Introduction Iterative Methods

Eric de Sturler
Department of Mathematics, Virginia Tech
sturler@vt.edu
http://www.math.vt.edu/people/sturler

Iterative Methods for Linear Systems: Basics to Research
Numerical Analysis and Software I

Purpose of this Course

- In this course we will discuss the most important methods for the iterative solution of systems of linear equations and their analysis.
- We will consider the performance of different methods on relevant model problems.
- We will consider links to systems of nonlinear equations and eigenvalue problems.
- We will look at several important areas of current research and some surprising problems/results.
  - Sequences of linear systems
  - New convergence theory
  - Preconditioning
    - Saddle-point problems, adaptive mesh refinement, …
  - Inexact matrix-vector products,
  - …

Overview

- Basic iterations, Krylov spaces and Krylov methods
- Conjugate Gradients
- GMRES, GMRES, MINRES, and variants
- Convergence theory
- Extensions of Krylov methods
- Biconjugate Gradients and variants (BiCGStab,QMR,TFQMR)
- Preconditoning (multilevel techniques)
- Solving sequences of problems (solvers and preconditioners)
- Inexact matrix-vector products
- Eigenvalue problems
- Nonlinear problems and optimization
- …

Sources of Linear Systems of Equations

- Scientific and engineering simulations require the solution of (many) very large, sparse, linear systems.
- The matrices arise from finite element/volume/difference discretization of partial differential or integral equations (and other areas) describing the physical behavior of complex systems.
- Accurate solution requires millions of unknowns.
- Time-dependent nonlinear problem: Solve a nonlinear system each timestep, which (Newton iteration) requires many linear systems to be solved.
- Very large optimization problems: each iteration requires the solution of a linear system.
- New fields of application: Financial modeling, Econometry, Biology, Computer Graphics, …

Problems, Problems, Problems

Problem defined by (non)linear partial differential equation

$$u_t = \nabla \cdot (D\nabla u) + V \cdot \nabla u + R(u) + F(x,t)$$

Flow and chemistry, absorption-scattering, deformation, …

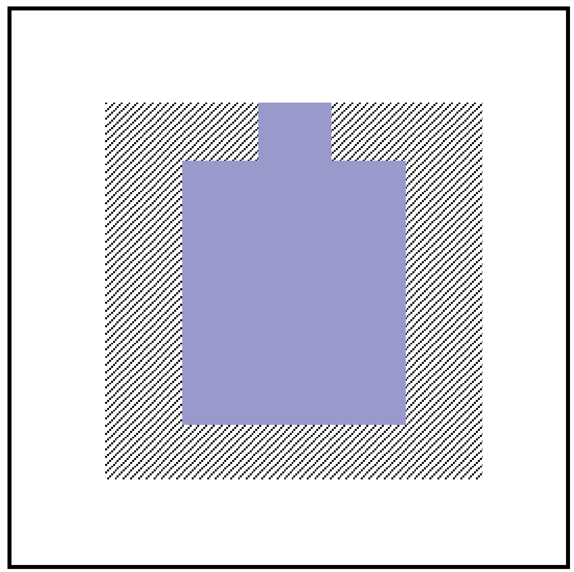
- Find u
- Find $D(x,\rho)$ s.t. $\int \|\nabla u\|_D^2$ minimal where u solves equation
- Find $D(x,p)$ and $R(x,p)$ to match (noisy) data for given F and u

Write u as linear combination of (time-dependent) localized basis functions, apply 'Galerkin' condition, and solve sparse matrix equation for the coefficients (solution)

Requires solution of large system of (non)linear equations

Diffusion / Convection-Diffusion

$u_n = 0$

$u_n = 0$  $u_n = 0$

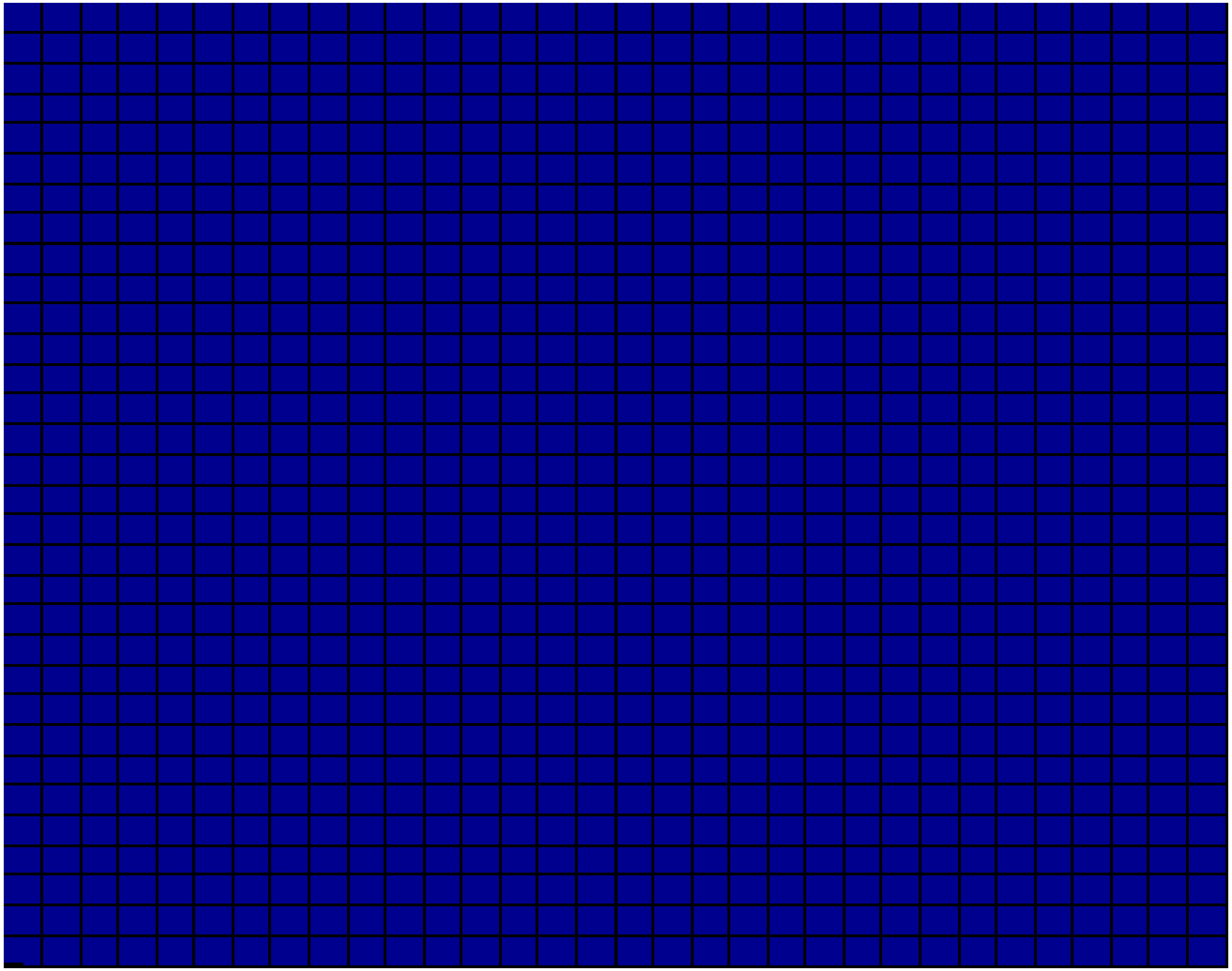
$u = 1$

$$u_t = \nabla \cdot (a \nabla u)$$

$$u_t = \nabla \cdot (a \nabla u) + b u_x$$

white	$a = 10^2, \quad b = 10^3$
hatched	$a = 10^{-5}, \quad b = 0$
shaded	$a = 10^3, \quad b = 0$

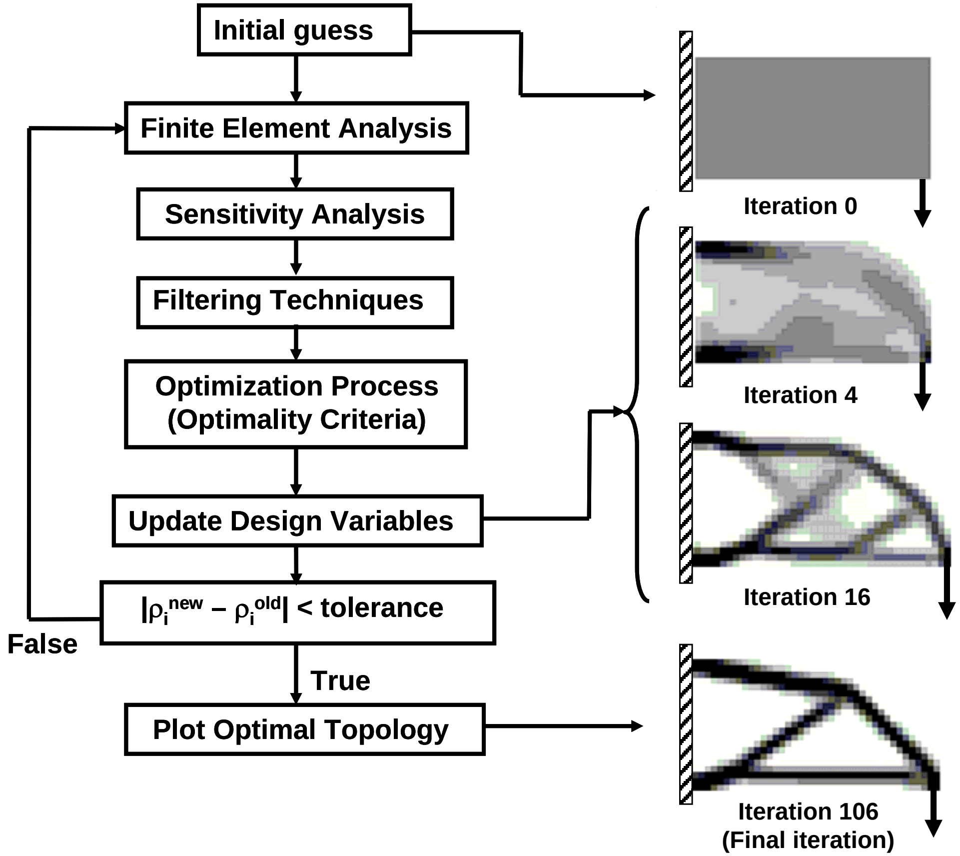
AMR Example



Topology Optimization (with Glaucio Paulino & Shun Wang)

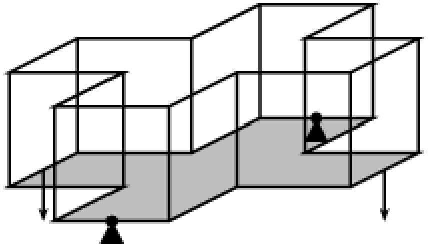
Optimize material distribution, $\boldsymbol{\rho}$, in design domain

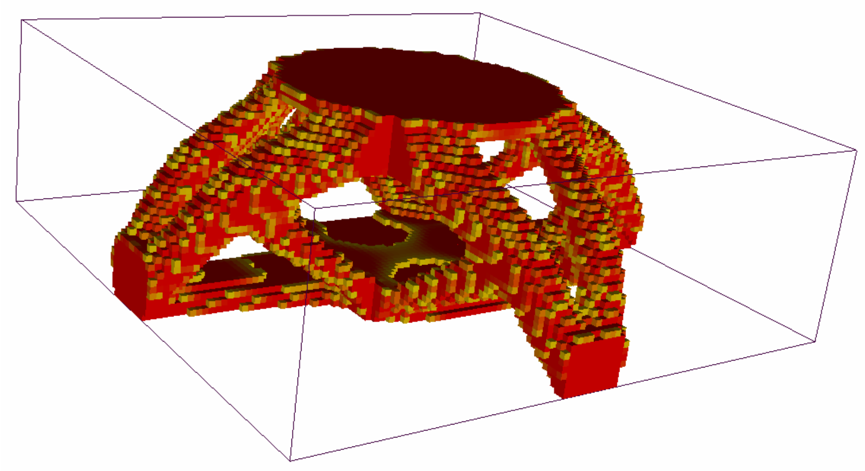
Minimize compliance $\boldsymbol{u}^T \boldsymbol{K}(\boldsymbol{\rho})\boldsymbol{u}$, where $\boldsymbol{K}(\boldsymbol{\rho})\boldsymbol{u}=\boldsymbol{f}$



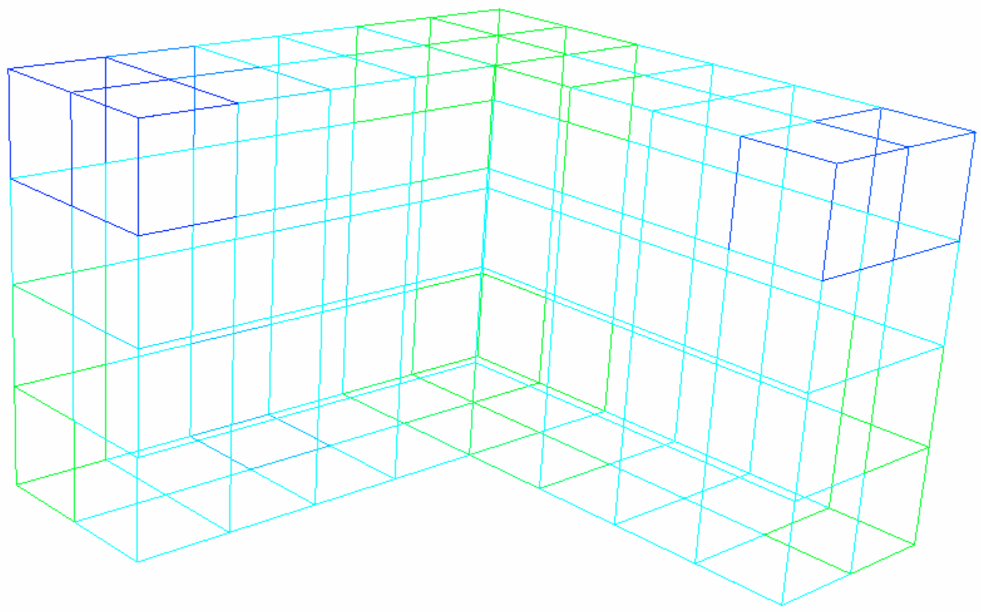
Collaboration with Glaucio Paulino, UIUC

Example: Topology Optimization





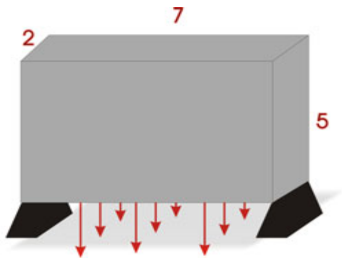
Topology Optimization

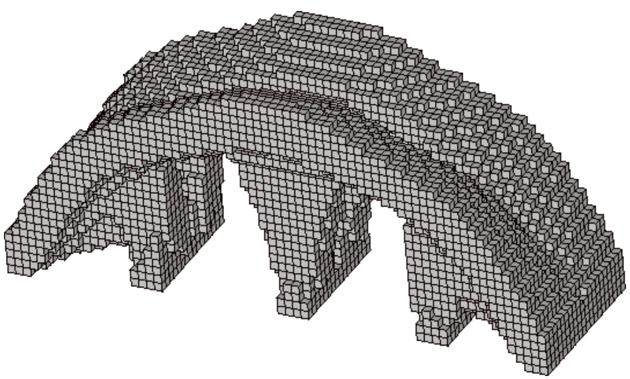


Topology Optimization

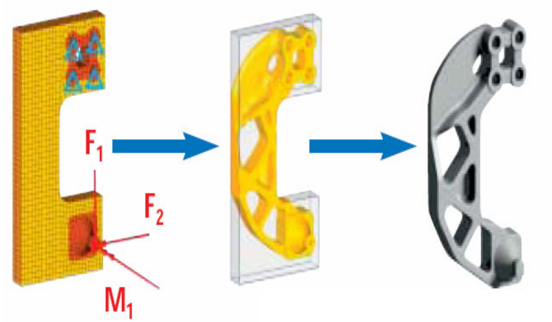
➢ **Design and construction of structures with better performance, in terms of admissible structural responses, without exceeding a certain limit cost;**

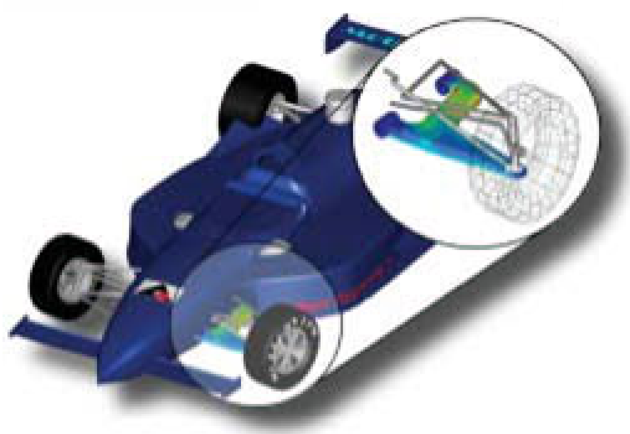
➢ **Applications:**



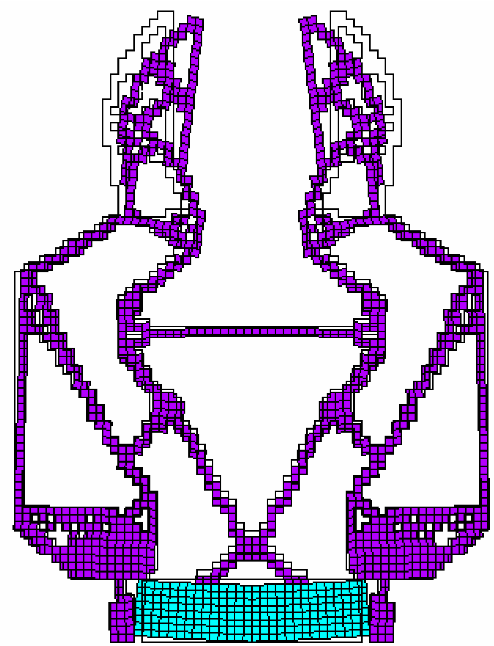


courtesy James Edward Shooter





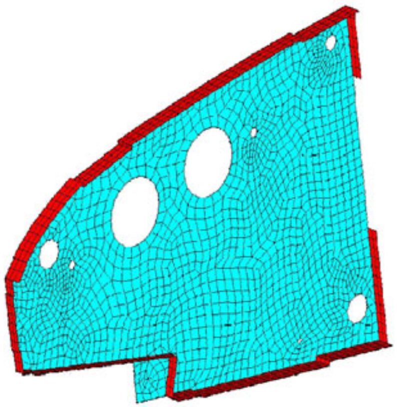
Altair Engineering website – www.altair.com

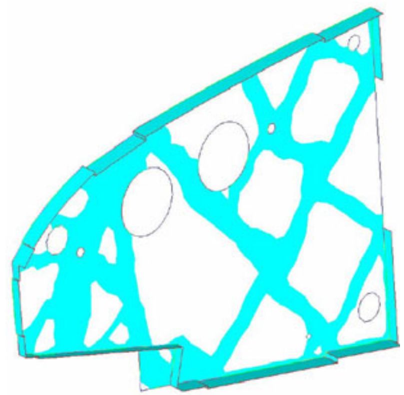


courtesy Ronny Calixto Carbonari

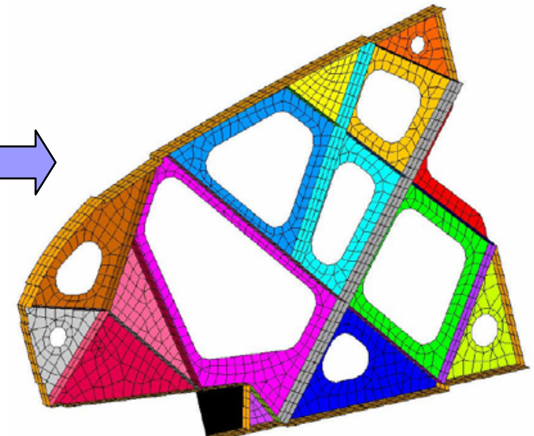
Wing Support Design Airbus

Design Domain

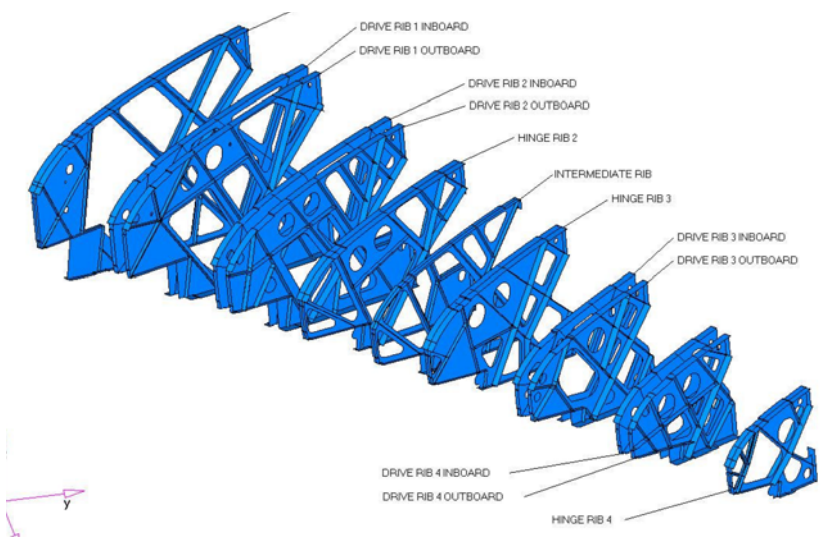




Topology Optimization Solution



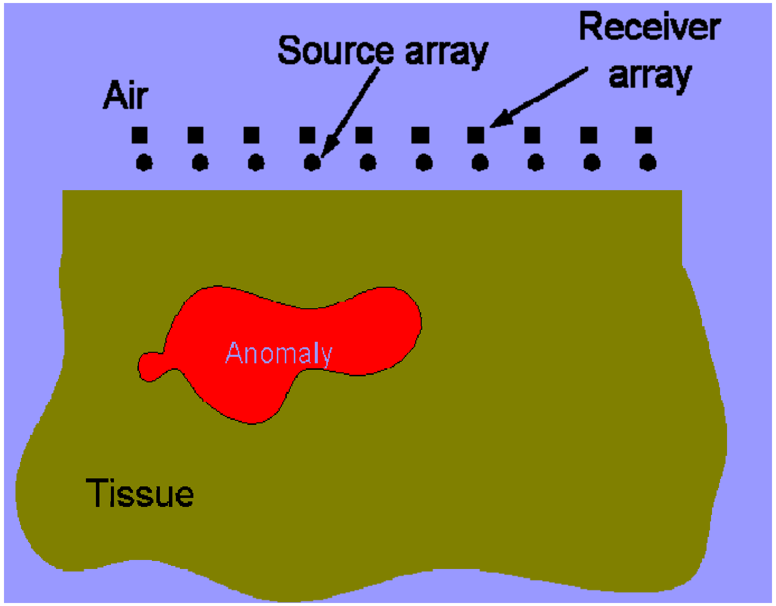
Interpretation and Verification



Fabrication

Courtesy Altair Engineering, Inc., Michigan, USA

Diffusive Tomography (with Misha Kilmer)



- Developed for medical imaging
  - Near IR not significantly absorbed by tissue, but it is highly scattered.
  - Photons diffuse through tissue in density waves
- Microstructure (electrical permittivity, magnetic permeability) is not resolvable.
- Macro-structure (optical absorption and diffusivity) *is* recoverable from light emerging from tissue
- Applications: Breast and brain imaging where local rise in absorption is tied to presence of oxygenated hemoglobin

Collaboration with Misha Kilmer, Tufts

Diffusion Forward Model

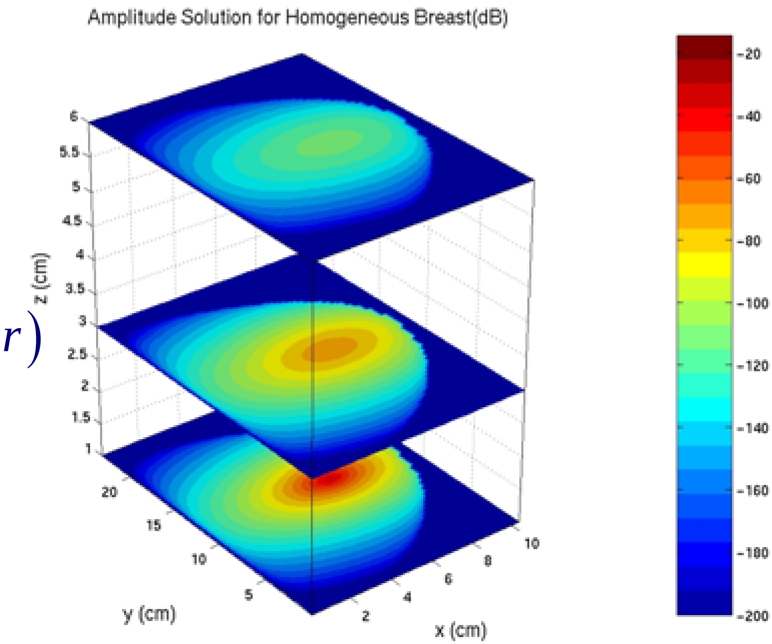
- Photon fluence due to source input s and frequency ω given by solution to the following PDE

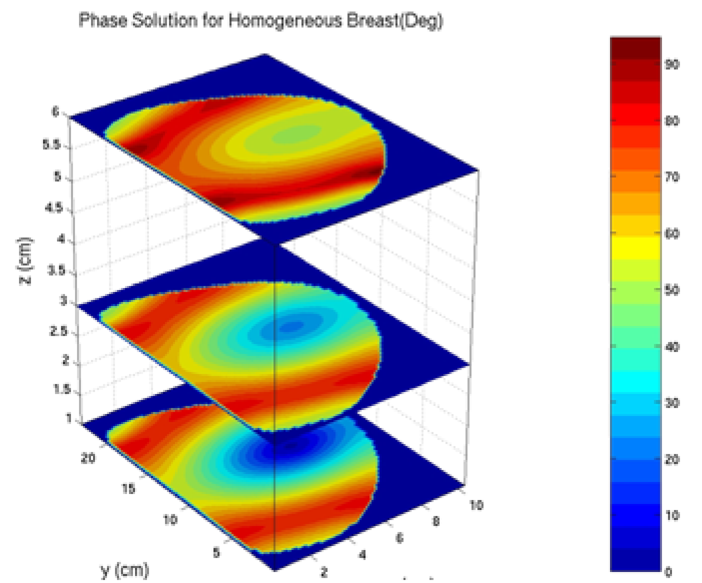
$$-\nabla\bullet\left(D(r)\nabla f_{\omega}^{s}(r)\right)+\left[\mu(r)+i\frac{\omega}{c}\right]f_{\omega}^{s}(r)=g^{s}(r)$$

  plus boundary conditions
- Desired are parameters p for diffusivity, D, and absorption, μ
- In matrix form

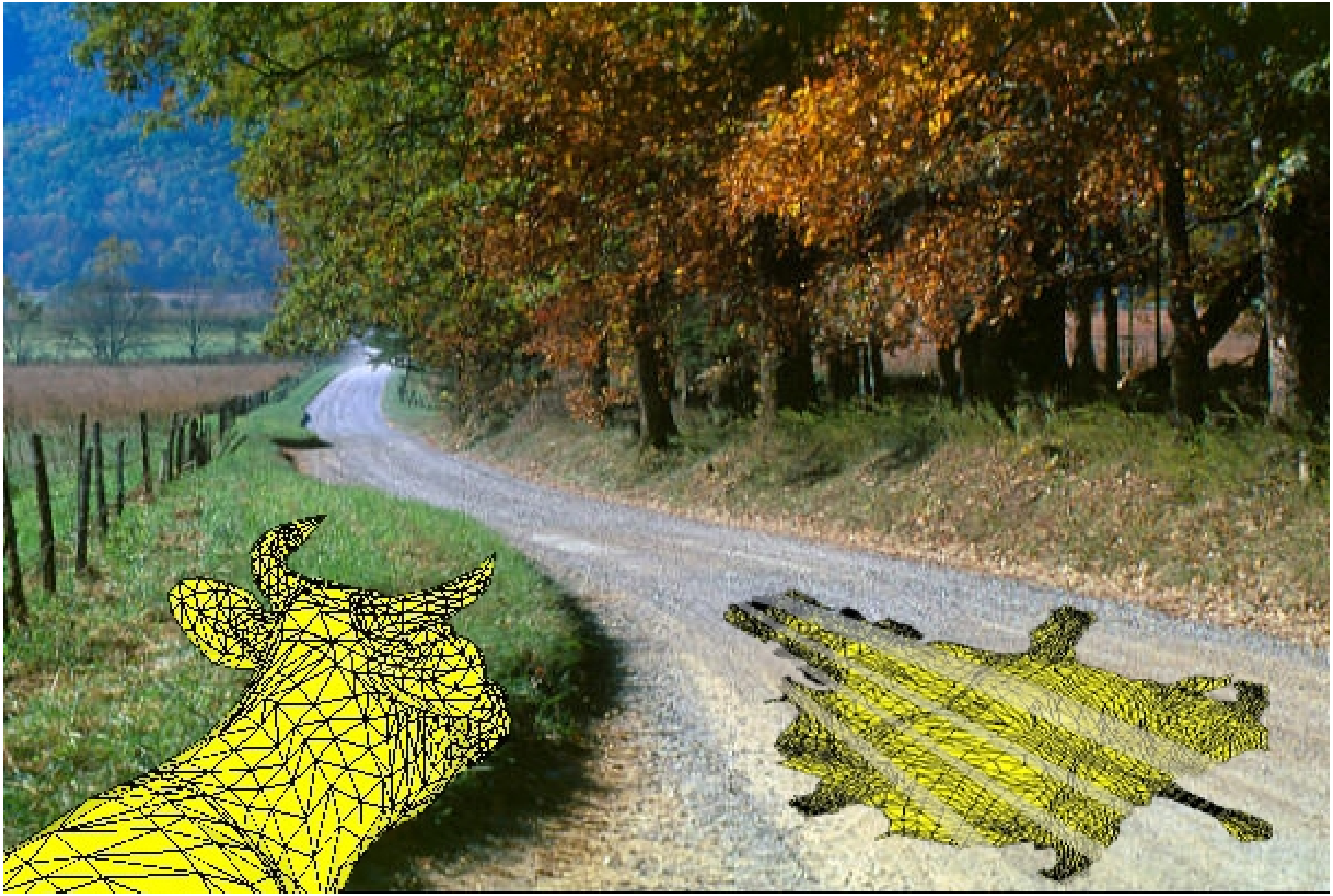
$$A_{\omega}\left(D(p),\mu(p)\right)f_{\omega}^{s}=g$$

  with A sparse, symmetric positive definite real part
- Solution at some positions known from measurements. Find parameters that match solution to data.



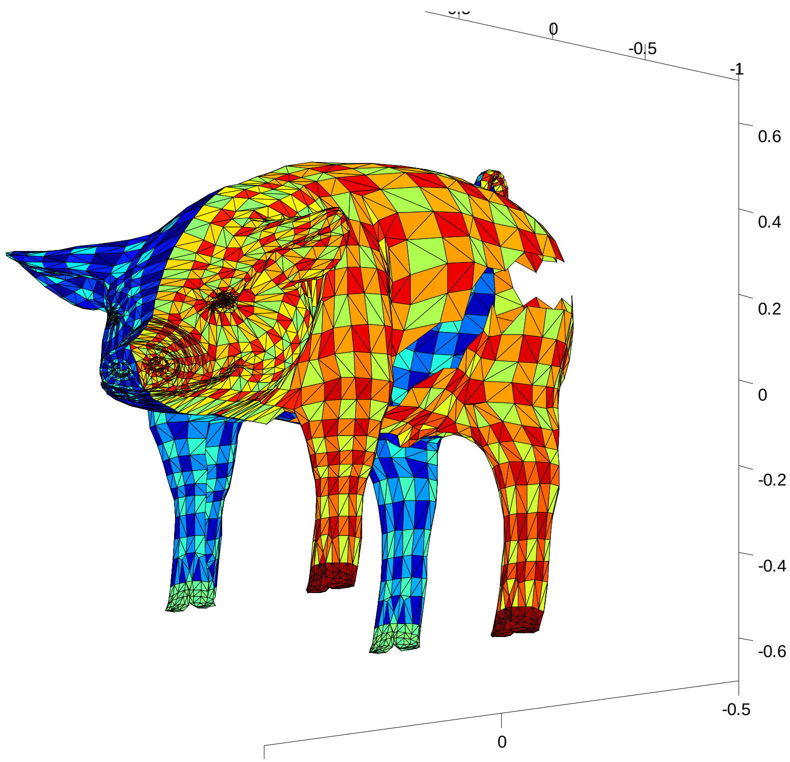


Roadkill Project

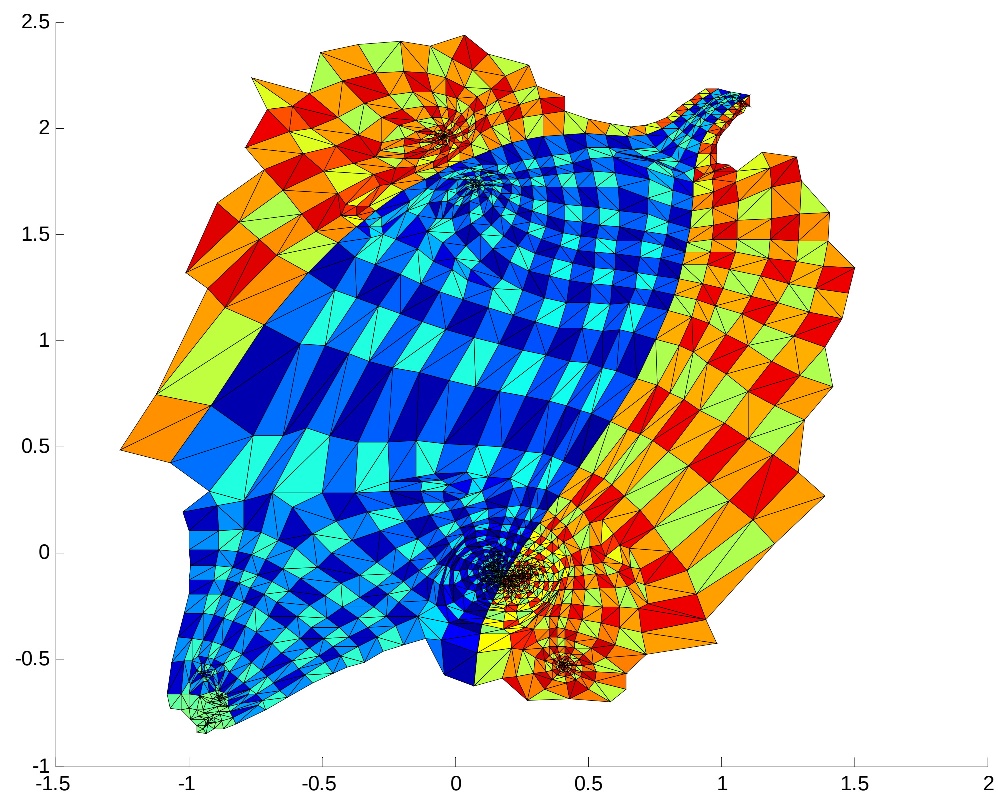


Courtesy Alla Sheffer, UBC

Surface Parameterization (with Alla Sheffer, UBC)

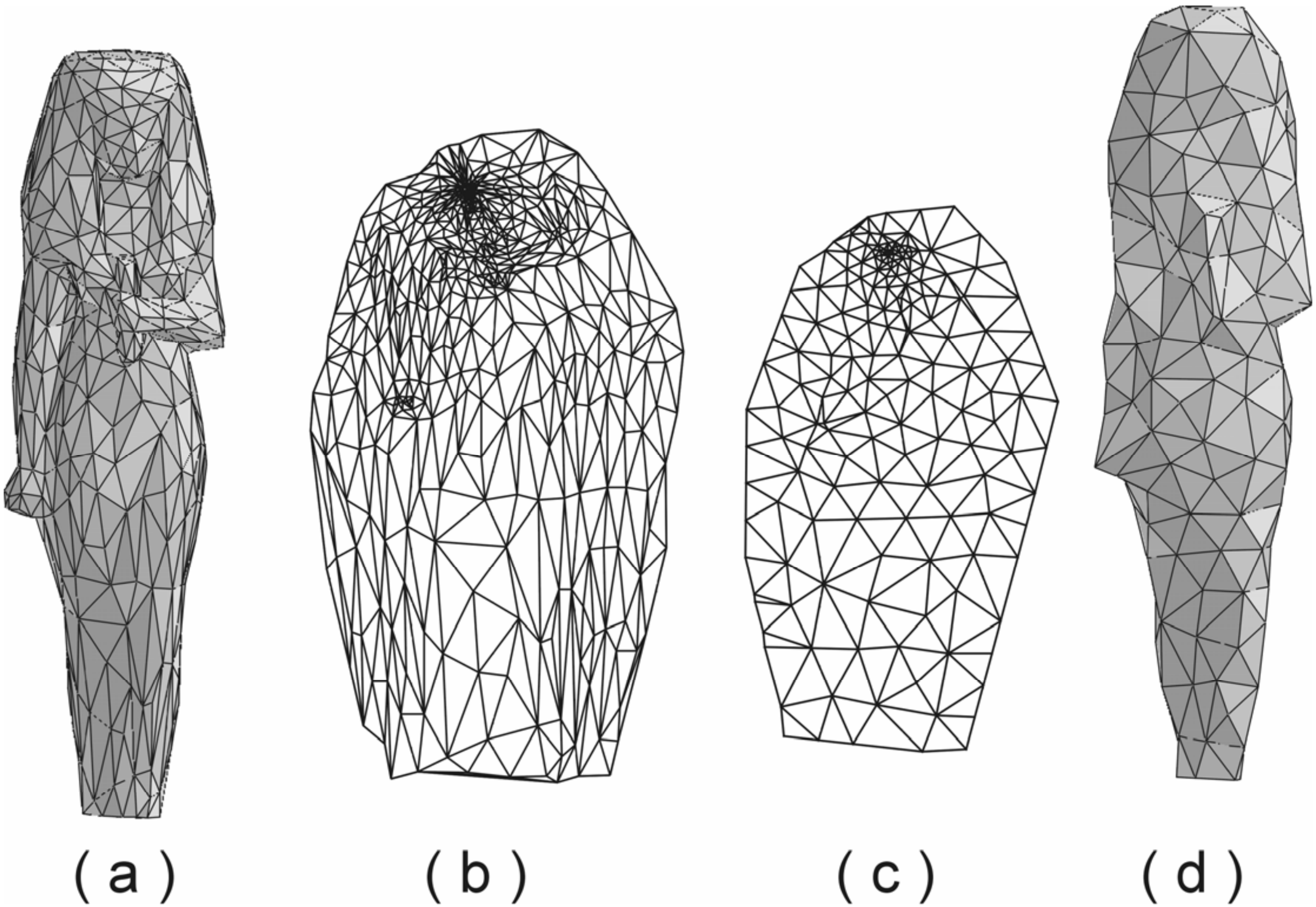


or How to Flatten a Pig?



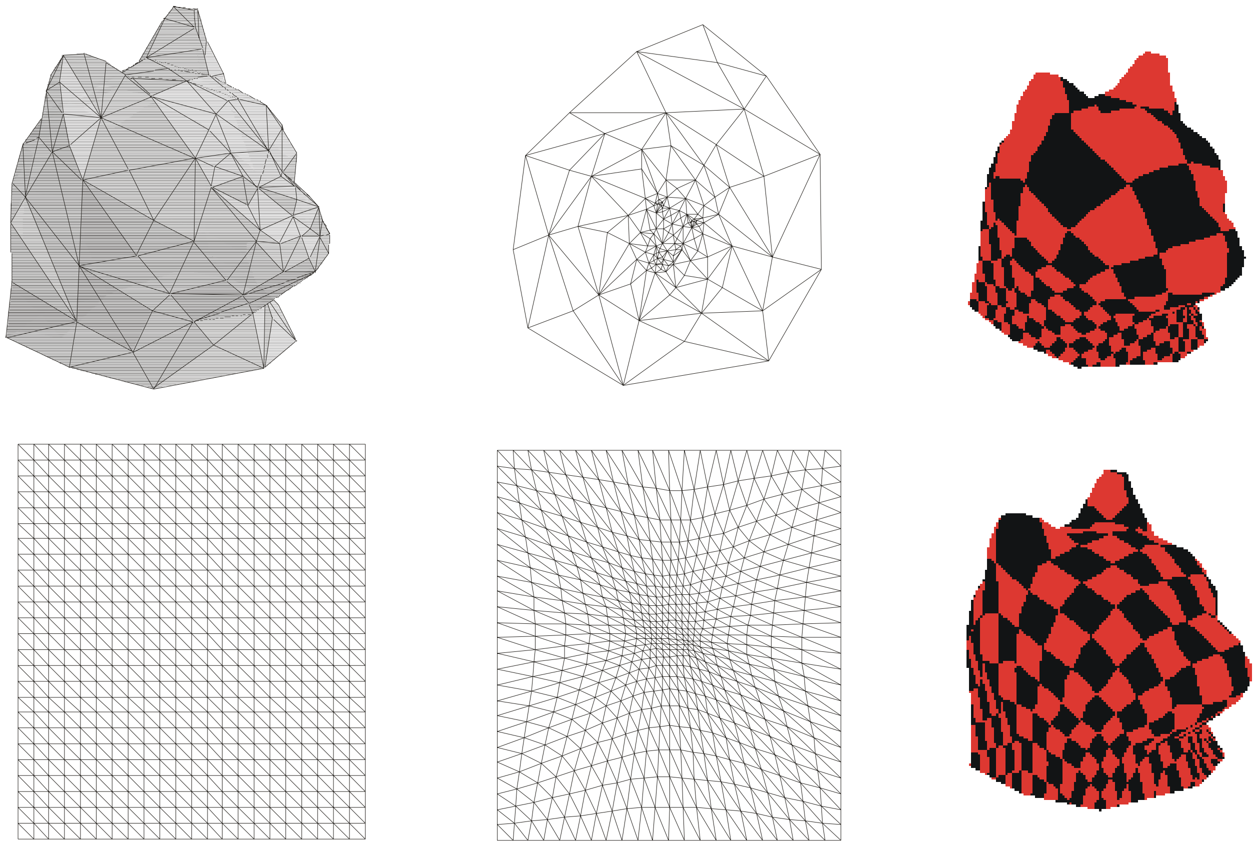
Collaboration with Alla Sheffer, UBC formerly UIUC

Example: Surface Meshing



Remeshed using 'area preserving' spacing function

Example: Texture Mapping



Angle-based Flattening

Solve mesh flattening as constrained optimization problem.

Minimize relative deformation of angles, $\sum_{i,j}\left(\alpha_{i,j}-\phi_{i,j}\right)^2/\phi_{i,j}^2$,

Subject to constraints on valid 2D mesh

1. Angles between 0 and π (orientation), hardly ever a problem (dealt with algorithmically)
2. Angles in triangle sum to π,
3. Angles around interior node sum to 2π,
4. Triangles at an interior node need to agree on edge lengths: nonlinear constraint.

$$\min\sum_{i,j}\left(\alpha_{i,j}-\phi_{i,j}\right)/\phi_{i,j}^2 \text{ subject to } \left[g_2(\alpha)\; g_3(\alpha)\; g_4(\alpha)\right]^T=0$$

Critical point of Lagrangian $L(\alpha,\lambda)=F(\alpha)+\lambda^T g(\alpha)$

Nonlinear System

Critical point of $\mathbf{L}(a,\lambda)=\boldsymbol{F}(a)+\lambda^T\mathbf{g}(a)$: $\nabla_{a,\lambda}\mathbf{L}(a,\lambda)=\mathbf{0}$

Newton iteration: $\nabla^2_{a,\lambda}\mathbf{L}(a,\lambda)\begin{bmatrix}\Delta a\\ \Delta\lambda\end{bmatrix}=-\nabla_{a,\lambda}\mathbf{L}(a,\lambda)$

Jacobian: $\begin{bmatrix}\nabla^2_a[\boldsymbol{F}(a)+\lambda^T\mathbf{g}(a)] & [\nabla_a\mathbf{g}(a)]^T\\ \nabla_a\mathbf{g}(a) & \mathbf{0}\end{bmatrix}=$

$$\begin{bmatrix}A+G_k & B^T & C_k^T\\ B & \mathbf{0} & \mathbf{0}\\ C_k & \mathbf{0} & \mathbf{0}\end{bmatrix}$$ *(symmetric and indefinite)*

$A=\mathbf{diag}\left(2w_i^j\right)$ *and* G_k *depends only on* $\mathbf{g}^{(4)}(a)$

$\boldsymbol{B}$ *depends only on* $\mathbf{g}^{(2)}(a)$ *and hence is constant (zero,one)*

C_k *depends on* $\mathbf{g}^{(3)}$ *and* $\mathbf{g}^{(4)}$ *and is partially constant (*$\mathbf{g}^{(3)}$*)*

Solving Linear Systems

- All the systems derived from these applications are nonlinear
- ‘Some’ are more nonlinear than ‘others’
- The ‘some’ require special treatment
- The ‘others’ can be solved (in straightforward fashion) by variants of Newton’s method
- Results in sequence of linear systems

- How to solve a single linear system fast?
- How to solve a thousand slowly evolving linear systems fast?

Why Iterative Methods?

Consider $N \times N$ matrix with k nonzeros/row (average), $k \ll N$

direct solver (LU):	work: $O\left(N^3\right)$	storage: $O\left(N^2\right)$
direct solver for band matrix:	work: $O\left(b^2 N\right)$	storage: $O\left(bN\right)$
2D problems: $b = O\left(N^{1/2}\right)$:	work: $O\left(N^2\right)$	storage: $O\left(N^{3/2}\right)$
3D problems: $b = O\left(N^{2/3}\right)$:	work: $O\left(N^{7/3}\right)$	storage: $O\left(N^{5/3}\right)$
sparse matrix-vector product:	work: $O\left(Nk\right)$	storage: $O\left(Nk\right)$

For large problems direct methods are impossible, and even for moderate problems they are much more expensive than iterative methods (in work and storage).

Why Iterative Methods?

Consider $N \times N$ matrix with k nonzeros/row (average), $k \ll N$

Consider iterative methods and convergence in m iterations:

- typically $m \ll N$ (independent of 2D, 3D, ... problem),
- m depends on characteristics of the problem rather than its size,
- in general m increases only as moderate function of N,
- for several problem classes constant m algorithms are known,
  - Holy Grail of Linear Solvers – Linear Cost
  - next step: model reduction → less than linear (cheating)
- for many Krylov subspace methods convergence in $m \leq N$ iterations guaranteed (in exact arithmetic).

Just in case someone asks

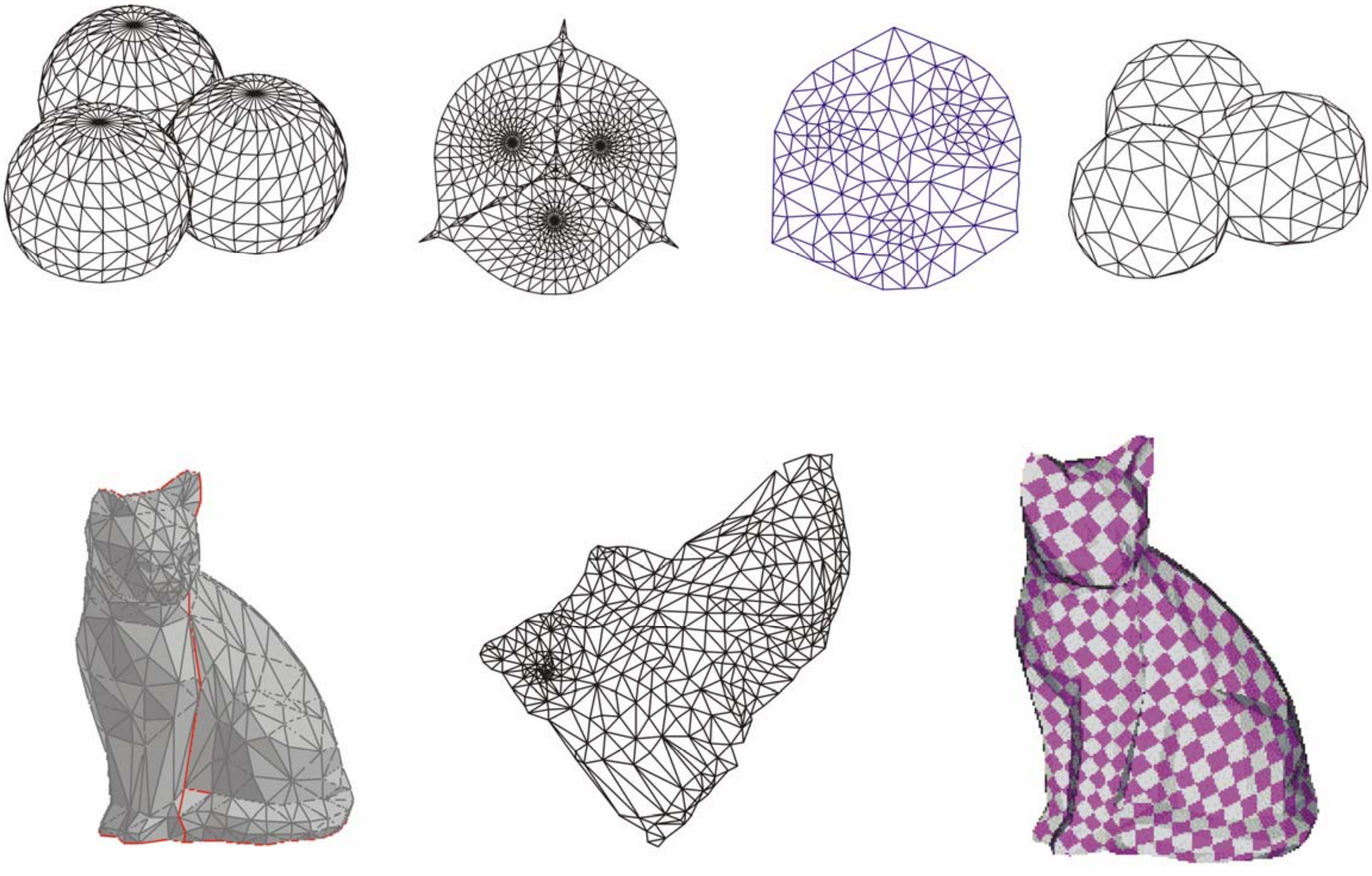
Table I. Three discretizations used for the example in Figure 6.

Problem	Mesh size	No. of unknowns (in simulation)	Solution time (h)	Optimization steps	Iterative solver
Small	$36 \times 12 \times 6$	9360	0.1	142	RMINRES(100, 10)
Medium	$84 \times 28 \times 14$	107 184	2.4	139	RMINRES(100, 10)
Large	$180 \times 60 \times 30$	1 010 160	45.7	130	RMINRES(200, 10)

Table II. Run time comparison (s) of direct and iterative solvers. The direct solver is the multifrontal, supernodal Cholesky factorization in TAUCS; the iterative solver is RMINRES with rescaling and incomplete Cholesky preconditioner and the continuation on the solver tolerance. These timings were obtained on a PC with an AMD Opteron ™252 2.6 GHz 64-bit processor, 8 GB RAM of memory, and the SuSE Linux system.

Problem size	No. of unknowns in simulation	Direct solver time (s)			Iterative solver time (s)			
		Decompose	Solve	Total	min	max	Average	IC(0)
Small	9360	0.95	0.01	0.96	0.94	2.25	1.66	0.02
Medium	107 184	179.0	0.3	179.3	21.2	71.9	50.5	3.6
Large	1 010 160	21 241	4904	26 154	254	1546	1170	26.3

Surface Parameterization



Generalized Saddle-point Problems (with Joerg Liesen)

We consider systems of the type

$$\begin{pmatrix} A & B^T \\ C & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}$$

Systems of this type arise in a variety of problems:

- Constrained optimization problems
  - o FETI (type) methods (Mike Parks)
  - o Surface parameterization
- Systems of PDEs with continuity constraints
  - o Navier-Stokes
  - o Potential flow in porous media
  - o Polycrystal plasticity – metal deformation
  - o Electrostatics / electromagnetics

Nonlinear System

Critical point of $\mathbf{L}(a,\lambda)=\boldsymbol{F}(a)+\lambda^T\mathbf{g}(a)$: $\nabla_{a,\lambda}\mathbf{L}(a,\lambda)=\mathbf{0}$

Newton iteration: $\nabla^2_{a,\lambda}\mathbf{L}(a,\lambda)\begin{bmatrix}\Delta a\\ \Delta\lambda\end{bmatrix}=-\nabla_{a,\lambda}\mathbf{L}(a,\lambda)$

Jacobian: $\begin{bmatrix}\nabla^2_a[\boldsymbol{F}(a)+\lambda^T\mathbf{g}(a)] & [\nabla_a\mathbf{g}(a)]^T\\ \nabla_a\mathbf{g}(a) & \mathbf{0}\end{bmatrix}=$

$$\begin{bmatrix}A+G_k & B^T & C_k^T\\ B & \mathbf{0} & \mathbf{0}\\ C_k & \mathbf{0} & \mathbf{0}\end{bmatrix}$$ *(symmetric and indefinite)*

$A=\mathbf{diag}\left(2w_i^j\right)$ *and* G_k *depends only on* $\mathbf{g}^{(4)}(a)$

B *depends only on* $\mathbf{g}^{(2)}(a)$ *and hence is constant (zero,one)*

C_k *depends on* $\mathbf{g}^{(3)}$ *and* $\mathbf{g}^{(4)}$ *and is partially constant (*$\mathbf{g}^{(3)}$*)*

Preconditioners

Significant body of work by Elman, Golub, Wathen, Benzi, Silvester, Gould, Nocedal, Hribar, Simoncini, Perugia, BP, …

Use splitting $A = F - E$ (dS&L):

$$\begin{pmatrix} F^{-1} & \\ & (CF^{-1}B^T)^{-1} \end{pmatrix} \begin{pmatrix} F - E & B^T \\ C & 0 \end{pmatrix} = \begin{pmatrix} I - S & N \\ M & 0 \end{pmatrix}$$

$$M = \left(CF^{-1}B^T\right)^{-1} C,\ N = F^{-1}B^T,\ MN = I,\ (NM)^2 = NM$$

Oblique projection: $NM = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} 0 & \\ & I \end{pmatrix} \begin{pmatrix} U_1 & U_2 \end{pmatrix}^{-1}$

Principal angles ($\omega_i = \cos \varphi_i$) between $\mathrm{null}(NM)$ and $\mathrm{range}(NM)$ play important role in eigenvalue bounds.

Preconditioners

Preconditioned system: $\begin{pmatrix} I-S & N \\ M & 0 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \hat{f} \\ \hat{g} \end{pmatrix}$

Eigenvalues: $|\lambda_S - \lambda| \le 1.5\left(\frac{1+\omega_{\max}}{1-\omega_{\max}}\right)^{1/2} \|S\|,\ \lambda \in \left\{1, \left(1 \pm \sqrt{5}\right)/2\right\}$

Further splitting gives fixed point iteration that depends only on x: $x_{k+1} = (I - NM)Sx_k + \tilde{f}$

Solves *related system* $\left(I - (I - NM)S\right)x = \tilde{f}$

Eigenvalues $|1 - \lambda_R| \le \left(1 - \omega_{\max}^2\right)^{-1/2} \|S\|$

Preconditioning

Consider the following choice of blocks

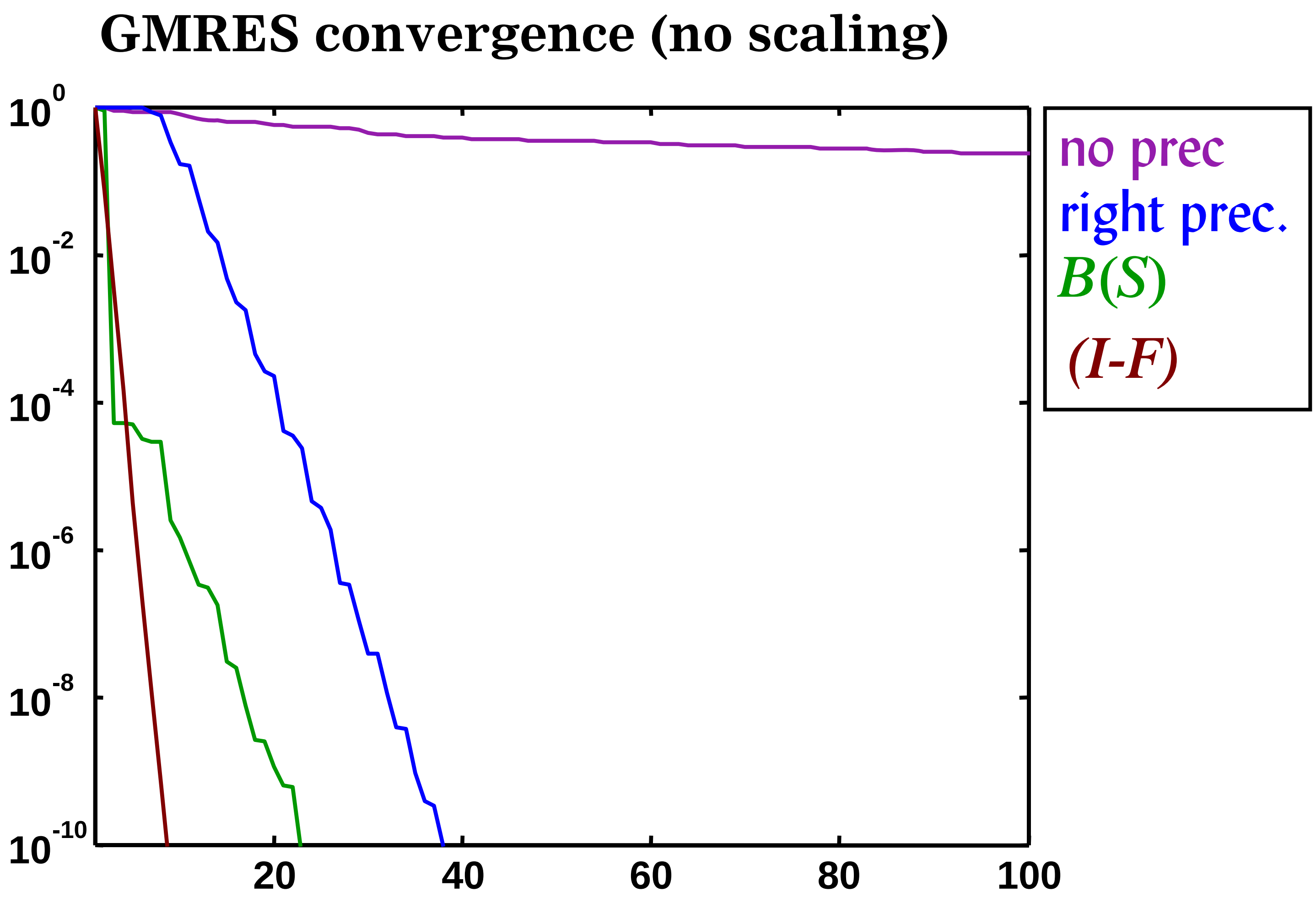
$$\begin{bmatrix} F - E_k & C_k^T \\ C_k & 0 \end{bmatrix} \text{ with } F = \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \text{ and } E_k = \begin{bmatrix} G_k & 0 \\ 0 & 0 \end{bmatrix}.$$

Explicit inverse of F known (very good splitting).

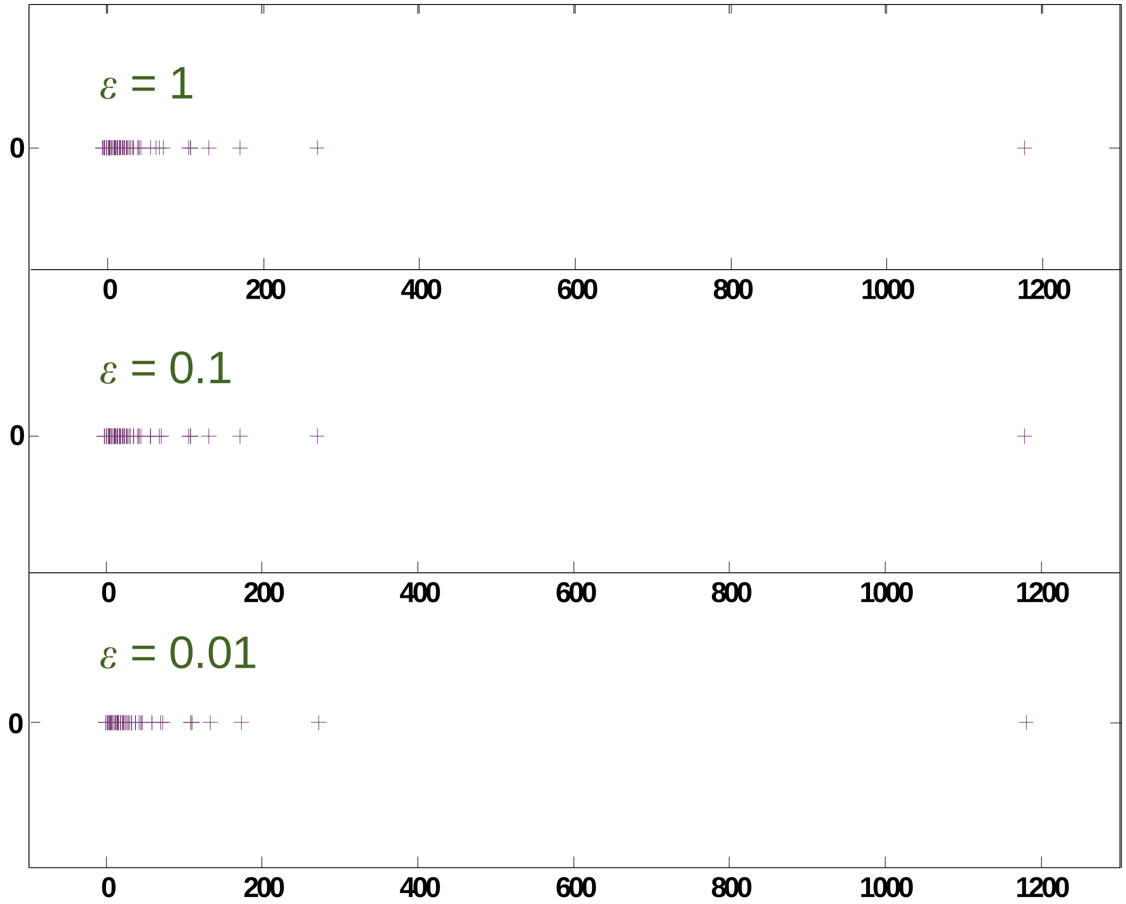
$$\begin{bmatrix} I - S_k & N_k \\ M_k & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix} \quad \text{and} \quad \left(I - \left(I - N_k M_k\right) S_k\right) x = \tilde{f}$$

$$|\lambda_S - \lambda| \le 1.5 \left(\frac{1 + \omega_{\max}}{1 - \omega_{\max}} \right)^{1/2} \|S\| \text{ and } |1 - \lambda_R| \le \left(1 - \omega_{\max}^2\right)^{-1/2} \|S\|$$

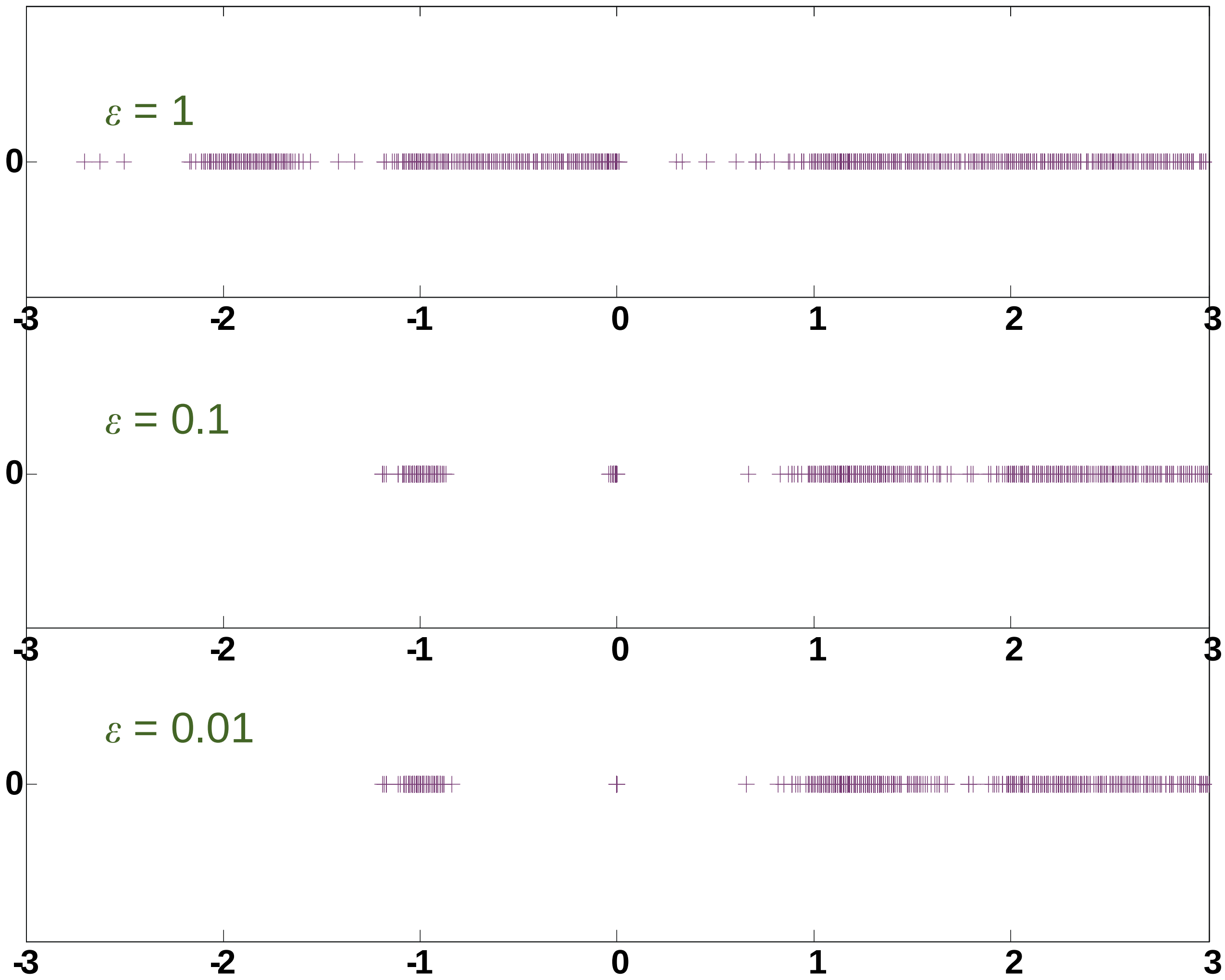
Convergence "Three Balls"



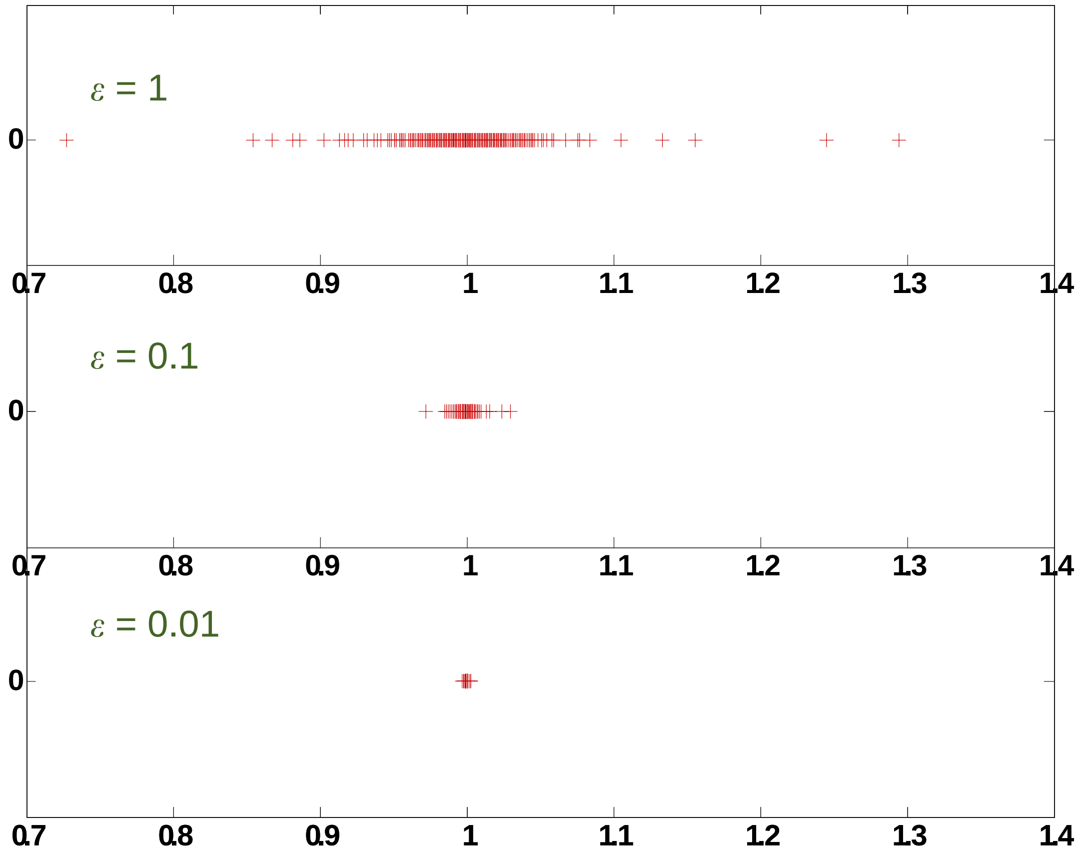
Eigenvalues Original Matrix



Eigenvalues Original Matrix – Closer Look



Eigenvalues of preconditioned system



Test Case: Tomography (with Misha Kilmer)

- Reconstruct medium by measuring how signals propagate
- Parameterize medium and optimize parameters by matching measured signal with computed signal (at receivers)
- Forward problem $-\nabla \bullet \left(a(x,\omega;p) \nabla u_j \right) + m(x,\omega;p) u = f_j$
- Have to solve (forward) problem many times (optimization)
- Problem is Hermitian for zero frequency and nonzero frequency gives imaginary shift
- Multiple sources give multiple right hand sides
- Nonlinear least squares/Gauss-Newton with line search
- First few steps fix background parameters, later steps mainly change shape of tumor: 'diffusion' jump in small region
- Change in matrix concentrated in high frequency modes
- Lot of opportunity to exploit structure

Iteration Counts for Recycle Version