Research Problems in Finite Element Theory: Analysis, Geometry, and Application

Andrew Gillette

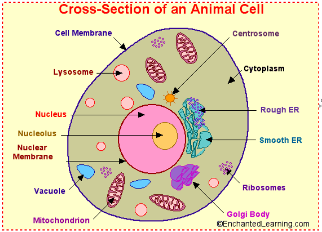
Department of Mathematics
University of Arizona

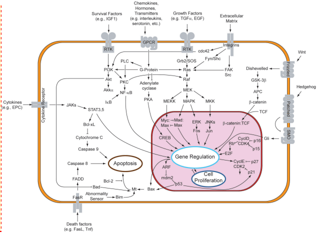
Research Tutorial Group Presentation

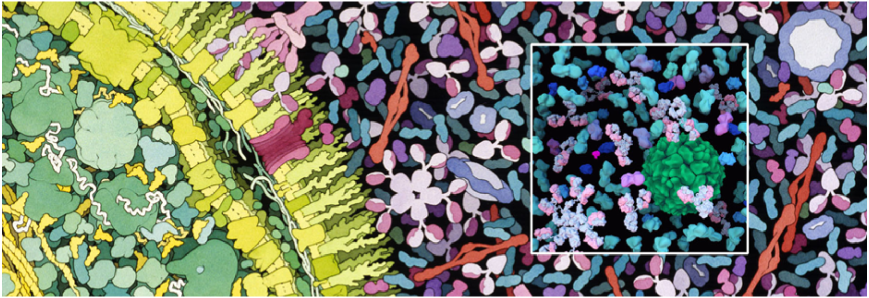
Slides and more info at:

http://math.arizona.edu/∼agillette/

What’s relevant in molecular modeling?

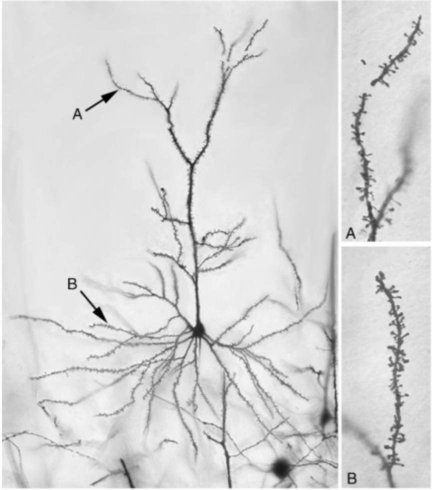


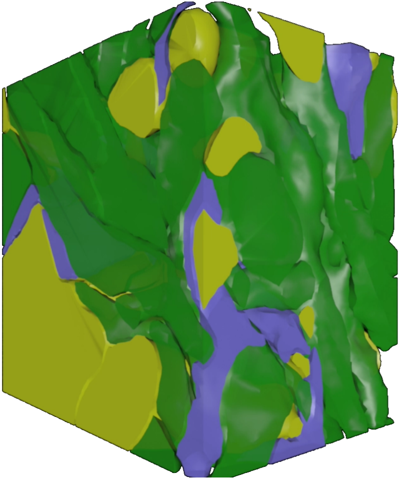




(bottom image: David Goodsell)

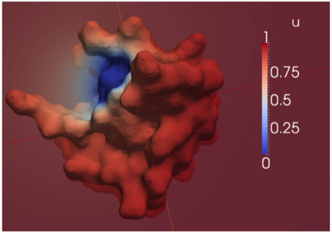
What’s relevant in neuronal modeling?

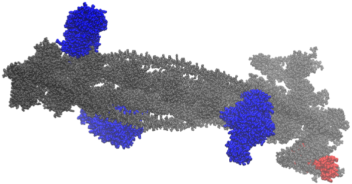


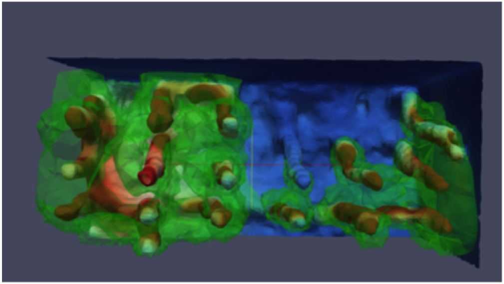


(right image: Chandrajit Bajaj)

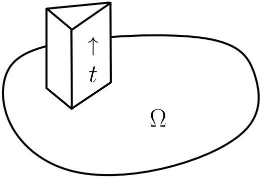
What's relevant in diffusion modeling?

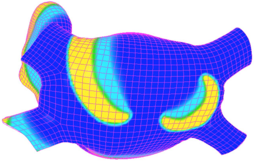






Mathematics used in biological models





$$\mathbb{A}\,\vec{\mathbf{x}} = \vec{\mathbf{b}}$$

Analysis

PDEs

Geometry

Topology

Combinatorics

Linear algebra

Numerical analysis

Mathematics helps answer distinguish relevant and irrelevant features of a model:

- Does the PDE have a unique solution, bounded in some norm?
- Does the domain discretization affect the quality of the approximate solution?
- Is the solution method optimally efficient? (e.g. Why isn't my code working?)

Focus of my research in these areas: the **Finite Element Method**

Table of Contents

Outline

1. Introduction to the Finite Element Method
2. Tensor product finite element methods
3. The minimal approximation question
4. Serendipity finite element methods
5. RTG Project Ideas

The Finite Element Method: 1D

The **finite element method** is a way to numerically approximate the solution to PDEs.

(Example worked out on board)

Ex: The 1D Laplace equation: find $u(x) \in U$ (dim $U = \infty$) s.t.

$$\begin{cases} -u''(x) = f(x) & \text{on } [a, b] \\ u(a) = 0, \\ u(b) = 0 \end{cases}$$

Weak form: find $u(x) \in U$ (dim $U = \infty$) s.t.

$$\int_a^b u'(x)v'(x)\,dx = \int_a^b f(x)v(x)\,dx, \quad \forall v \in V \quad (\dim V = \infty)$$

Discrete form: find $u_h(x) \in U_h$ (dim $U_h < \infty$) s.t.

$$\int_a^b u_h'(x)v_h'(x)\,dx = \int_a^b f(x)v_h(x)\,dx, \quad \forall v_h \in V_h \quad (\dim V_h < \infty)$$

The Finite Element Method: 1D

(Example worked out on board)

Suppose $u_h(x)$ can be written as linear combination of V_h elements:

$$u_h(x) = \sum_{v_i \in V_h} u_i v_i(x)$$

The discrete form becomes: find coefficients $u_i \in \mathbb{R}$ such that

$$\sum_i \int_a^b u_i v_i'(x) v_j'(x)\ dx = \int_a^b f(x) v_j(x)\ \ dx, \quad \forall v_h \in V_h \quad (\dim V_h < \infty)$$

Written as a linear system:

$$[\ \mathbb{A}\ ]_{ji}\ [\ u\ ]_i = [\ f\ ]_j\,, \quad \forall v_j \in V_h$$

With some functional analysis we can prove: $\exists\ C > 0$, independent of h, s.t.

$$\underbrace{||u - u_h||_{H^1(\Omega)}}_{\substack{\text{error between cnts} \\ \text{and discrete solution}}} \quad \leq \quad \underbrace{C\, h\, |u|_{H^2(\Omega)}}_{\substack{\text{bound in terms of} \\ \text{2nd order osc. of } u}} \quad , \quad \underbrace{\forall u \in H^2(\Omega)}_{\substack{\text{holds for any } u \text{ with} \\ \text{bounded 2nd derivs.}}} \quad .$$

where h = maximum width of elements use in discretization.

Outline

Generalizing the 1st order, 1D method

Goal: Efficient, accurate approximation of the solution to a PDE over $\Omega \subset \mathbb{R}^n$ for arbitrary dimension n and arbitrary rate of convergence r.

Standard $O(h^r)$ **tensor product** finite element method in $\mathbb{R}^n$:

$\rightarrow$ Mesh Ω by n-dimensional cubes of side length h.

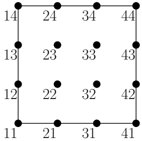
$\rightarrow$ Set up a linear system involving $(r+1)^n$ degrees of freedom (DoFs) per cube.

$\rightarrow$ For unknown continuous solution u and computed discrete approximation u_h:

$$\underbrace{||u - u_h||_{H^1(\Omega)}}_{\text{approximation error}} \leq \underbrace{C\, h^r\, |u|_{H^{r+1}(\Omega)}}_{\text{optimal error bound}}, \qquad \forall u \in H^{r+1}(\Omega).$$

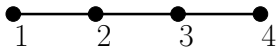
Implementation requires a clear characterization of the isomorphisms:

$$\left\{ \begin{array}{c} x^r y^s \\ 0 \leq r, s \leq 3 \end{array} \right\} \longleftrightarrow \left\{ \begin{array}{c} \psi_i(x)\psi_j(y) \\ 1 \leq i, j \leq 4 \end{array} \right\} \longleftrightarrow$$



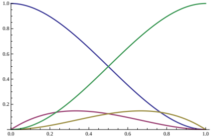
monomials $\longleftrightarrow$ **basis functions** $\longleftrightarrow$ **domain points**

Cubic Hermite Geometric Decomposition (1D, r=3)

monomials	approxim.	basis functions	geometry	domain points
$\{1, x, x^2, x^3\}$	$\longleftrightarrow$	$\{\psi_1, \psi_2, \psi_3, \psi_4\}$	$\longleftrightarrow$	

Cubic Hermite Basis on $[0, 1]$

$$\begin{bmatrix}\psi_1\\ \psi_2\\ \psi_3\\ \psi_4\end{bmatrix} := \begin{bmatrix} 1-3x^2+2x^3 \\ x-2x^2+x^3 \\ x^2-x^3 \\ 3x^2-2x^3\end{bmatrix}$$



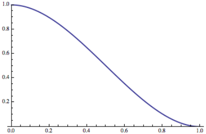
Approximation: $x^r = \sum_{i=1}^{4} \varepsilon_{r,i}\psi_i$, for $r = 0, 1, 2, 3$, where $[\varepsilon_{r,i}] = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 1 \\ 0 & 0 & -2 & 1 \\ 0 & 0 & -3 & 1 \end{pmatrix}$

Geometry: If $a(x)$ is a cubic polynomial then:

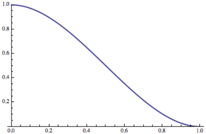
$$a(x) = \underbrace{a(0)}_{\text{value}}\psi_1 + \underbrace{a'(0)}_{\text{derivative}}\psi_2 - \underbrace{a'(1)}_{\text{derivative}}\psi_3 + \underbrace{a(1)}_{\text{value}}\psi_4$$

Tensor Product Polynomials

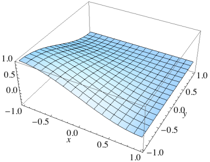
We can use our 1D Hermite functions to make 2D Hermite functions:



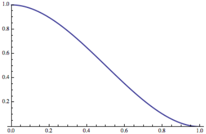
$\times$



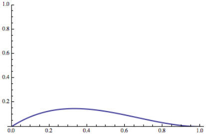
$=$



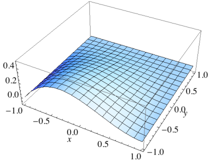
$$\psi_1(x) \times \psi_1(y) = \psi_{11}(x, y)$$



$\times$



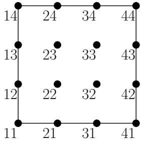
$=$



$$\psi_1(x) \times \psi_2(y) = \psi_{12}(x, y)$$

Cubic Hermite Geometric Decomposition (2D, r=3)

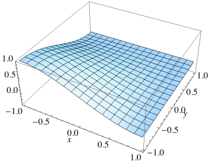
$$\left\{ \begin{array}{c} x^r y^s \\ 0 \le r, s \le 3 \end{array} \right\} \quad \longleftrightarrow \quad \left\{ \begin{array}{c} \psi_i(x)\psi_j(y) \\ 1 \le i, j \le 4 \end{array} \right\} \quad \longleftrightarrow$$



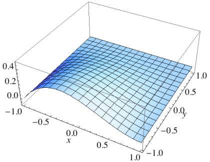
monomials $\longleftrightarrow$ **basis functions** $\longleftrightarrow$ **domain points**

Approximation: $x^r y^s = \displaystyle\sum_{i=1}^{4} \sum_{j=1}^{4} \varepsilon_{r,i}\varepsilon_{s,j}\psi_{ij}$, for $0 \le r, s \le 3$, $\varepsilon_{r,i}$ as in 1D.

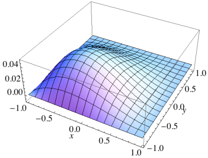
Geometry:



ψ_{11}



ψ_{21}

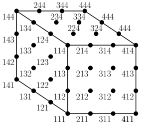


ψ_{22}

$$a(x,y) = \underbrace{a|_{(0,0)}}_{\text{value}} \psi_{11} + \underbrace{\partial_x a|_{(0,0)}}_{\text{1st deriv.}} \psi_{21} + \underbrace{\partial_y a|_{(0,0)}}_{\text{1st deriv.}} \psi_{12} + \underbrace{\partial_x \partial_y a|_{(0,0)}}_{\text{2nd deriv.}} \psi_{22} + \cdots$$

Cubic Hermite Geometric Decomposition (3D, r=3)

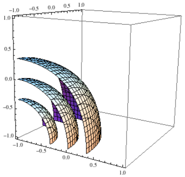
$$\left\{ \begin{array}{c} x^r y^s z^t \\ 0 \le r, s, t \le 3 \end{array} \right\} \longleftrightarrow \begin{array}{c} \psi_i(x)\psi_j(y)\psi_k(z) \\ 1 \le i, j, k \le 4 \end{array} \longleftrightarrow$$



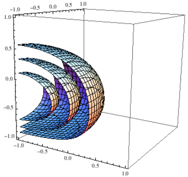
monomials $\longleftrightarrow$ **basis functions** $\longleftrightarrow$ **domain points**

Approximation: $x^r y^s z^t = \sum_{i=1}^{4} \sum_{j=1}^{4} \sum_{k=1}^{4} \varepsilon_{r,i}\varepsilon_{s,j}\varepsilon_{t,k}\psi_{ijk}$, for $0 \le r, s, t \le 3$, $\varepsilon_{r,i}$ as in 1D.

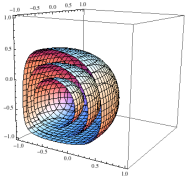
Geometry: Contours of level sets of the basis functions:



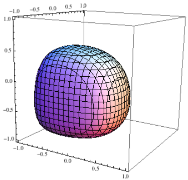
ψ_{111}



ψ_{112}

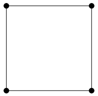
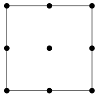
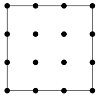
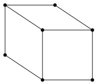
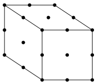
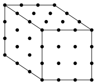


ψ_{212}



ψ_{222}

Tensor Product FEM Summary

$O(h)$	$O(h^2)$	$O(h^3)$	$O(h^r)$
$\left\{ \begin{array}{c} x^r y^s \\ r, s \leq 1 \end{array} \right\}$	$\left\{ \begin{array}{c} x^r y^s \\ r, s \leq 2 \end{array} \right\}$	$\left\{ \begin{array}{c} x^r y^s \\ r, s \leq 3 \end{array} \right\}$	$\left\{ \begin{array}{c} x^r y^s \\ r, s \leq r \end{array} \right\}$
			
4	9	16	$(r+1)^2$
$\left\{ \begin{array}{c} x^r y^s z^t \\ r, s, t \leq 1 \end{array} \right\}$	$\left\{ \begin{array}{c} x^r y^s z^t \\ r, s, t \leq 2 \end{array} \right\}$	$\left\{ \begin{array}{c} x^r y^s z^t \\ r, s, t \leq 3 \end{array} \right\}$	$\left\{ \begin{array}{c} x^r y^s z^t \\ r, s, t \leq r \end{array} \right\}$
			
8	27	64	$(r+1)^3$ $\leftarrow$ a lot!

Outline

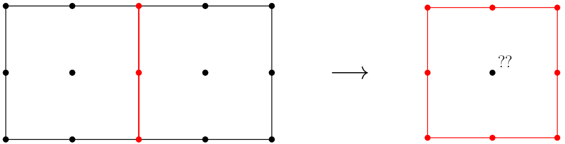
How many functions are *minimally* needed?

For unknown continuous solution u and computed discrete approximation u_h:

$$\underbrace{||u - u_h||_{H^1(\Omega)}}_{\text{approximation error}} \leq \underbrace{C\, h^r\, |u|_{H^{r+1}(\Omega)}}_{\text{optimal error bound}}, \quad \forall u \in H^{r+1}(\Omega).$$

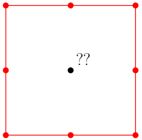
The proof of the above estimate relies on two properties of finite elements:

Continuity: Adjacent elements agree on order r polynomials their shared face



Approximation: Basis functions on each element span all degree r monomials

$$\underbrace{\{1, x, y, x^2, y^2, xy\}}_{\substack{\text{required for} \\ O(h^2)\text{ approximation}}} \longrightarrow$$



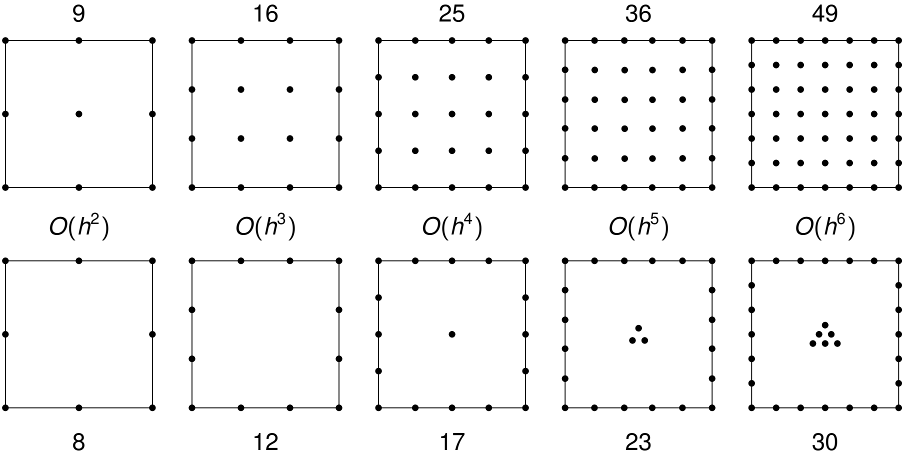
$$\longleftrightarrow \underbrace{\{1, x, y, x^2, y^2, xy, x^2y, xy^2, x^2y^2\}}_{\substack{\text{standard polynomials in} \\ O(h^2)\text{ tensor product method}}}$$

Next time...

- Characterization of the ‘minimal’ approximation question for any order
- Intriguing mathematical difficulties and recent ‘serendipitous’ solutions
- Benefits of serendipity solutions to biological modeling
- Open research problems for an RTG study

Outline

Serendipity Elements

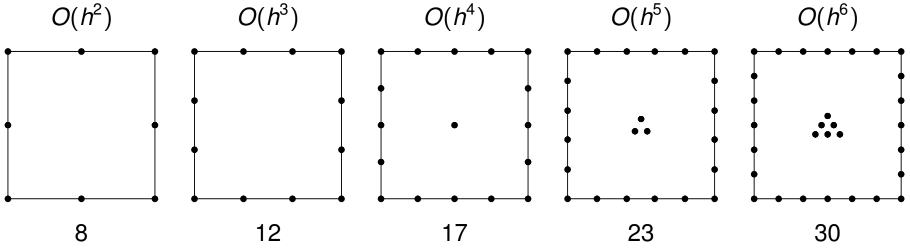


For $r \geq 4$ on squares:

$O(h^r)$ tensor product method :	$r^2 + 2r + 1$	dots
$O(h^r)$ serendipity method:	$\frac{1}{2}(r^2 + 3r + 6)$	dots

$$\underbrace{||u - u_h||_{H^1(\Omega)}}_{\text{approximation error}} \leq \underbrace{C\, h^r\, |u|_{H^{r+1}(\Omega)}}_{\text{optimal error bound}}, \quad \forall u \in H^{r+1}(\Omega).$$

Serendipity Elements

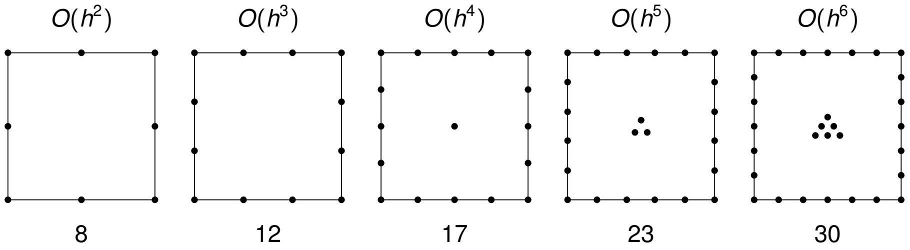


$\rightarrow$ Why $r + 1$ dots per edge?
Ensures continuity between adjacent elements.

$\rightarrow$ Why interior dots only for $r \geq 4$?
Consider, e.g. $p(x, y) := (1 + x)(1 - x)(1 - y)(1 + y)$
Observe p is a degree 4 polynomial but $p \equiv 0$ on $\partial([-1, 1]^2)$.

$\rightarrow$ How can we recover tensor product-like structure...
...without a tensor product structure?

Mathematical Challenges More Precisely

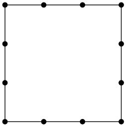


Goal: Construct basis functions for serendipity elements satisfying the following:

- **Symmetry:** Accommodate interior degrees of freedom that grow according to triangular numbers on square-shaped elements.
- **Tensor product structure:** Write as linear combinations of standard tensor product functions.
- **Hierarchical:** Generalize to methods on n-cubes for any $n \geq 2$, allowing restrictions to lower-dimensional faces.

Which monomials do we need?

$O(h^3)$ serendipity element:



$$\Big\{ \overbrace{1, x, y, x^2, y^2, xy, x^3, y^3, x^2y, xy^2}^{\substack{\text{total degree at most cubic} \\ \text{(req. for } O(h^3) \text{ approximation)}}}, x^3y, xy^3, x^2y^2, x^3y^2, x^2y^3, x^3y^3 \Big\}$$

$$\underbrace{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}_{\substack{\text{at most cubic in each variable} \\ \text{(used in } O(h^3) \text{ tensor product methods)}}}$$

We need an intermediate set of 12 monomials!

The **superlinear** degree of a polynomial ignores linearly-appearing variables.

Example: $\text{sldeg}(xy^3) = 3$, even though $\deg(xy^3) = 4$

Definition: $\text{sldeg}(x_1^{e_1} x_2^{e_2} \cdots x_n^{e_n}) := \left(\sum_{i=1}^{n} e_i \right) - \# \{ e_i \; : \; e_i = 1 \}$

$$\Big\{ \underbrace{1, x, y, x^2, y^2, xy, x^3, y^3, x^2y, xy^2, x^3y, xy^3}_{\text{superlinear degree at most 3 (dim=12)}}, x^2y^2, x^3y^2, x^2y^3, x^3y^3 \Big\}$$

ARNOLD, AWANOU *The serendipity family of finite elements*, Found. Comp. Math, 2011.

Superlinear polynomials form a lower set

Given a monomial $x^\alpha := x_1^{\alpha_1} \cdots x_d^{\alpha_d}$,

associate the multi-index of d non-negative integers $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_d) \in \mathbb{N}_0^d$.

Define the superlinear norm of α as $|\alpha|_{sprlin} := \sum_{\substack{j=1 \\ \alpha_j \geq 2}}^{d} \alpha_j$,

so that the superlinear multi indices are $S_r = \left\{ \alpha \in \mathbb{N}_0^d : |\alpha|_{sprlin} \leq r \right\}$.

Observe that S_r has a partial ordering $\mu \leq \alpha$ means $\mu_i \leq \alpha_i$.

Thus S_r is a **lower set**, meaning $\boxed{\alpha \in S_r,\ \mu \leq \alpha \implies \mu \in S_r}$

We can thus apply the following recent result.

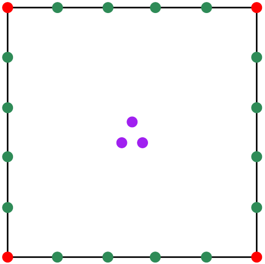
Theorem (Dyn and Floater, 2013)

Fix a lower set $L \subset \mathbb{N}_0^d$ and points $z_\alpha \in \mathbb{R}^d$ for all $\alpha \in L$. For any sufficiently smooth d-variate real function f, there is a unique polynomial $p \in \operatorname{span}\{x^\alpha : \alpha \in L\}$ that interpolates f at the points z_α, with partial derivative interpolation for repeated z_α values.

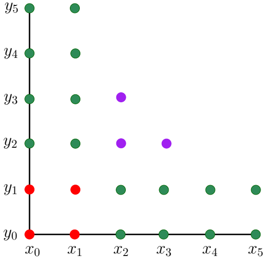
DYN AND FLOATER *Multivariate polynomial interpolation on lower sets*, J. Approx. Th., to appear.

Partitioning and reordering the multi-indices

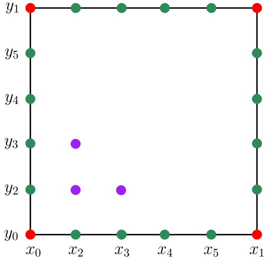
By a judicious choice of the interpolation points $z_\alpha = (x_i, y_j)$, we recover the dimensionality associations of the degrees of freedom of serendipity elements.



The order 5 serendipity element, with degrees of freedom color-coded by dimensionality.



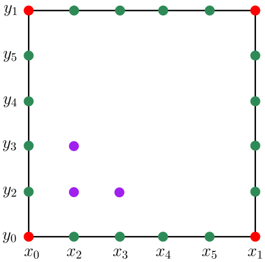
The lower set S_5, with equivalent color coding.



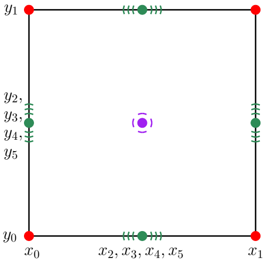
The lower set S_5, with domain points z_α reordered.

Symmetrizing the multi-indices

By collecting the re-ordered interpolation points $z_\alpha = (x_i, y_j)$, at midpoints of the associated face, we recover the dimensionality associations of the degrees of freedom of serendipity elements.



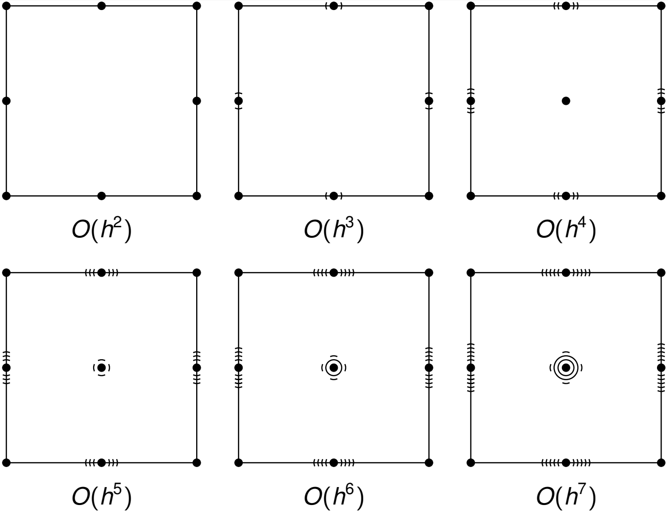
The lower set S_5, with domain points z_α reordered.



A symmetric reordering, with multiplicity. The associated interpolant recovers values at dots, three partial derivatives at edge midpoints, and two partial derivatives at the face midpoint.

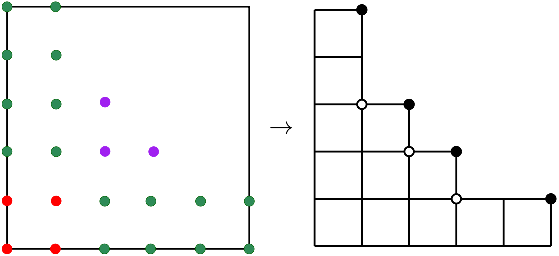
2D symmetric serendipity elements

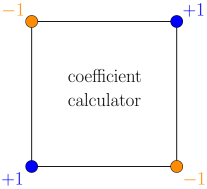
Symmetry: Accommodate interior degrees of freedom that grow according to triangular numbers on square-shaped elements.



Tensor product structure

The Dyn-Floater interpolation scheme is expressed in terms of tensor product interpolation over ‘maximal blocks’ in the set using an inclusion-exclusion formula.





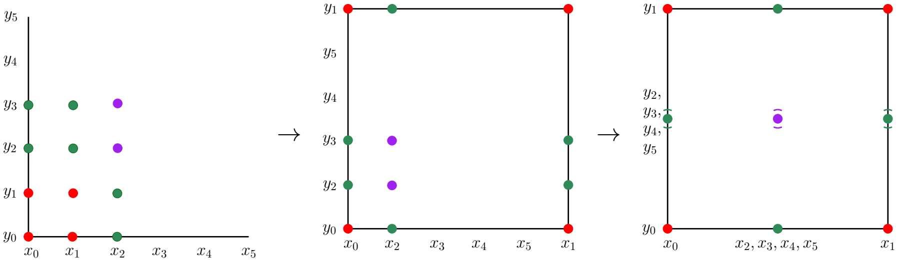
Put differently, the linear combination is the sum over *all* blocks within the lower set with coefficients determined as follows:

$\rightarrow$ Place the coefficient calculator at the extremal block corner.
$\rightarrow$ Add up all values appearing in the lower set.
$\rightarrow$ The coefficient for the block is the value of the sum.

Hence: black dots $\rightarrow$ +1;   white dots $\rightarrow$ -1;   others $\rightarrow$ 0.

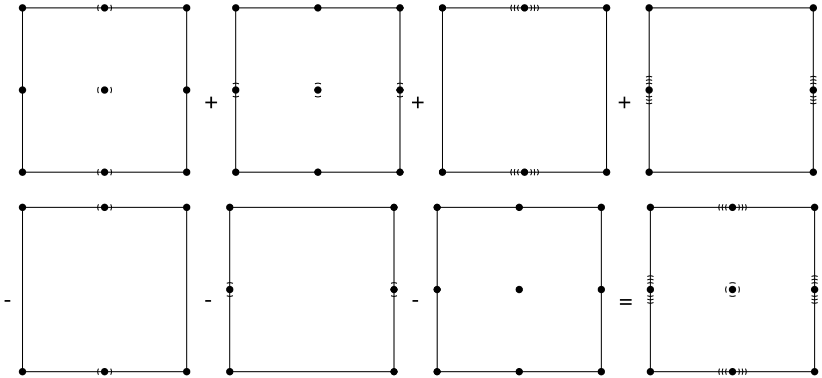
Tensor product structure

Thus, using our symmetric approach, each maximal block in the lower set becomes a standard tensor-product interpolant.



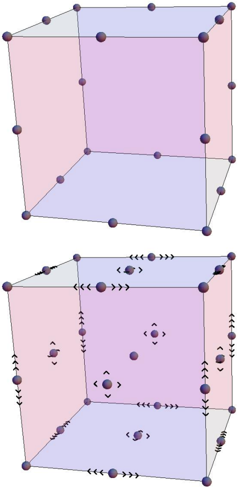
Linear combination of tensor products

Tensor product structure: Write basis functions as linear combinations of standard tensor product functions.



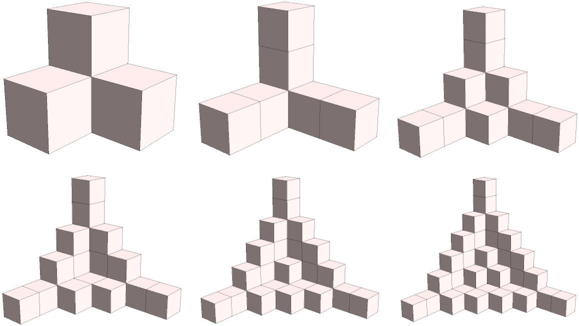
3D elements

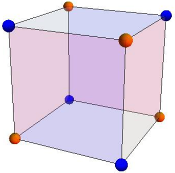
Hierarchical: Generalize to methods on n-cubes for any $n \geq 2$, allowing restrictions to lower-dimensional faces.



3d coefficient computation

Lower sets for superlinear polynomials in 3 variables:



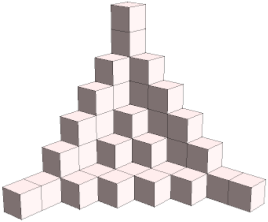


Decomposition into a linear combination of tensor product interpolants works the same as in 2D, using the 3D coefficient calculator at left. (Blue $\rightarrow$ +1;   Orange $\rightarrow$ -1).

FLOATER, GILLETTE *Nodal basis functions for the serendipity family of finite elements*, in preparation.

Brief aside: historical quiz

What video game is shown on the right?



Outline

RTG Project ideas

Email me if you'd like a copy of the slides with the project ideas.