Software Design for Non-conforming Finite Elements

Matthew Knepley and Tobin Isaac

Computational and Applied Mathematics
Rice University

Applied Modeling and Computation Group Seminar
Earth Science and Engineering
Imperial College, London March 17, 2017

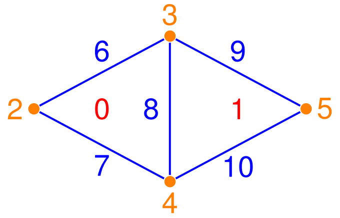
Main Point

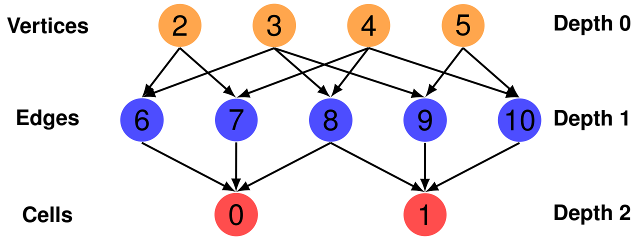
We support structured AMR with an unstructured interface efficiently.

https://arxiv.org/abs/1508.02470

Sample Meshes

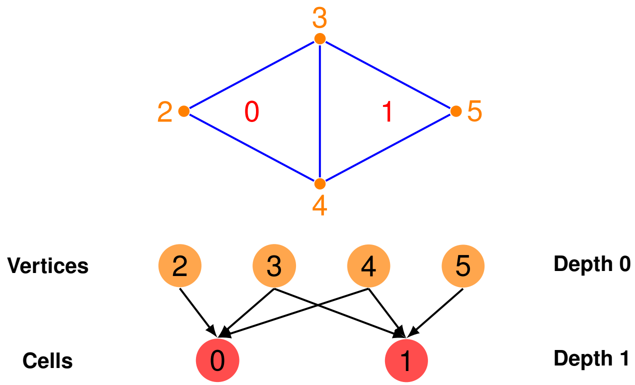
Interpolated triangular mesh





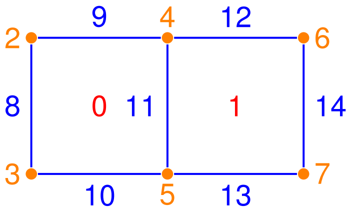
Sample Meshes

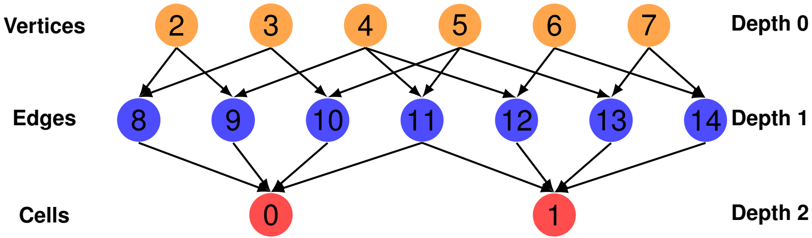
Optimized triangular mesh



Sample Meshes

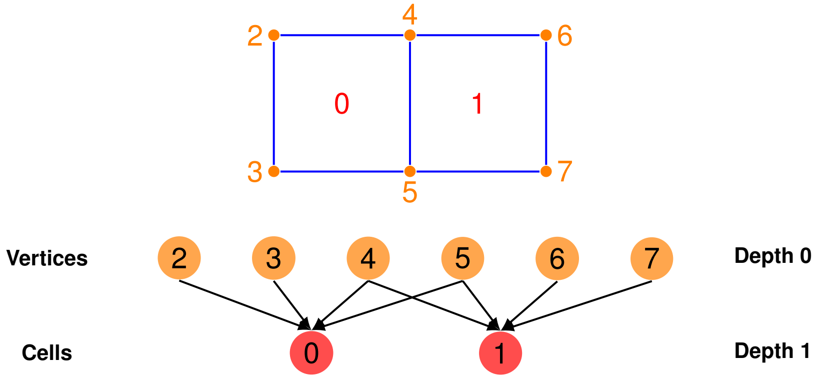
Interpolated quadrilateral mesh





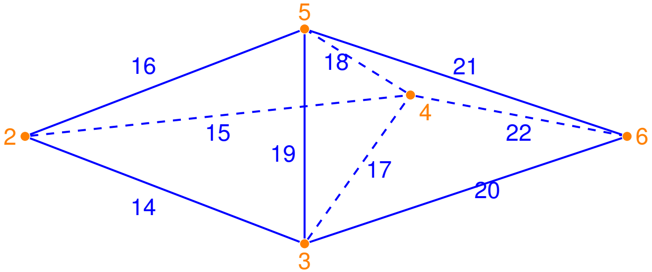
Sample Meshes

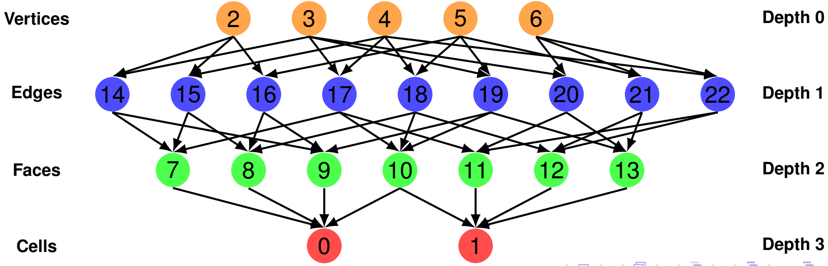
Optimized quadrilateral mesh



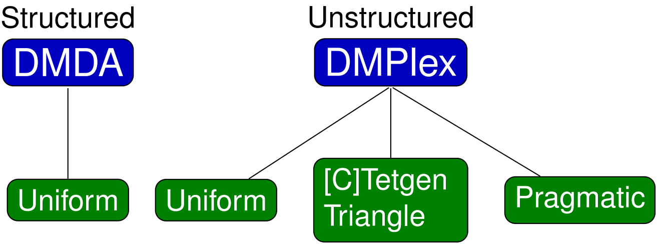
Sample Meshes

Interpolated tetrahedral mesh

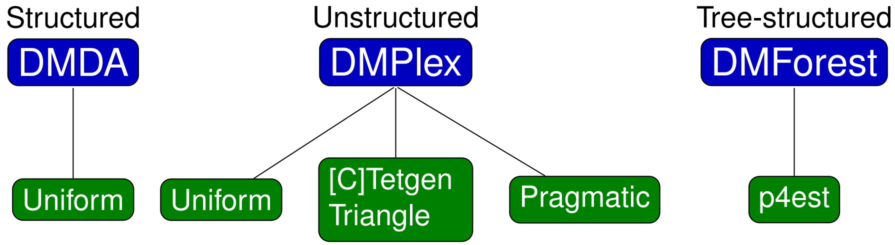




Mesh Refinement in PETSc



Mesh Refinement in PETSc

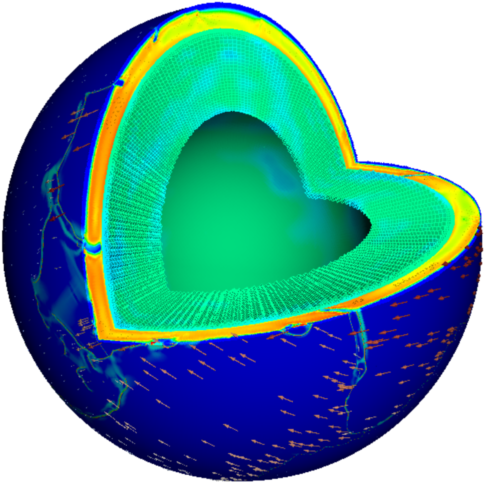


p4est overview

The p4est library (Carsten Burstedde and Toby Isaac) provides scalable AMR routines via a forest-of-octrees/quadtrees:

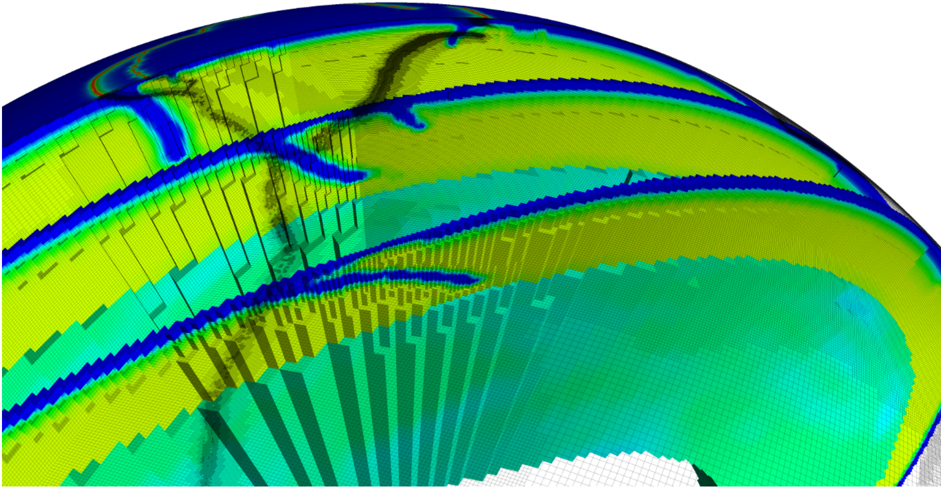
- a unstructured hexahedral mesh (“the forest”);
- where each hexahedron contains an arbitrarily refined octree;
- space-filling curve (SFC) orders elements;
- philosophy: as-simple-as-possible coarse mesh describes geometry, refinement captures all detail.
- not a framework: does not have numerical methods
  - Used for parallelism by Deal.II
  - Tight integration with solvers (e.g., multilevel) is still the domain of experts (next slide)

p4est in geophysics



(Rudi et al., 2015), "An extreme-scale implicit solver for complex PDEs: highly heterogeneous flow in earth's mantle," doi:10.1145/2807591.2807675.

p4est in geophysics



(Rudi et al., 2015), “An extreme-scale implicit solver for complex PDEs: highly heterogeneous flow in earth’s mantle,” doi:10.1145/2807591.2807675.

Outline

FEM Axioms

Three FEM axioms allow an element to be
computable in our framework,
meaning we can form
a global nodal basis W for the dual space V_h^*.

- Sparsity
- Matching
- Independence

Notation

P	Reference approximation (primal) space
Q	Reference measurement (dual) space
T	Reference Cell
S	Reference complex for T
P_i	Primal space on cell T_i
Q_i	Dual space on cell T_i

I. Sparsity

For each $\sigma_j \in Q$ there exists a point $p \in S$ such that,
if $\psi_k \in P(T)$ is σ_k's shape function,
meaning $\sigma_j(\psi_k) = \delta_{jk}$,
then $\mathrm{supp}(\psi_k) = \bigcup \mathrm{star}(p)$.

- Dual basis functions are *attached* to points in S
- Topological support describes function support
- Allows for compactly supported basis functions

Notation

φ_i^*	Pullback of T_i onto T, for H_1 $\varphi_i^* f = f \circ \varphi_i$
$\varphi_{*,i}$	Pushforward of T onto T_i, the adjoint of φ_i^*
$P(X)$	Trace space of $P(T)$ on $X \subset \overline{T}$

II. Matching

$$\text{If } \mathcal{F} := \overline{T_i} \cap \overline{T_j} \neq \emptyset, \text{ then}$$
$$\psi \in P(\varphi_j^{-1}\mathcal{F}) \Rightarrow \varphi_i^* \varphi_j^{-*}\psi \in P(\varphi_i^{-1}\mathcal{F})$$

- Traces of primal spaces for adjacent cells “line up”
- Can pullback or pushfoward to F from either side

For H_1, we have

$$\begin{aligned} \varphi_i^* \varphi_j^{-*}\psi &\in P(\varphi_i^{-1}\mathcal{F}) \\ \varphi_j^{-*}\psi &\in P(\mathcal{F}) \\ \psi &\in P(\varphi_j^{-1}\mathcal{F}) \end{aligned}$$

Notation

Q^p	Reference functionals associated with $p \in S$, so that $Q = \bigcup_{p \in S} Q^p$
Q^p_i	Pushforward of functionals to cell T_i, $\varphi_{i*} Q^p$, so that $Q_i = \bigcup_{p \in Q} Q^p_i$
Sym_N	The symmetric group on N elements

III. Independence

$$\text{If } \exists p, q \in S \text{ such that } \varphi_i(p) = \varphi_j(q)$$
$$\text{for adjacent cells } T_i \text{ and } T_j,$$
$$\text{then } \exists M \in \mathrm{Sym} \text{ such that } \mathcal{Q}_i^p = M\mathcal{Q}_j^q.$$

- Traces of dual spaces for adjacent cells “line up”
- Mappings push functionals forward into each other
- M encodes symmetries of polytopes in S

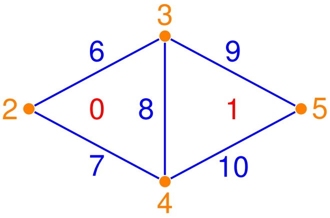
Outline

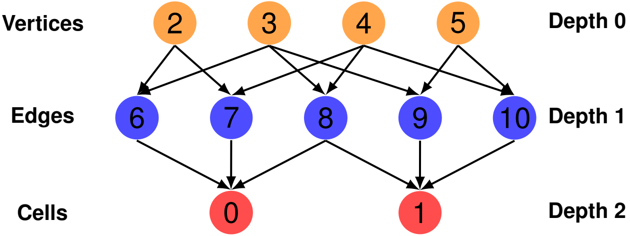
Outline

2 Plex Enhancement
- Short Review of Plex
- Parent-Child and Support Additions
- Dual Basis Calculation

Sample Meshes

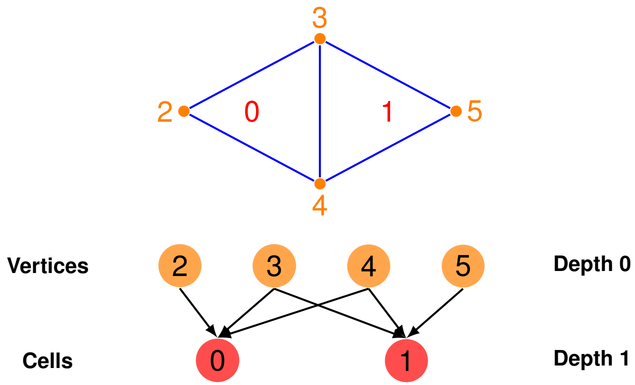
Interpolated triangular mesh





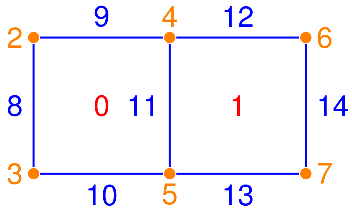
Sample Meshes

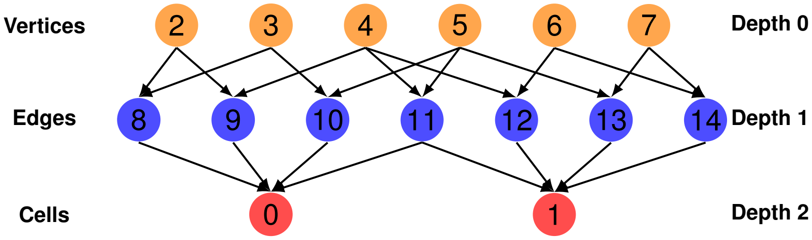
Optimized triangular mesh



Sample Meshes

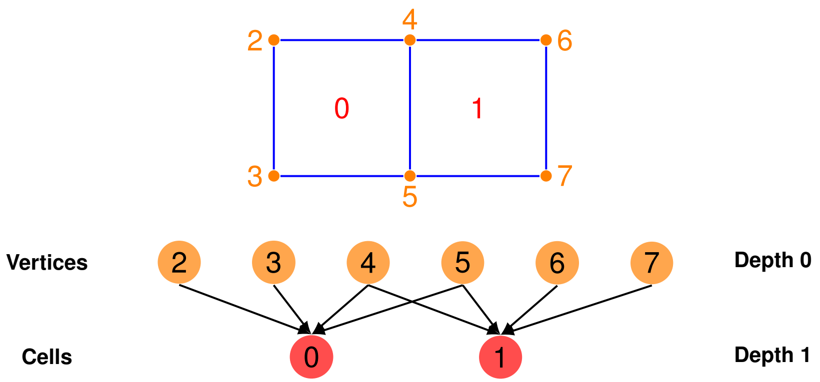
Interpolated quadrilateral mesh





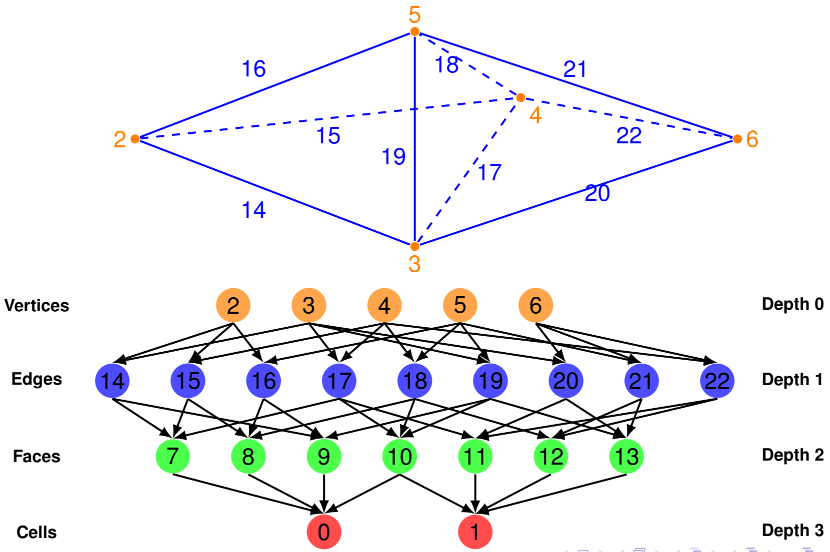
Sample Meshes

Optimized quadrilateral mesh



Sample Meshes

Interpolated tetrahedral mesh

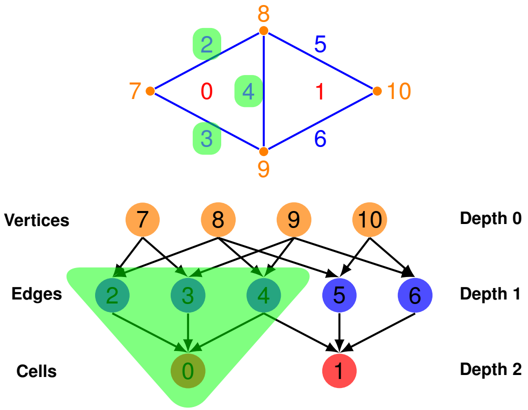


Basic Operations

Cone

We begin with the basic covering relation,

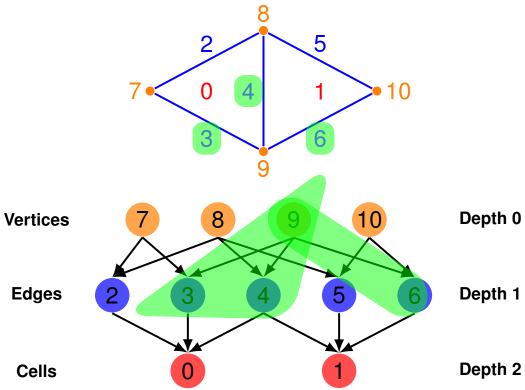
cone(0) = {2, 3, 4}



Basic Operations
Support

reverse arrows to get the dual operation,

support(9) = {3, 4, 6}

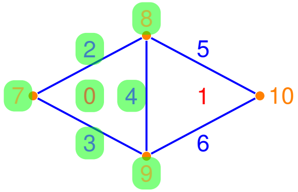


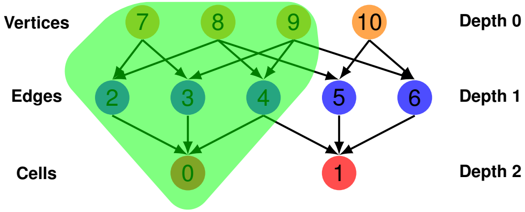
Basic Operations

Closure

add the transitive closure of the relation,

closure(0) = {0, 2, 3, 4, 7, 8, 9}

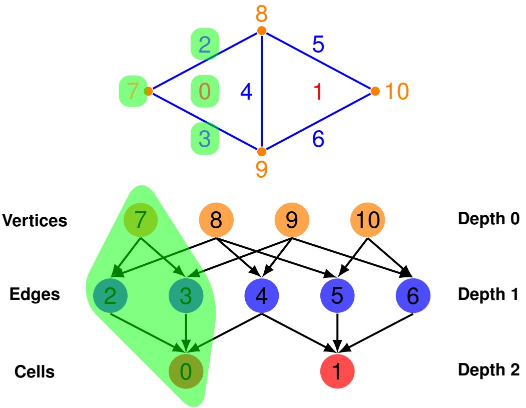




Basic Operations
Star

and the transitive closure of the dual,

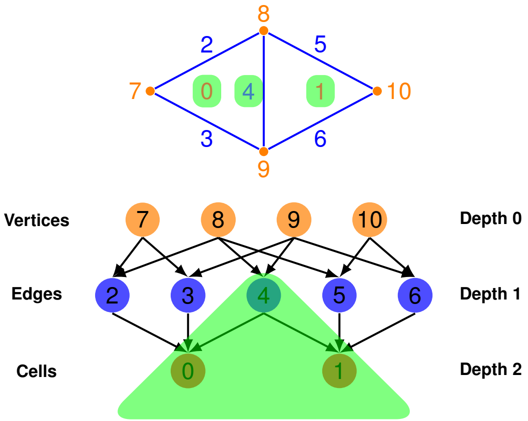
star(7) = {7, 2, 3, 0}



Basic Operations
Meet

and augment with lattice operations.

meet(0, 1) = {4}

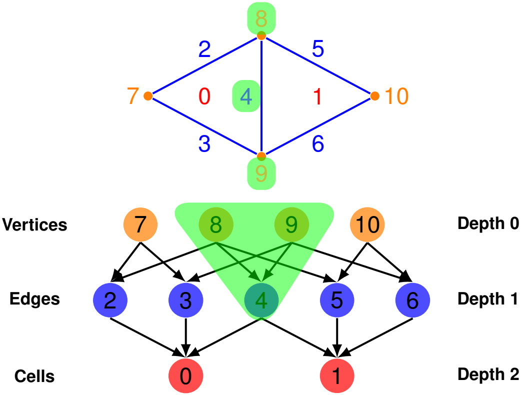


Basic Operations

Join

and augment with lattice operations.

join(8, 9) = {4}

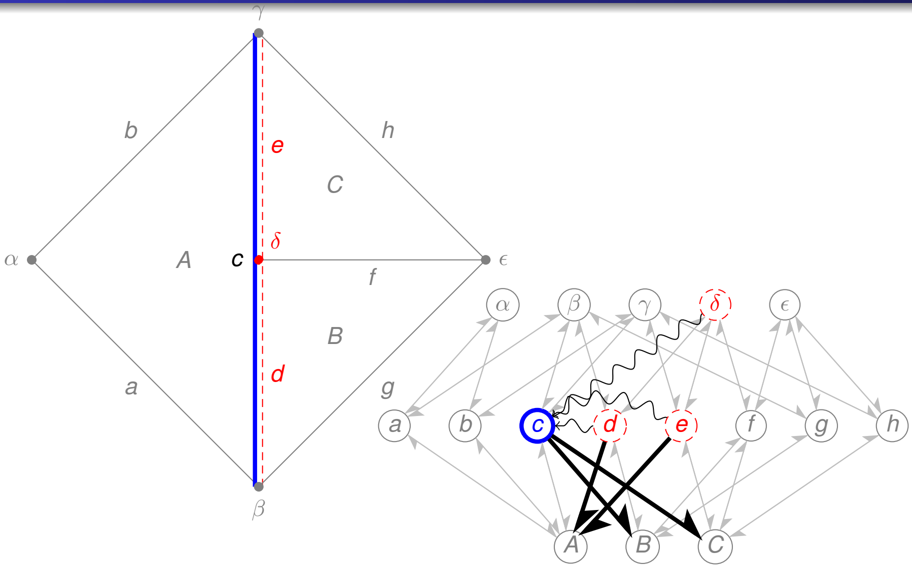


Outline

2 Plex Enhancement
- Short Review of Plex
- Parent-Child and Support Additions
- Dual Basis Calculation

Nonconforming Doublet

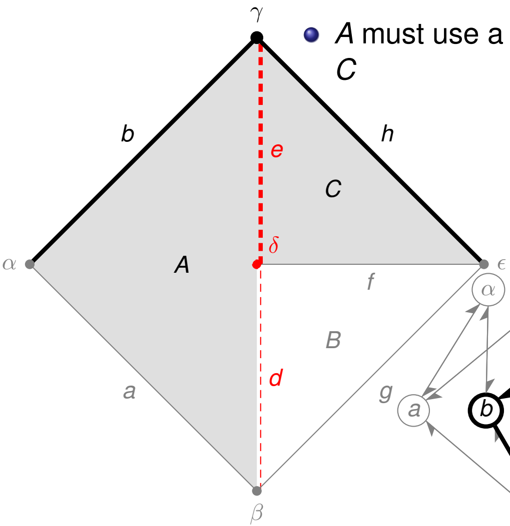
How to encode in Plex?

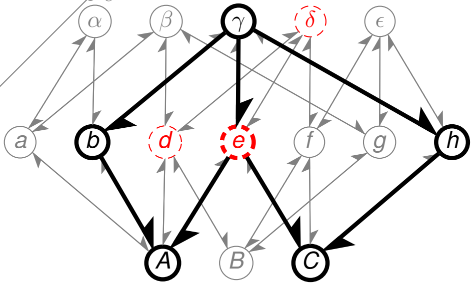


Nonconforming Doublet

Choice 1: Make A a degenerate quadrilateral

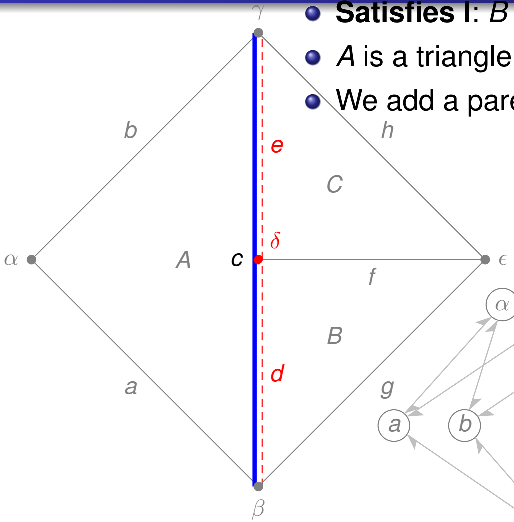
- **Violates I**: $B \in \text{supp}(\gamma)$, but $B \notin \text{star}(\gamma)$
- *A* must use a different element than *B*, *C*



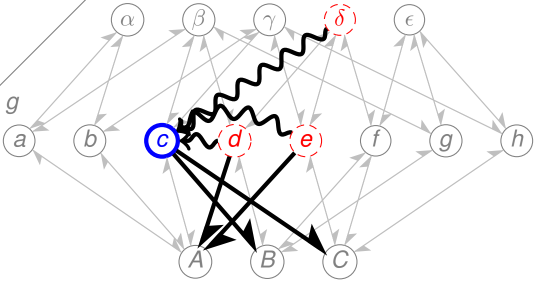


Nonconforming Doublet

Choice 2: Break cone-support duality



- **Satisfies I**: $B \in \mathrm{supp}(\gamma)$, and $B \in \mathrm{star}(\gamma)$
- *A* is a triangle element like *B*, *C*
- We add a parent-child relation



Outline

2 Plex Enhancement
- Short Review of Plex
- Parent-Child and Support Additions
- Dual Basis Calculation

Dual Bases

In general, the union of all cell functionals

$$W^u = \bigcup_{i=1}^{N_T} \bigcup_{p \in S} Q_i^p$$

will contain linear dependencies. Instead, we use

$$W^c = \bigcup_{i=1}^{N_T} \bigcup_{\{p \in S : \mathrm{parent}(\varphi_i(p)) = \emptyset\}} Q_i^p.$$

and we must have a linear relation

$$W^u = I_c^u W^c$$

Creating I_c^u

If we have a child point p such that

- $p, q \in S$
- $\varphi_i(p) \subset \varphi_j(q)$
- $\varphi_j^{-1} \circ \varphi_i : p \to q$ is affine

then we can expand Q_i^p in terms of Q_j.

Creating I_c^u

For $\sigma_r \in Q^p$, by Axiom II,

$$\begin{aligned}
(\varphi_{*,i}\sigma_r)(v) &= (\varphi_{*,i}\sigma_r)(\varphi_j^{-*}\varphi_j^* v) \\
&= (\varphi_{*,j}^{-1}\varphi_{*,i}\sigma_r)(\varphi_j^* v) \\
&= \sum_{\sigma_s \in Q} (\varphi_{*,j}^{-1}\varphi_{*,i}\sigma_r)(\psi_s)\sigma_s(\varphi_j^* v) \\
&= \sum_{\sigma_s \in Q_j} (\varphi_{*,j}^{-1}\varphi_{*,i}\sigma_r)(\psi_s)\sigma_s(v) \\
&= \sum_{\sigma_s \in \cup_{t \in \mathrm{clos}(\mathrm{parent}(p))} Q^t} (\varphi_{*,j}^{-1}\varphi_{*,i}\sigma_r)(\psi_s)\sigma_s(v).
\end{aligned}$$

where we use Axiom I in the last line.

Creating I_c^u

Two Key Points:

- Sparsity of I_c^u
  We find *anchor points*, the points in clos of the transitive closure of $\mathrm{parent}(p)$ that are in W^c.

- Entries in I_c^u
  The matrix interpolates Q_i^p from its anchor point functionals. The entries have the form $(\varphi_{*,j}^{-1}\varphi_{*,i}\sigma_r)(\psi_s)$ for $\sigma_r \in Q$ and shape function $\psi_s \in P(K)$.

Creating I_c^u

Also, refinement usually follows a predictable pattern,

so we can evaluate the transfer functionals for the refined reference cell,

using a *reference tree* stored as a Plex,

and then map to an actual cell.

Creating I_c^u

```
/* Concatenate functionals of Q as pointsRef and weights */
EvaluateBasis(bspace,fSize,nPoints,sizes,pointsRef,weights,work,Amat);
/* Amat(i,j) evaluates basis i at dual basis functional j */
MatLUFactor(Amat,NULL,NULL,NULL);
/* loop over cells */
for (c = cStart; c < cEnd; c++) {
  DMPlexGetTreeParent(dm,c,&parent,NULL);
  if (parent == c) continue;
  /* Ref. tree mappings are affine, corner (v0) and Jacobian (J) */
  DMPlexComputeCellGeometryFEM(dm,c,NULL,v0,J,NULL,&detJ);
  DMPlexComputeCellGeometryFEM(dm,parent,NULL,v0parent,Jparent,invJparent,&detJpar);
  for (i = 0; i < nPoints; i++) {
    /* spdim is the spatial dimension */
    /* push coordinates of functionals forward from child */
    CoordinatesRefToReal(spdim,spdim,v0,J,&pointsRef[i*spdim],vtmp);
    /* pull coordinates of functionals back to parent */
    CoordinatesRealToRef(spdim,spdim,v0parent,invJparent,vtmp,&pointsReal[i*spdim]);
  }
  EvaluateBasis(bspace,fSize,nPoints,sizes,pointsReal,weights,work,Bmat);
  /* Bmat(i,j) evaluates basis i at transfered functional j */
  MatMatSolve(Amat,Bmat,Xmat);
  /* ... partition the columns of Xmat between the points in clos(c)
```

Creating I_c^u

If σ_r is associated with $p \in \mathrm{clos}(c)$,

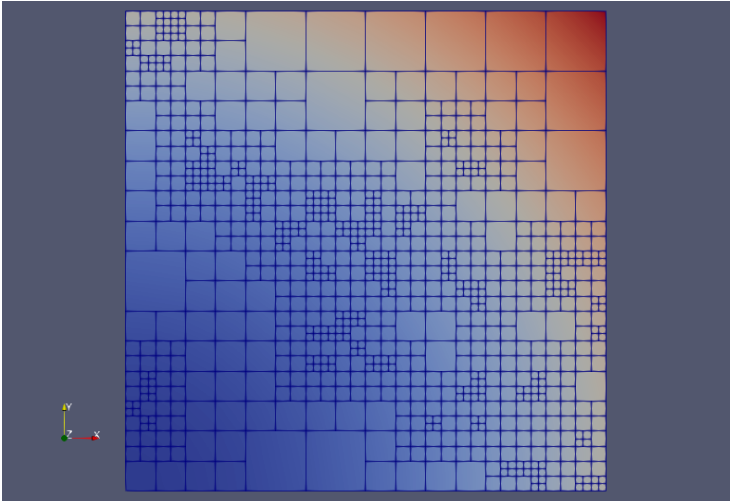
column r of X constrains σ_r to the dual basis of root cell $\mathrm{parent}(c)$,

X_{sr} is only nonzero if functional σ_s is associated to a point in $\mathrm{clos}(\mathrm{parent}(p))$.

Outline

Poisson with Finite Elements

A Poisson problem discretized with Q_2 elements



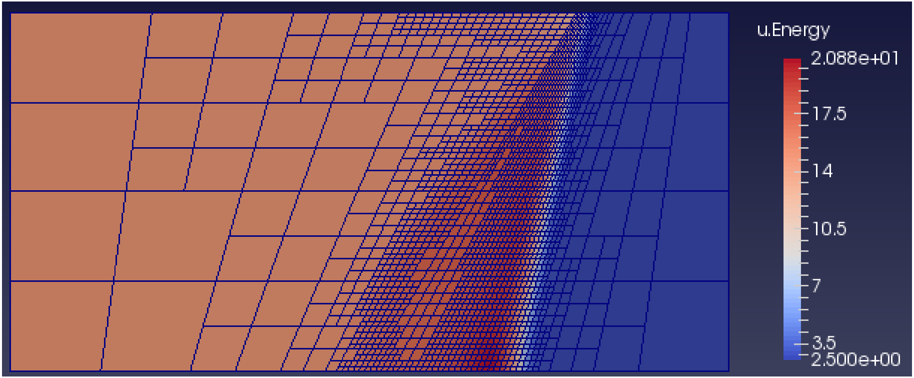
Poisson with Finite Elements

A Poisson problem discretized with Q_2 elements reproduced using SNES ex12:

```
./ex12 -run_type test -simplex 0 -interpolate 1
  -petscspace_order 2 -petscspace_poly_tensor
  -dm_plex_convert_type p4est -dm_forest_initial_refinement 2
    -dm_forest_minimum_refinement 0
    -dm_forest_maximum_refinement 6
    -dm_p4est_refine_pattern hash
  -dm_view vtk:amr.vtu:vtk_vtu
  -vec_view vtk:amr.vtu:vtk_vtu:append
```

Euler with Finite Volumes

A shock impinging on an oblique density contrast modeled using the Euler equation discretized with a TVD FV method



Euler with Finite Volumes

A shock impinging on an oblique density contrast modeled using the Euler equation discretized with a TVD FV method reproduced using TS ex11:

```
./ex11 -ufv_vtk_interval 1 -monitor density,energy -f -grid_size 2,1 -grid_bounds -1,1.,0.,1
   -bc_wall 1,2,3,4
 -dm_type p4est -dm_forest_partition_overlap 1 -dm_forest_maximum_refinement 6
   -dm_forest_minimum_refinement 2 -dm_forest_initial_refinement 2
 -ufv_use_amr -refine_vec_tagger_box 0.5,inf -coarsen_vec_tagger_box 0,1.e-2
   -refine_tag_view -coarsen_tag_view
 -physics euler -eu_type iv_shock -ufv_cfl 10 -eu_alpha 60. -grid_skew_60 -eu_gamma 1.4
   -eu_amach 2.02 -eu_rho2 3.
 -petscfv_type leastsquares -petsclimiter_type minmod -petscfv_compute_gradients 0
 -ts_final_time 1 -ts_ssp_type rks2 -ts_ssp_nstages 10
```

Advantages

Why is this good?

- Can do unstructured refinement as well
- Can do arbitrary refinements (not just 2:1)
- Can do arbitrary shapes (not just quads)
- Integrates seamlessly with solvers

Thank You!

http://www.caam.rice.edu/~mk51