution?
FEMs as a Galerkin Method

- Turn your differential equation into the bilinear form by finding the distributional solution
- Your $V_n$ is the space of all piece-wise defined polynomials of degree $n$ defined on your elements
- Find $u_n$ so $a(u_n, e_i) = f(e_i)$ for all basis functions $e_i$
- Use $u_n$ as your solution

Example from Melenk et al.

Say we have the following differential equation on $\Omega \subset \mathbb{R}^2$

$$-\nabla \cdot (A(x)\nabla u) = l(x), \ u = 0 \ on \ \partial \Omega$$

Then the weak form of the solution would be $a(u, v) = f(v) \ \forall v \in V$

$$a(u, v) = \int_{\Omega} \nabla u \cdot (A(x)\nabla v) \, dx \, dy, \quad f(v) = \int_{\Omega} l(x)v \, dx \, dy$$
Calculating your Approximation

- You generate local stiffness matrices for each element
- Compute a global stiffness matrix by assembling the local stiffness matrices
- Solve the global system for your approximation
- Generating local stiffness matrices require quadrature formulas to compute the functionals

\[
\begin{bmatrix}
K^{(1)} & = & \begin{bmatrix}
    k_{11}^{(1)} & k_{12}^{(1)} & 0 & 0 \\
    k_{21}^{(1)} & k_{22}^{(1)} & 0 & 0 \\
    0 & 0 & 0 & 0 \\
    0 & 0 & 0 & 0
\end{bmatrix} & \quad K^{(2)} & = & \begin{bmatrix}
    0 & 0 & 0 & 0 \\
    0 & k_{22}^{(2)} & k_{23}^{(2)} & 0 \\
    0 & k_{32}^{(2)} & k_{33}^{(2)} & 0 \\
    0 & 0 & 0 & 0
\end{bmatrix} \\
K^{(3)} & = & \begin{bmatrix}
    0 & 0 & 0 & 0 \\
    0 & k_{22}^{(3)} & k_{23}^{(3)} & 0 \\
    0 & k_{32}^{(3)} & k_{33}^{(3)} & 0 \\
    0 & 0 & 0 & 0
\end{bmatrix} & \quad K^{(4)} & = & \begin{bmatrix}
    0 & 0 & 0 & 0 \\
    0 & k_{33}^{(4)} & k_{34}^{(4)} & 0 \\
    0 & 0 & k_{43}^{(4)} & k_{44}^{(4)} \\
    0 & 0 & k_{43}^{(4)} & k_{44}^{(4)}
\end{bmatrix}
\]

Figure 2: Assembling your local stiffness matrices source: US Patent US 20020042663 A1
The original FEMs and most widely used are the h FEMs. There was also developed spectral FEMs and the p FEMs. Both methods have very good algorithms to calculate them but both have only algebraic convergence. Combine the strengths of both to achieve geometric convergence. Use Sum Factorization and some of the spectral methods to achieve optimal complexity.
Properties that allow for Fast Quadrature

Since we want to numerically calculate these FEMs, we need to use quadrature formulas to (exactly) approximate these polynomials. We can capitalize on this fact if we have two properties

- Tensorial product construction of polynomials
- Shape functions that are adapted to the quadrature scheme

In the end we are just calculating sum of sums of functions evaluated at certain points. Use that.
Fast Quadrature for hp spectral FEMs

Typically for two dimensions we are calculating a double summation that has this form:

$$\sum_{s=1}^{q_0} \sum_{t=1}^{q_1} \Phi_i^1(x_s) \Phi_j^2(y_t) \Psi_k^1(x_s) \Psi_l^2(y_t) w_s w_t b(x_s, y_t)$$

Where $\Phi$ and $\Psi$ come from our basis functions, $w_s$ and $w_t$ are weights associated with a quadrature formula and $b(x_s, y_t)$ is a map from a master element to the element we are currently working on. $i, j, k$ and $l$ index all the one dimensional basis functions we are working with. This requires $n^4 q_0 q_1$ number of operations!
Factor out certain things from the first sum and you get.

\[ \sum_{s=1}^{q_0} \Phi_1^1(x_s) \psi_1^1(x_s) \sum_{t=1}^{q_1} \Phi_2^2(y_t) \psi_2^2(y_t) w_s w_t b(x_s, y_t) \]

- This only takes \( n^2 q_1 (n^2 + q_0) \) operations.
- Very easy to do but hard to set up the basis
- We can employ sum factorization with simplicial elements by using Bernstein polynomials and Duffy transformation (Ainsworth et al.)
Consider again

\[
\begin{aligned}
q_0 \sum_{s=1}^{q_0} \Phi_1^1(x_s) \Psi_1^1(x_s) \sum_{t=1}^{q_1} \Phi_2^2(y_t) \Psi_2^1(x_s) \Psi_2^1(y_t) w_s w_t b(x_s, y_t)
\end{aligned}
\]

- What if most of \( \Phi_1^1(x_s) \Phi_2^2(y_t) \Psi_1^1(x_s) \Psi_2^1(y_t) \) was zero?
- Then the operation count would be \( n^2 q_1^* (n^2 + q_0^*) \) where \( q_1^*, q_0^* \ll q_1, q_0 \)
- It requires about fifty function evaluations to exactly approximate the integral of a hundred degree polynomial
- With classical spectral methods it would require one point
Basic idea on how to do it

- You have high degree polynomials in the interior which require variants of large Gaussian quadrature
- You allow for overintegration to make the method stable
- Write your basis function of Lagrange interpolants $l_i$ based on the points of Gaussian quadrature of $p + q$
- $l_i(\xi_j) = 0$ for all but $q + 1$ points where $\xi_j$ is the $j^{th}$ node in Gaussian quadrature of $p + q$ points
Considerations

Figure 3: Math is frustrating that way sometimes source: http://theinkedsage.com

We want a basis to
- Have some sort of tensorial construction
- Be adapted to our quadrature scheme
- Have the piece-wise polynomials agree on the boundaries so our solution is continuous
- be close enough to the true solution