Smooth and linear surface interpolations

applied to three dimensional Boundary Elements

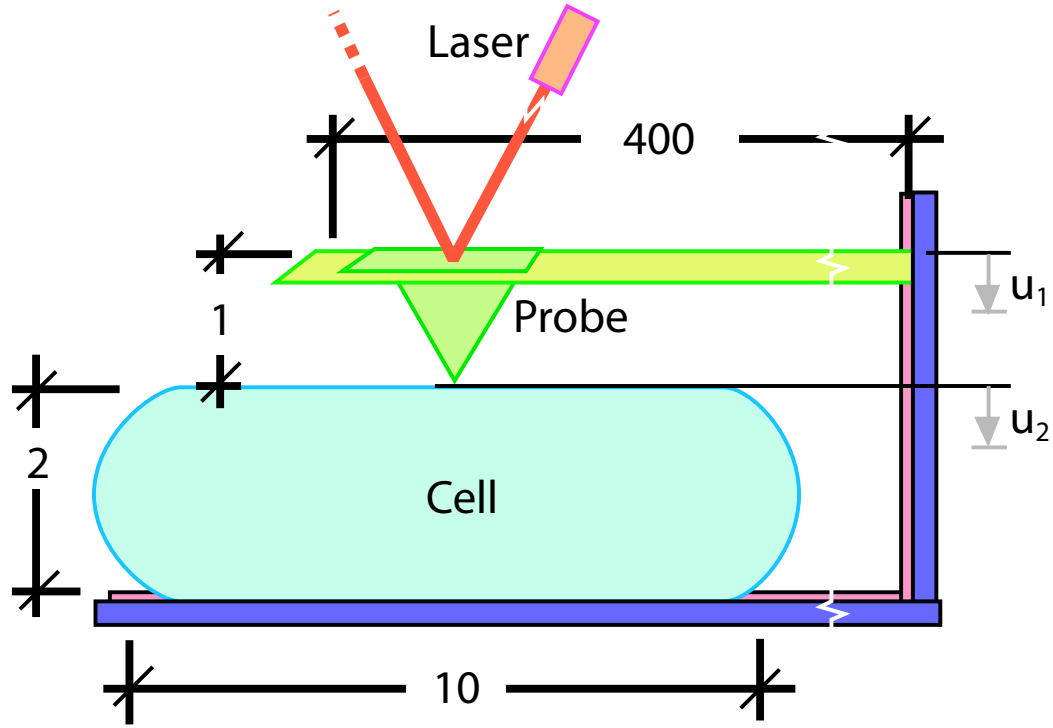
Elisabeth Anna Malsch

Institut für Angewandte Mechanik
Technische Universität Braunschweig

Columbia University
In the City of New York

Boundary Integral Methods ---- Reading, London ---- Sept. 2004

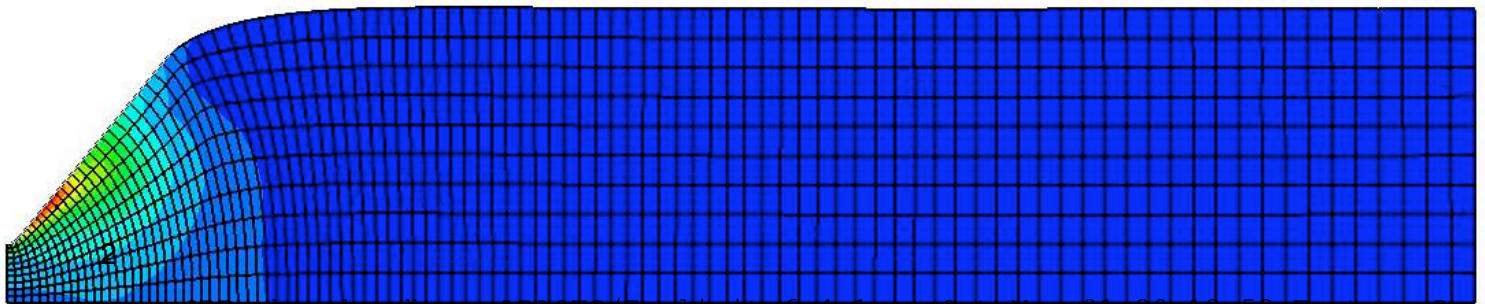
Model contact:
Atomic Force Microscopy



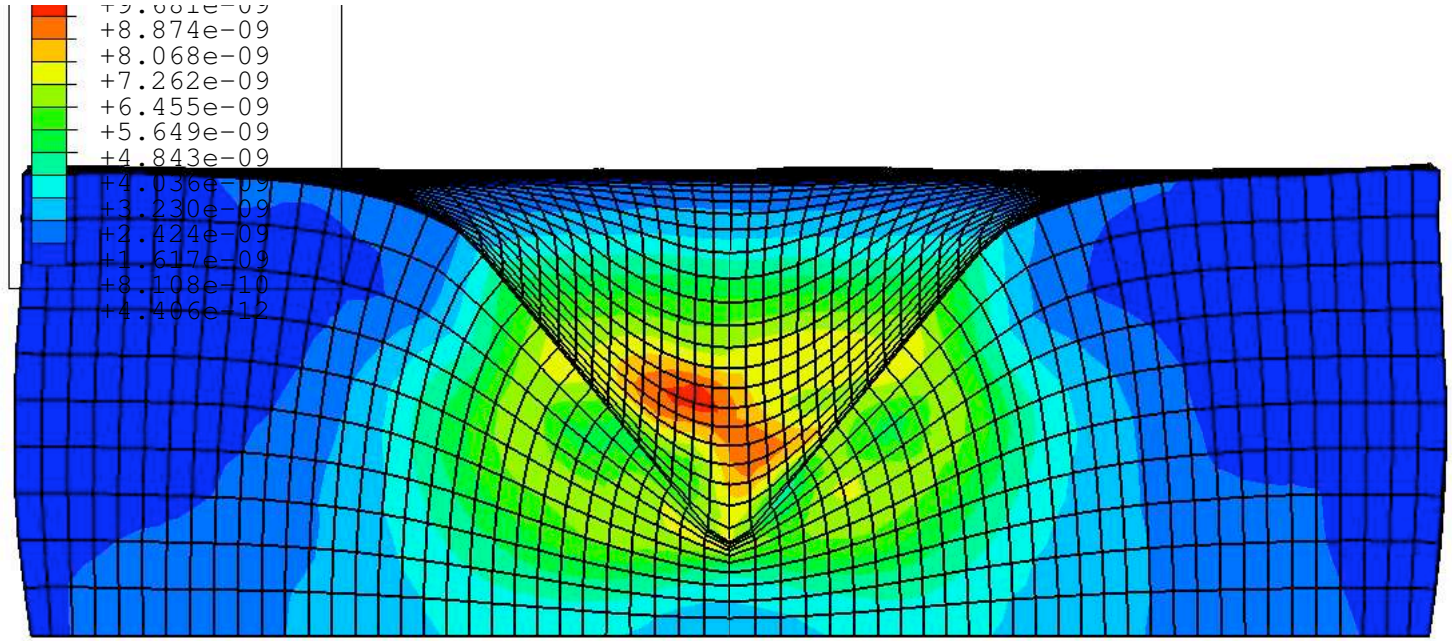
- Contact
- Large deformation
- Surface-probe interaction
- Small-scale material behavior

Problem: Large deformation and changing contact surface and specimen geometry

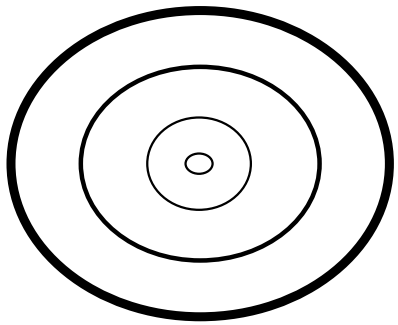
2D



3D



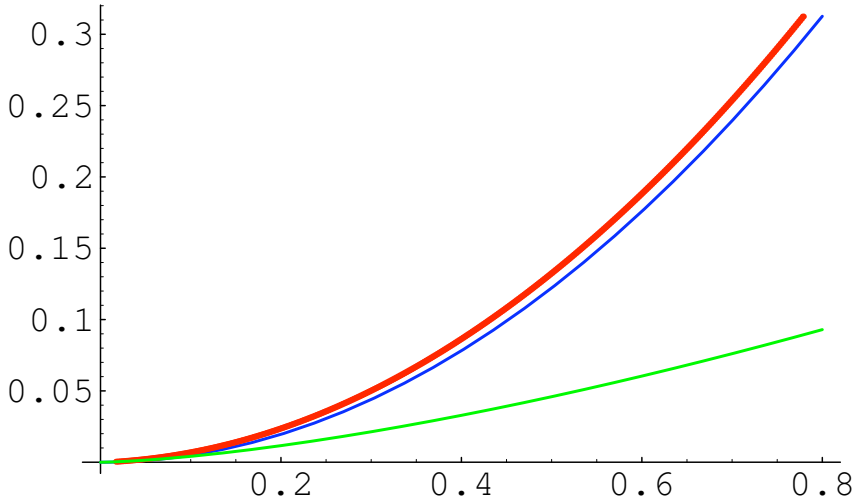
Surface indentation: Approximate models and simplified simulation



surface contours

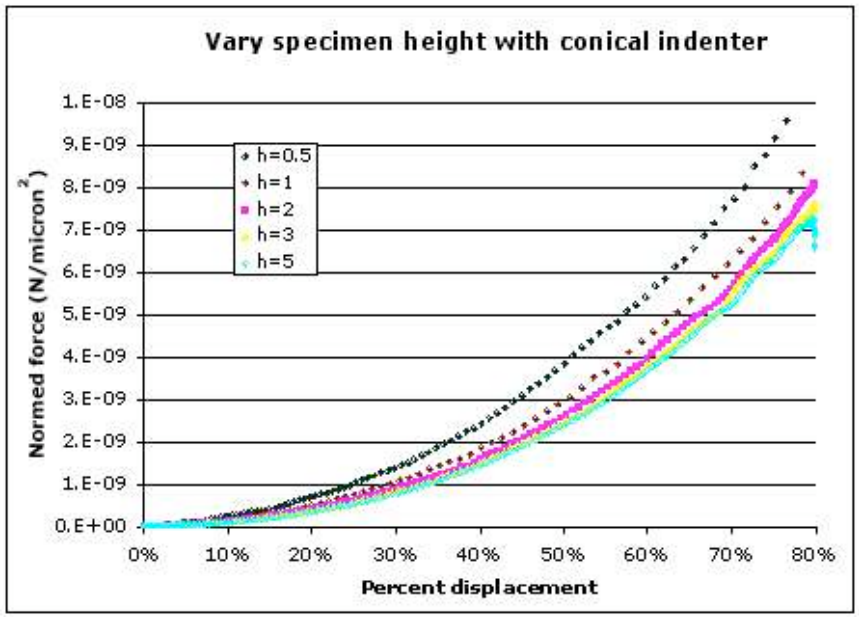
- Hertz contact
  - small deformation
  - isotropic material
- Scaling
  - homogeneous material

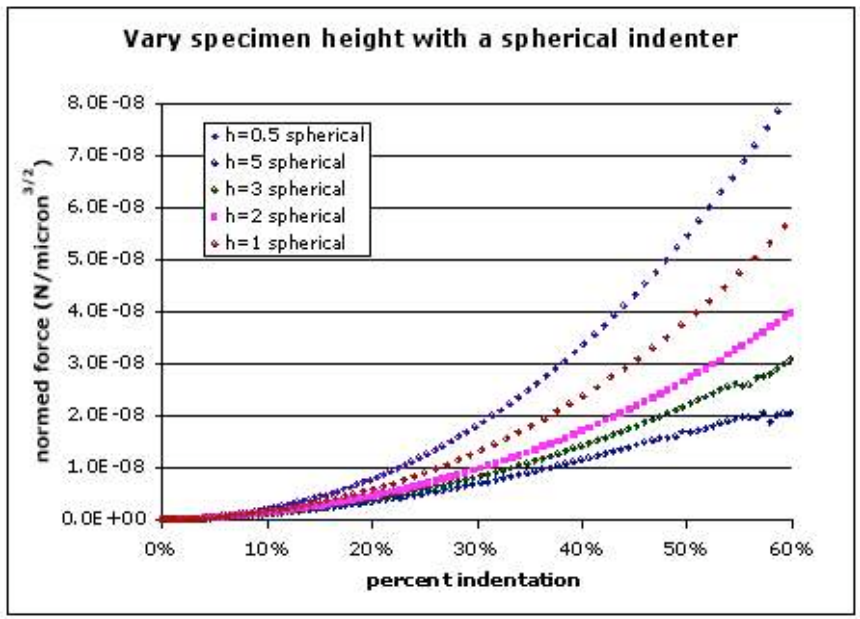
Typical force vs. displacement curves



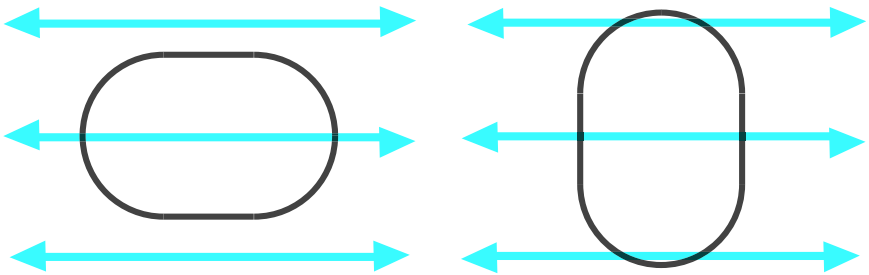
Hertz model for different tip geometries

Finite element simulation with differing specimen heights

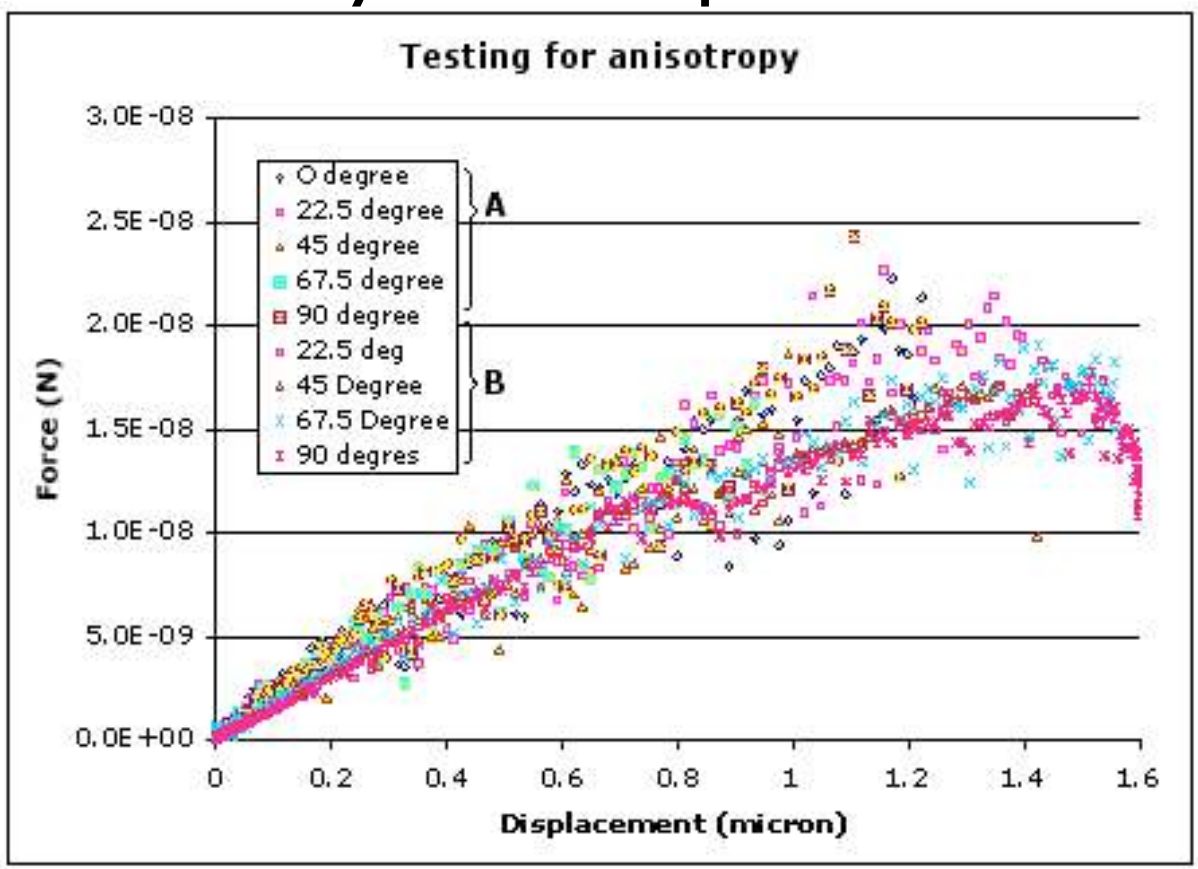


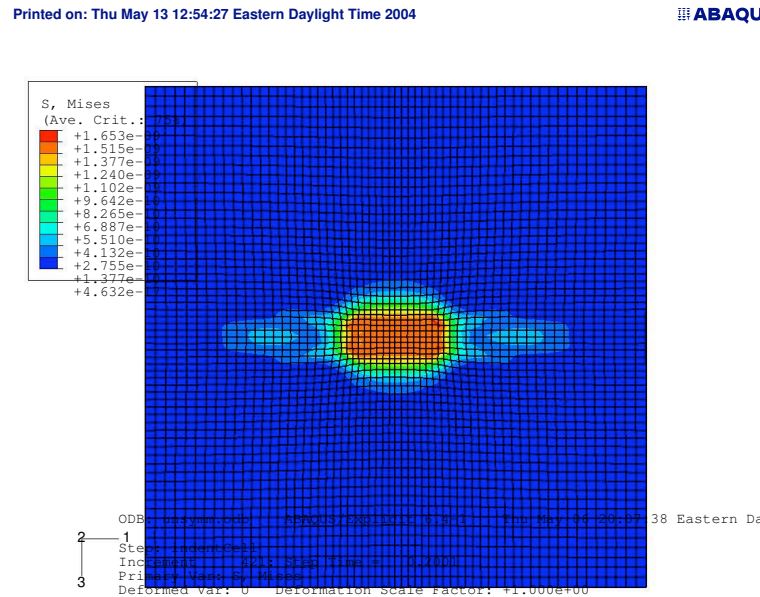


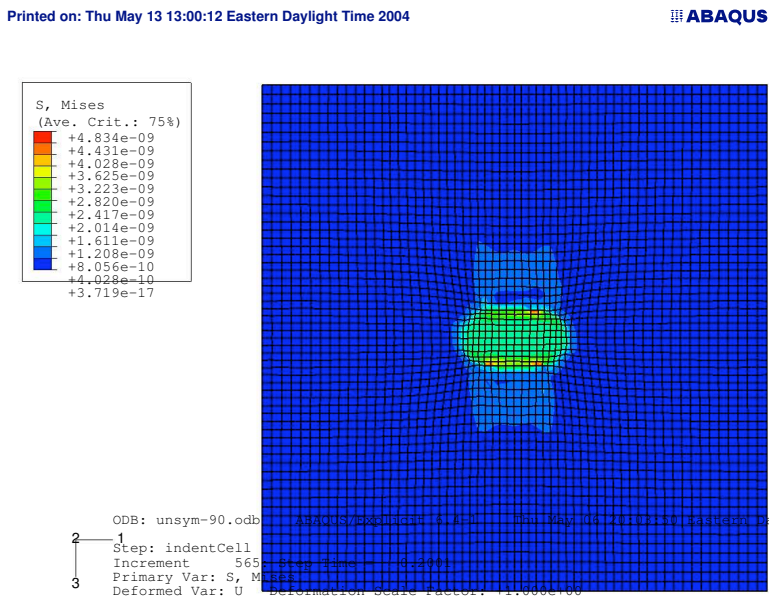
FEM Force - displacement difficulty

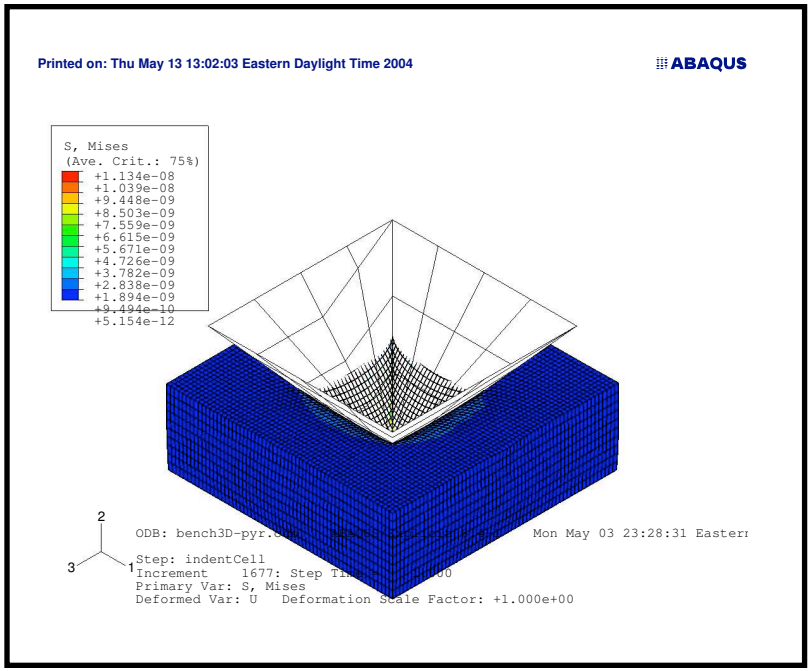


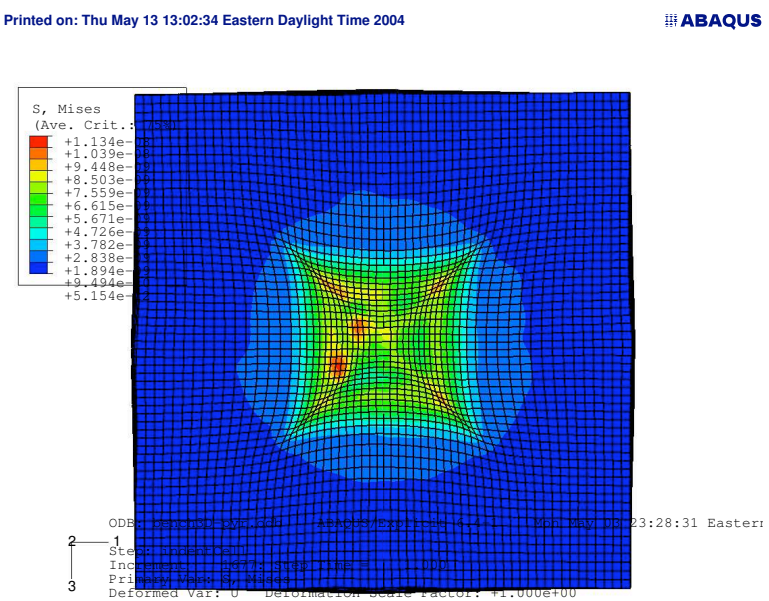
Example: Indentation of an extremely anisotropic membrane











Contact Surface

III. Undeformed

II. Deforming

☒ I. Contact

Boundary element formulation

$$c^i u^i + \int_S \frac{\partial}{\partial n} g(\vec{x}|\vec{x}_i)\hat{u}_j(\vec{x})\mathrm{d}S = \int_S g(\vec{x}|\vec{x}_i)\hat{q}_j(\vec{x})\mathrm{d}S$$

- Fundamental solution: $\nabla^2 g(\vec{x}|\vec{x}_i) = \delta_i(\vec{x})$
- Surfaces: S_j

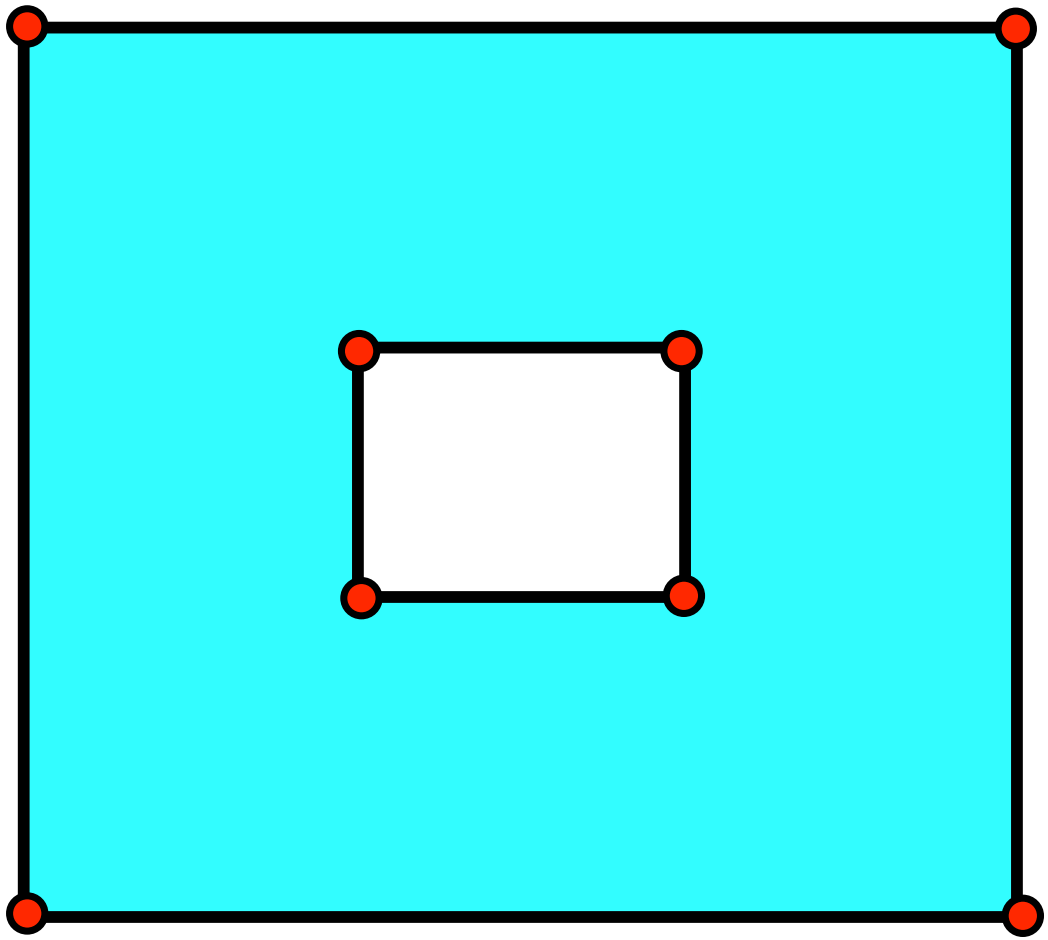
Let:

$$\hat{u}_j(\vec{x}) = \sum_k^{n_j} u_k N_k^j(\vec{x}) \quad \text{and} \quad \hat{p}_j(\vec{x}) = \sum_k^{n_j} p_k N_k^j(\vec{x})$$

Functions to integrate over each surface S_j:

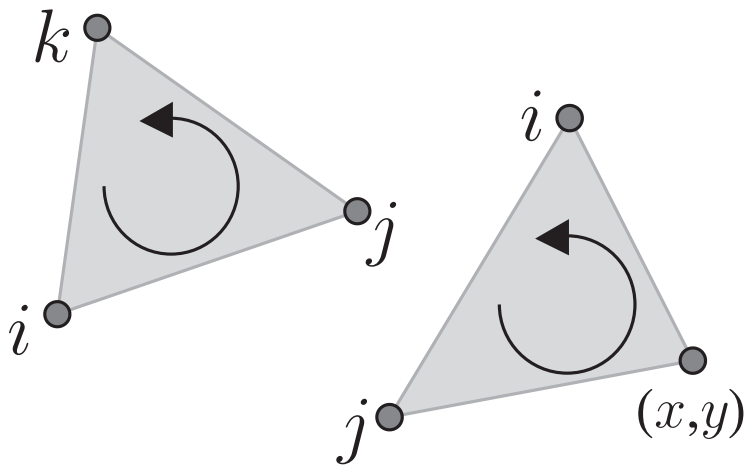
$$A_{ik}^j(\vec{x}) = g(\vec{x}|\vec{x}_i)N_k^j(\vec{x}) \quad \text{and} \quad B_{ik}^j(\vec{x}) = \frac{\partial}{\partial n} g(\vec{x}|\vec{x}_i)N_k^j(\vec{x})$$

I. Representing a surface geometry with very few nodes and sources



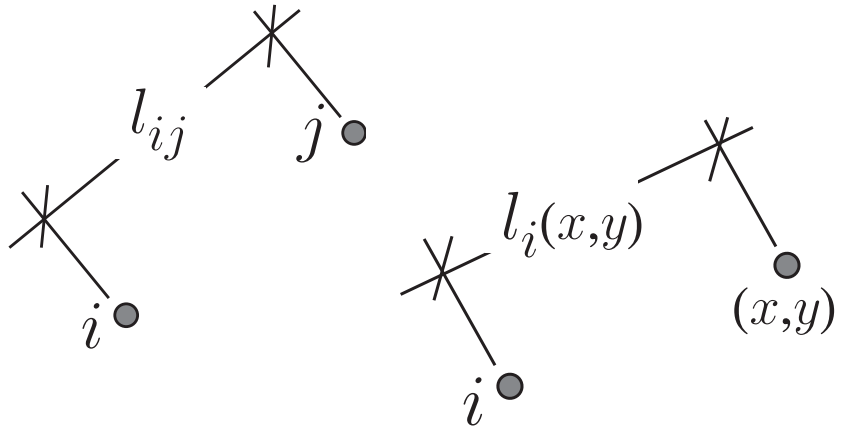
The displacements and pressures on the *far away* surface are close to zero or zero ---
very few nodes are needed.

Area



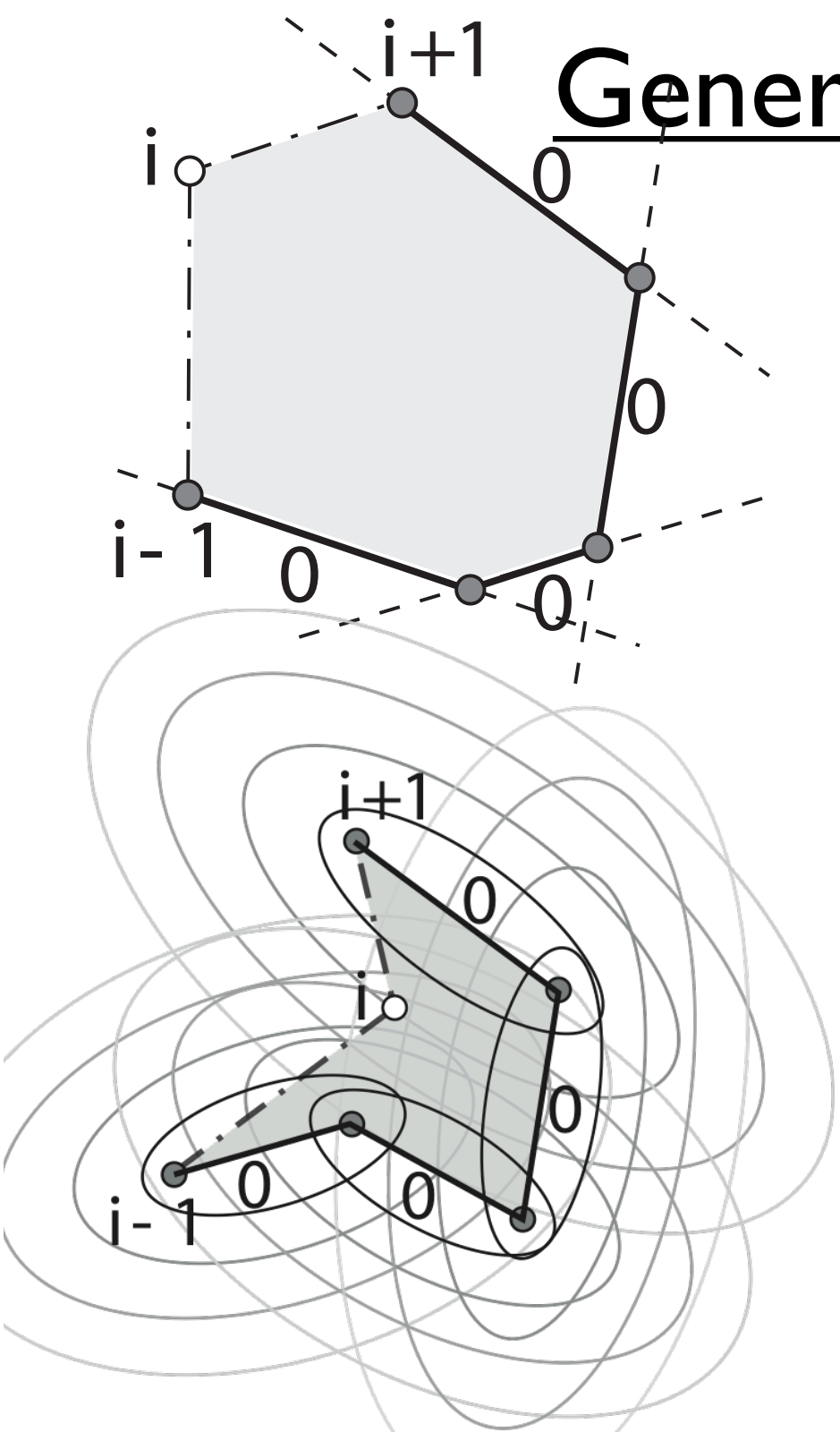
$$\triangle(i,j,k) = \frac{1}{2}\begin{vmatrix} x_i & y_i & 1 \\ x_j & y_j & 1 \\ x_k & y_k & 1 \end{vmatrix}$$

Length

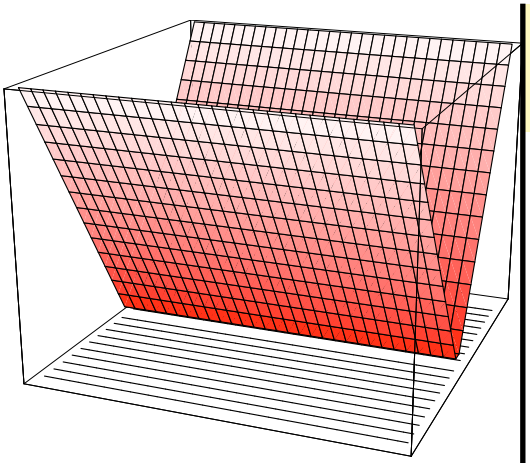


$$l_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

General element formulation

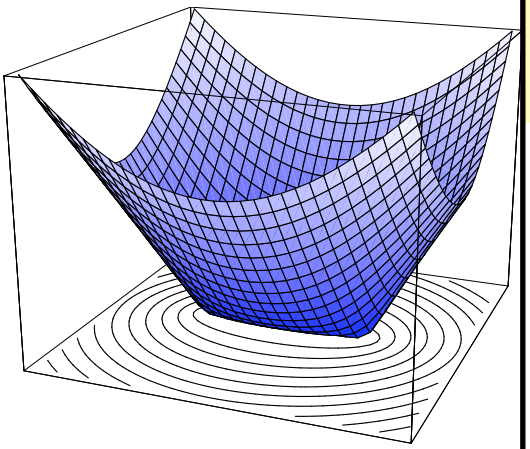


- Describe each boundary segment with a function which is zero only along the segment.
- Multiply the line segment functions together such that to form the numerator of the shape function such that it is zero at every boundary not adjacent to the node.
- Normalize the function.



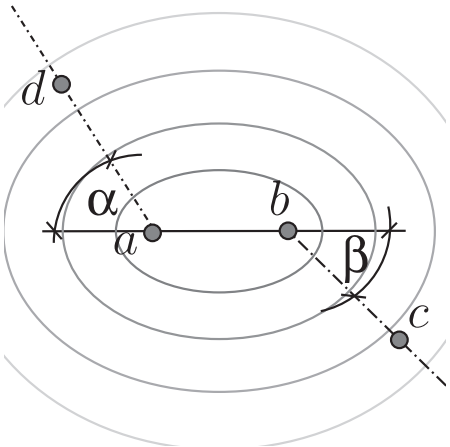
$$\triangle(\vec{x}, i, j)$$

The linear function which is zero along the line connecting points **i** and **j** can be described as the area of the triangle defined by the points **i**, **j** and the variable point **(x,y)**.



$$r_{ij}(\vec{x}) = l_i(\vec{x}) + l_j(\vec{x}) - l_{ij}$$

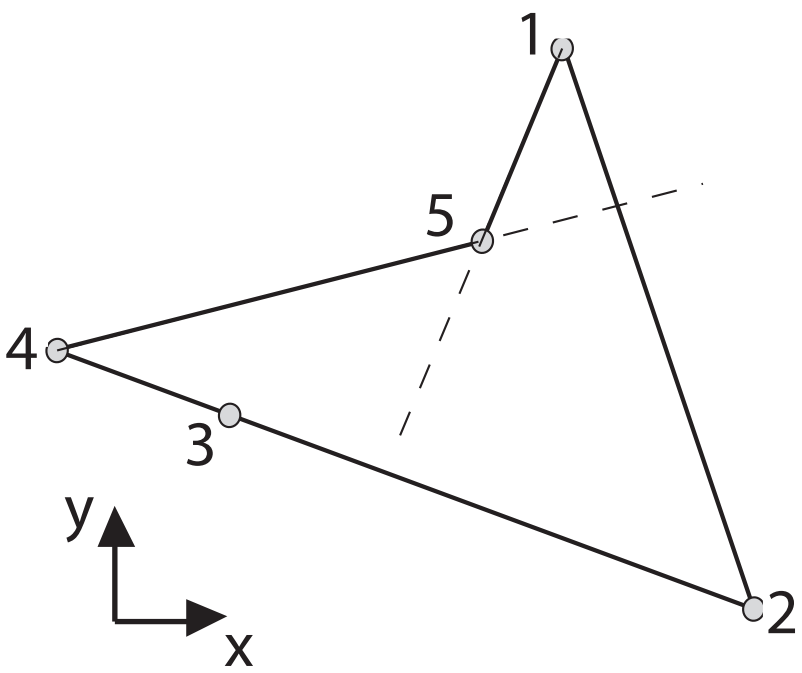
This elliptical function constructed as a combination of line lengths is zero only along the finite line connecting point **i** and **j**.



$$r_{dabc}(\vec{x})$$

A elliptical function which is linear along the lines **da** and **bc** and zero and minimum only along the line **ab** can be constructed as a combination of line lengths and areas.

Example:



Numerators:

1) $k_1\, r_{23}(x,y) r_{34}(x,y) r_{45}(x,y)$

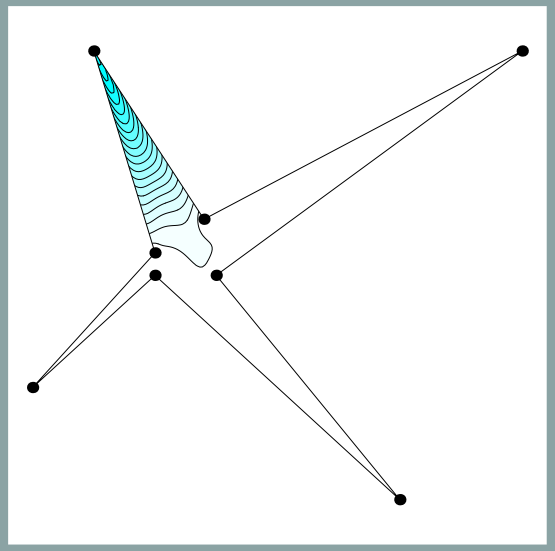
2) $k_2\, r_{34}(x,y) r_{45}(x,y) r_{51}(x,y)$

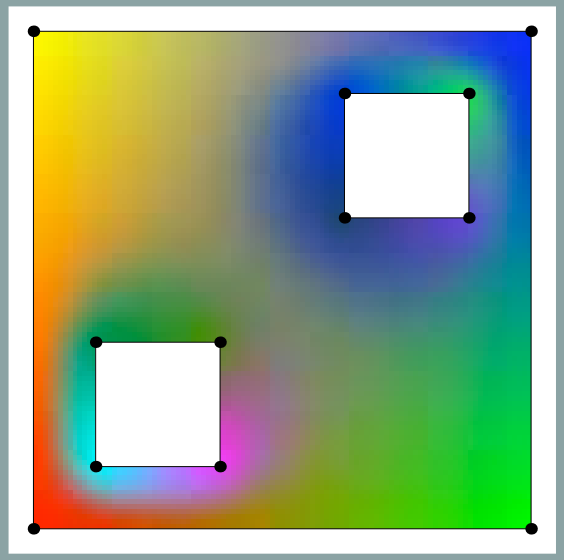
3) $k_3\, r_{45}(x,y) r_{51}(x,y) r_{12}(x,y)$

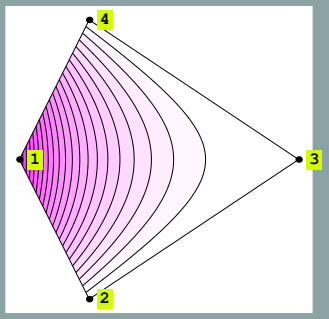
4) $k_4\, r_{51}(x,y) r_{12}(x,y) r_{23}(x,y)$

5) $k_5\, r_{12}(x,y) r_{23}(x,y) r_{34}(x,y)$

$$N_1(x,y) = \frac{k_1 r_{23}(x,y) r_{34}(x,y) r_{45}(x,y)}{k_1 r_{23}(x,y) r_{34}(x,y) r_{45}(x,y) + \cdots + k_5 r_{12}(x,y) r_{23}(x,y) r_{34}(x,y)}$$

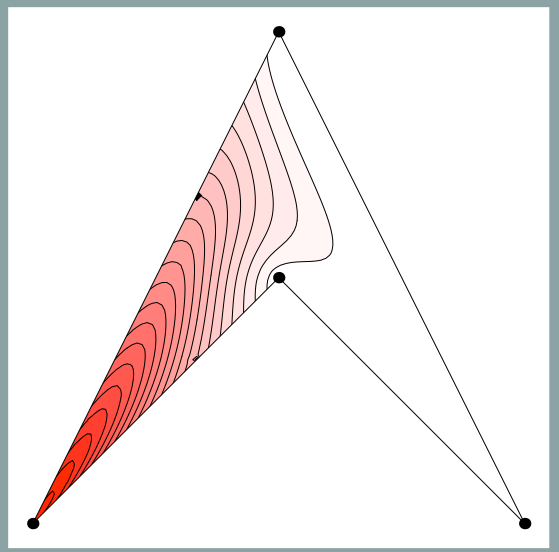


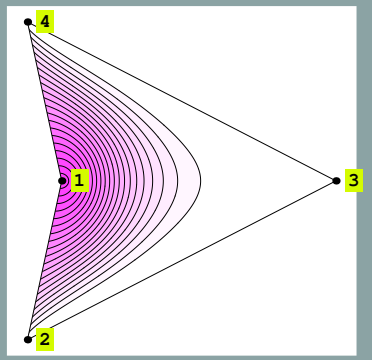






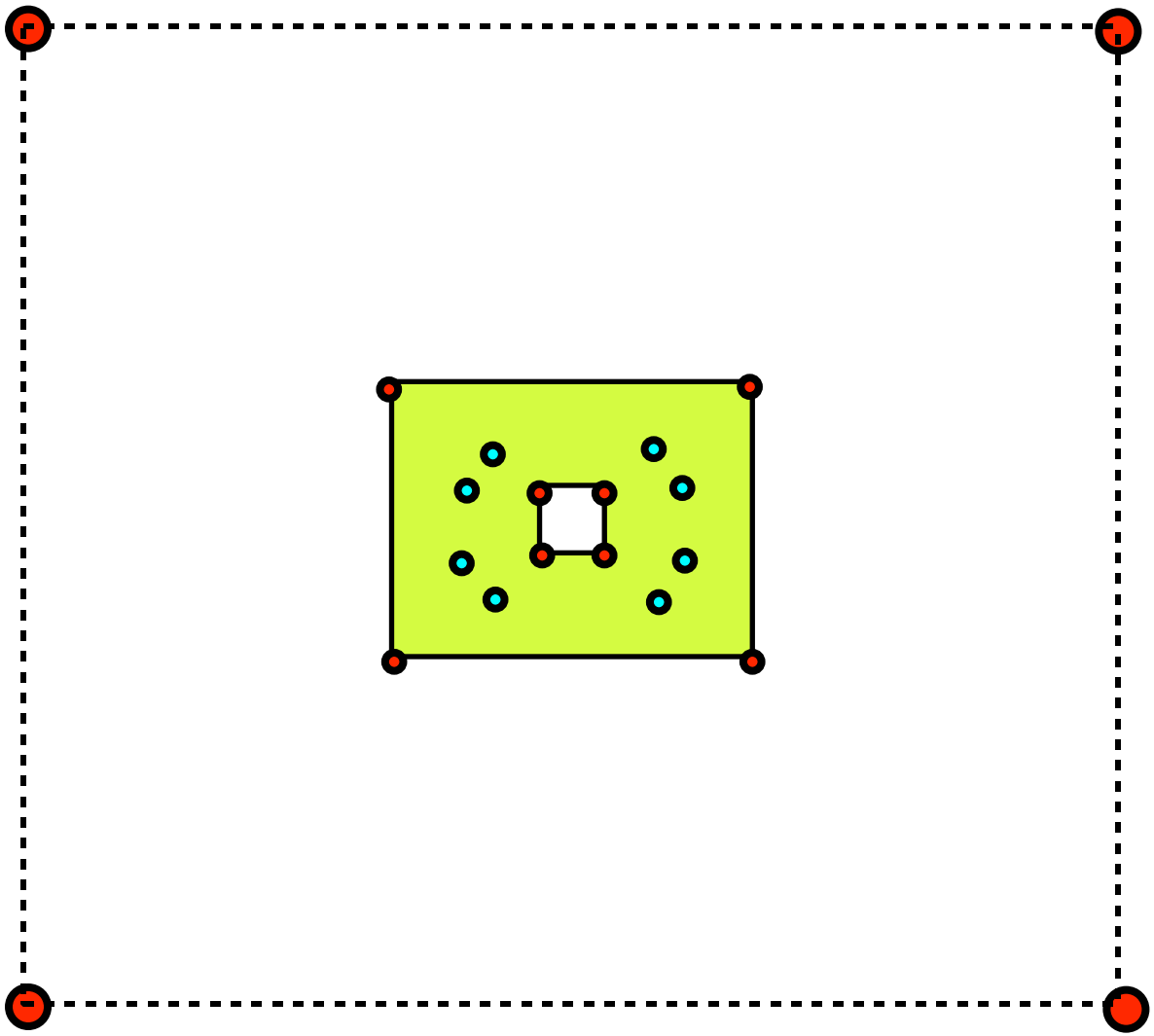






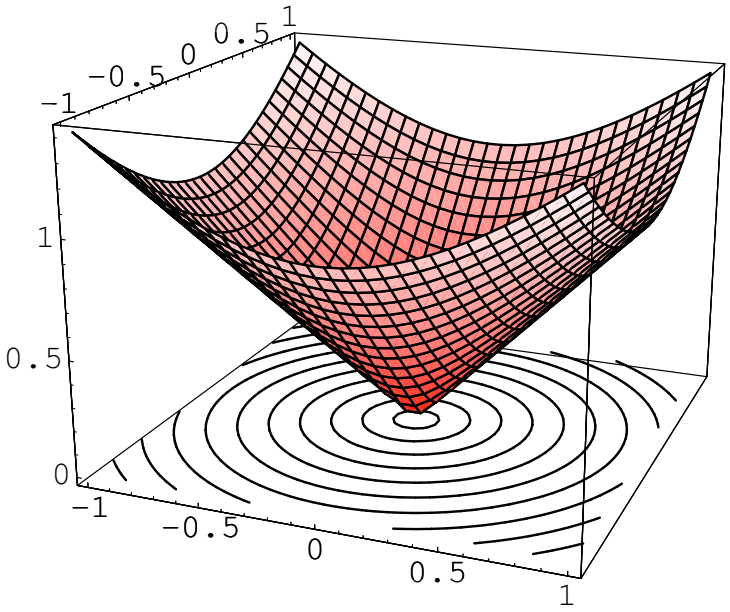
Any polygonal shape can be described

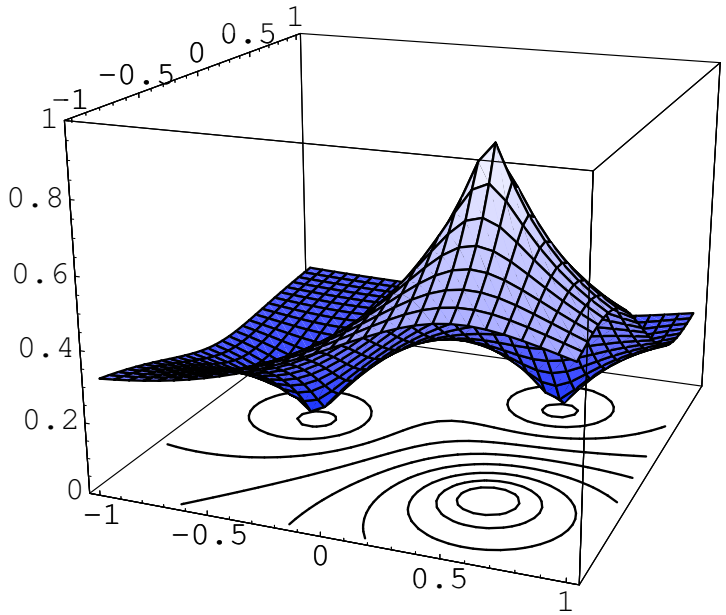
II. Adding interior nodes and sources



On surfaces where the greatest displacements take place add more nodes

Interpolations for interior point sources:

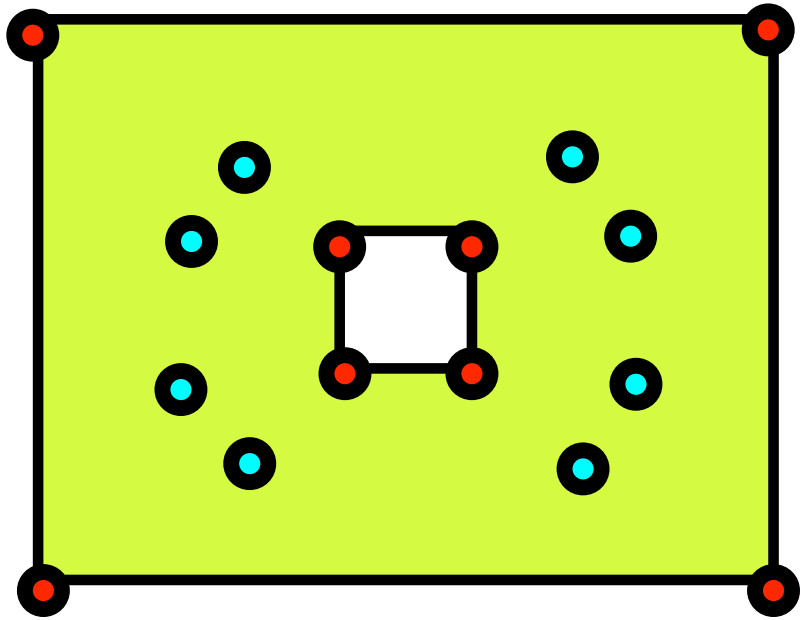




$$L_i(\vec{x}) = \frac{\prod_{j \neq i} l_j(\vec{x})}{\sum_{k=1}^{n} \left(\prod_{j \neq k} l_j(\vec{x}) \right)}$$

Linearly independent influence functions (radial basis functions)

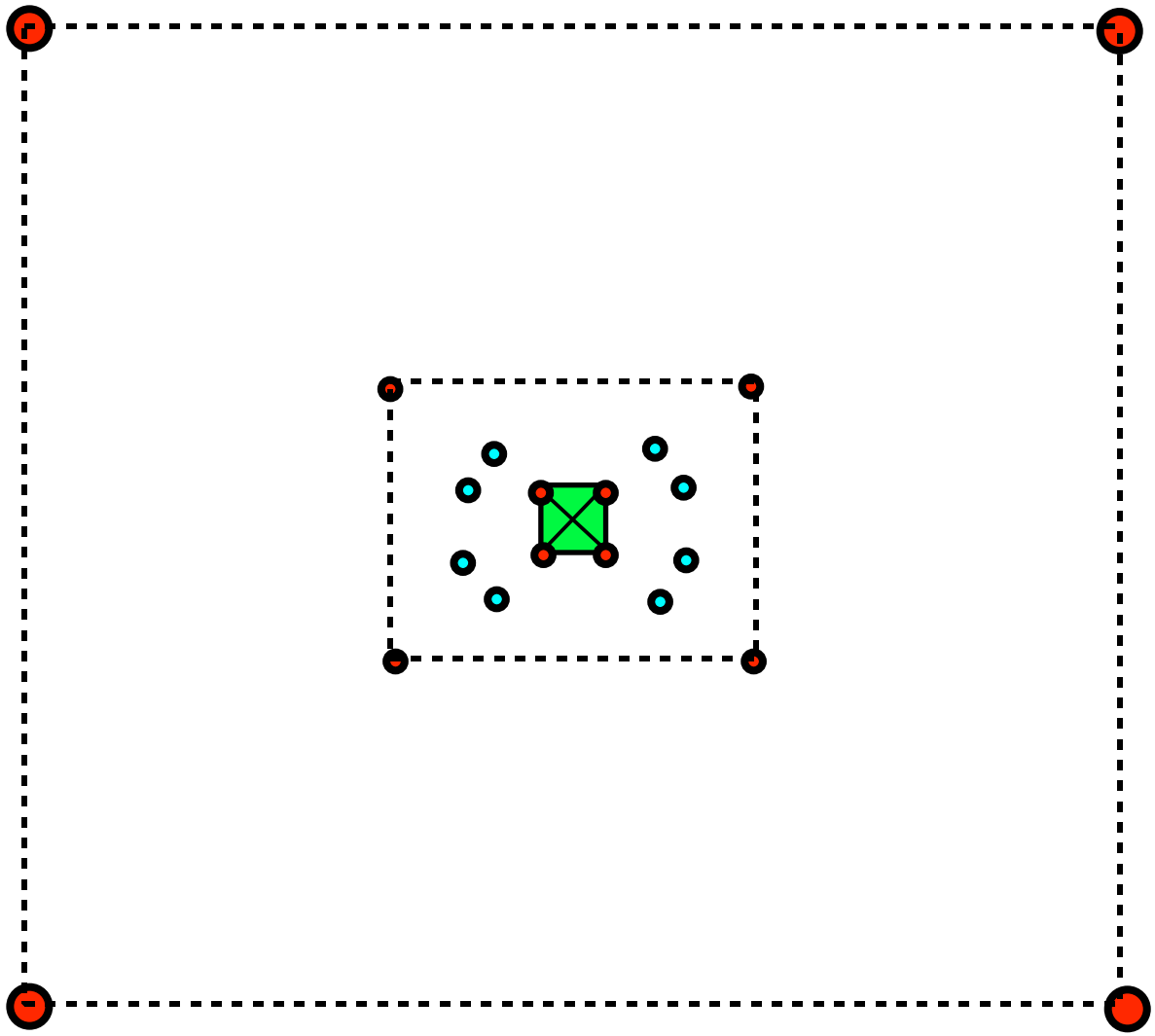
Combining boundaries and interior nodes



$$N_j(\vec{x}) = L_j(\vec{x}) \frac{\prod_{\text{all}}^{m} r_{kl\ldots}(\vec{x})}{\prod_{\text{all}}^{m} r_{kl\ldots(\vec{x})} + \prod_{\text{all}}^{n} l_k(\vec{x})} \quad \forall j \quad \text{where point } \vec{x}_j \text{ is an interior point.}$$

$$N_i(\vec{x}) = R_i(\vec{x}) \frac{\prod_{\text{all}}^{n} l_k(\vec{x})}{\prod_{\text{all}}^{m} r_{kl\ldots}(\vec{x}) + \prod_{\text{all}}^{n} l_k(\vec{x})} \quad \forall i \quad \text{where point } \vec{x}_i \text{ is a vertex point.}$$

III. Triangulated surface: Standard 3D Boundary Element



The simple geometry in the contact area can be modeled by triangulating the surface and using constant or linear elements.

Numerical integration over a polygonal domain

- Domain decomposition: *triangulation*
- Boundary decomposition: *Divergence Theorem*

$$\int_A Q_{,x}\,\mathrm{d}x\mathrm{d}y = \oint Q\mathrm{d}y = \sum_{j=1}^{N}\int_0^1 Q(\vec{x}(t))\frac{\partial y(t)}{\partial t}\mathrm{d}t$$

$$Q(x,y) = \int A^j_{ik}(x,y)\mathrm{d}x \qquad \text{Or}$$

$$Q(x,y) = \int B^j_{ik}(x,y)\mathrm{d}x$$

Combining the parts: simple implementation

```
surfNorm[pts_]:=Module[{surfZero,xyz={x,y,z}},
   surfZero =Plus@@(Det[Append[#,1]&/@Append[#,xyz]]&/@Partition[pts,Length[pts[[1]]],1,{1}]);
   (#/Norm[#])&[Map[D[surfZero,#]&,xyz]]]
surfXDir[pts_]:=(#/Norm[#])&[{1,-1}.pts[[{1,2}]]]
surfYDir[pts_]:=Cross[surfXDir[pts],surfNorm[pts]]
surfPtsST[surfPts_,pts_]:={surfXDir[surfPts].(#-surfPts[[1]]),surfYDir[surfPts].(#-surfPts[[1]])}&/@pts
surfRep[pts_, xyz_,{s_,t_}]:=Thread[{x,y,z}\[Rule]pts[[1]]+s*surfXDir[pts] + t*surfYDir[pts]];
```

Vector Algebra

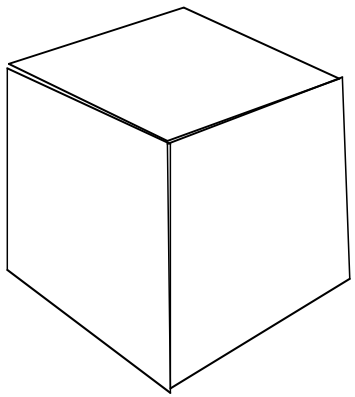
```
GH3D[f_,pts_,surfs_]:=Module[{num,surfPts,xyz,s,t(*,g,h*)},
   surfPts=Map[pts[[#]]&,(Flatten/@surfs)]; num[pts]=Length[pts];num[surfs]=Length[surfs];
   xyz={x,y,z};
   {g=Nest[Map[Function[n,D[#,n]], Array[un,num[pts]]]&,Sum[surfInt[1, f[r]/.r\[Rule]Norm[
   xyz-pts[[i]]]/.surfRep[surfPts[[j]],xyz,{s,t}],surfPtsST[surfPts[[j]],pts],
   surfs[[j]],{s,t},i],{i,num[pts]},{j,num[surfs]}],2],
     h=Nest[Map[Function[n,D[#,n]],Array[un,num[pts]]]&,
        Sum[surfInt[0, -surfNorm[surfPts[[j]]].((D[Norm[xyz-pts[[i]]],#]&/@xyz))
             D[f[r],r]/.r\[Rule]Norm[xyz-pts[[i]]]/.surfRep[surfPts[[ j]],xyz,{s,t}],surfPtsST[surfPts[[j]], pts],surfs[[j]],{s,t}, i],
         {i,num[pts]},{j,num[surfs]}],2];
     h-=IdentityMatrix[num[pts]]*(Plus@@h)}]
```

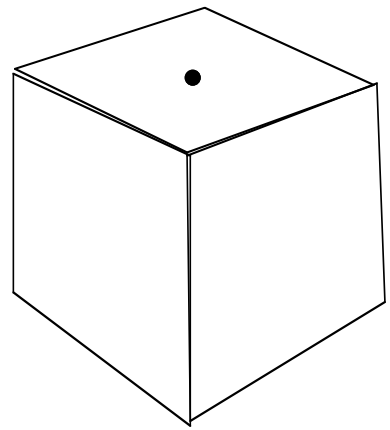
Boundary Element Formulation

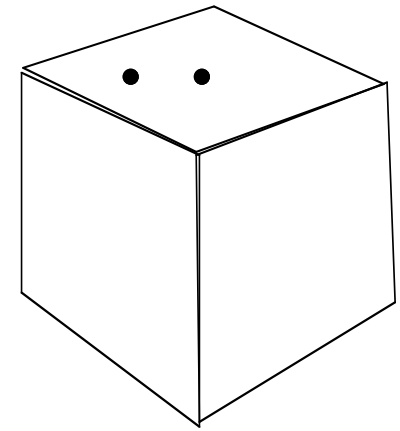
```
surfInt[type_,z_,pts_,surf_,{s_,t_},i_]:=Module[{nsf,num=Length[Flatten[
   surf]],j},
   nsf=combPrt[{s,t},pts,surf,surf[[1]]];
   un[i]*Table[If[(type\[Equal]2&&i\[Equal]surf[[j]]),
     1/2,NIntegrate[Evaluate[z*nsf[[j]]/.Abs\[Rule](
       Sqrt[#^2]&)],{s,-1,-3/4,-1/2,-1/
   4,0},{t,-1,-3/4,-1/2,-1/4,0}]],{j,num}].(un/@Flatten[surf])
 ]
```

Numerical Integration

First simple example -- dice:







Comparing eigenvalues of the computed conductivity matrix:

```
 concave shape functions
(8.49134  4.9545  4.9545  4.9545  2.04401  2.04401  2.04401  0)
 convex shape functions
(11.1928  5.9679  5.9679  5.9679  2.24487  2.24487  2.24487  0)
mixed shape functions with an interior node
(19.4708  12.6058  6.9725  6.9725  6.61716  2.5346  2.5346  2.47528  0)
mixed shape functions with two interior nodes
(53.6157  21.61  14.3256  7.95518  7.41952  7.3351  2.8024  2.75249  2.63596  0)
```

Summary:
New 3D Boundary element formulation

- Smooth, bounded shape functions on the surface.
- Surface geometry can be modeled using a minimum number of nodes.
- Interior nodes added using modified Radial Basis Functions.
- Numerical integration can be performed without a mesh using the Divergence Theorem.

Source locations depend on the geometry and boundary requirements, not on a heuristic mesh

Further study

- **Improving the numerical integration**

- **Including other surface behaviors**

  Non-local formulation:

$$\bar{\sigma}_{ij}(r) = \int_v \bar{C}_{ijkl}(r, r')\tilde{\epsilon}_{kl}(r) + \tilde{D}_{ijk}(r, r')\tilde{\phi}_k(r)\mathrm{d}r'$$

  Intrinsic and extrinsic variables:

$$\bar{\xi}_m = \xi_m(\bar{\sigma}_{kl}, \bar{T}_m, \bar{\xi}_n)$$

Acknowledgments:

Alexander von Humboldt Fellowship
http://www.avh.de

Alexander von Humboldt
Stiftung/Foundation

Department of Civil Engineering & Engineering Mechanics, Columbia University
NSF Grant CMS: 0202232

Profs. Gautam Dasgupta and Eugene Wachspress

Cardiac Biomechanics Group, Columbia University
NSF Career Grant: 0239138

Prof. Kevin Costa

Thank You.