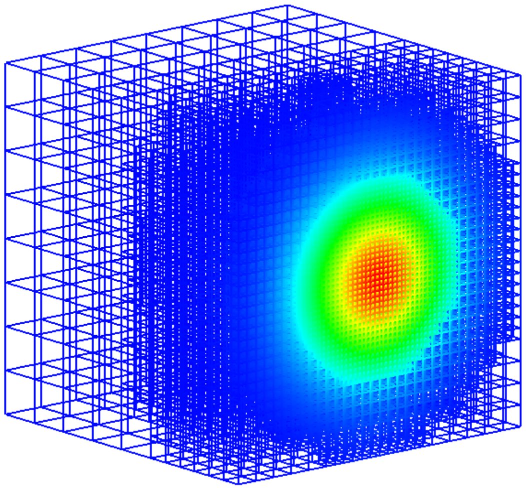
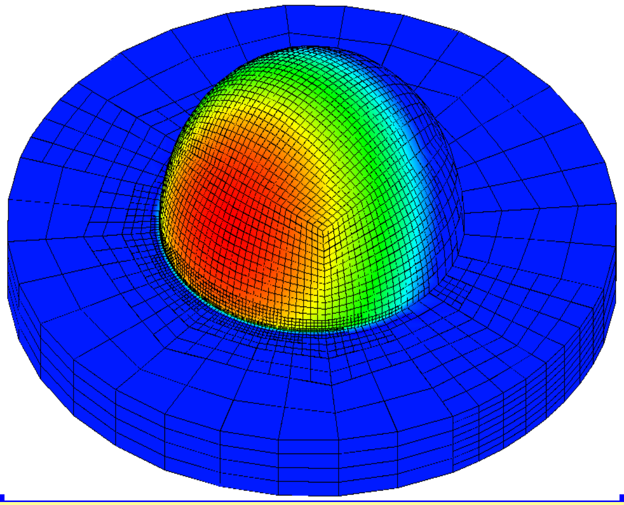
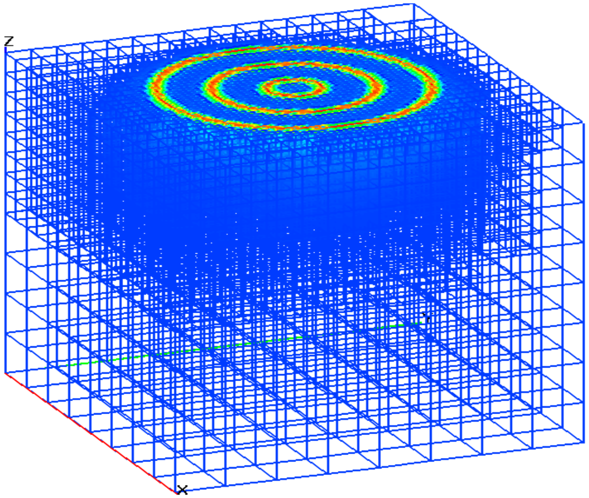
Numerics for Inverse Problems in Biomedical Imaging

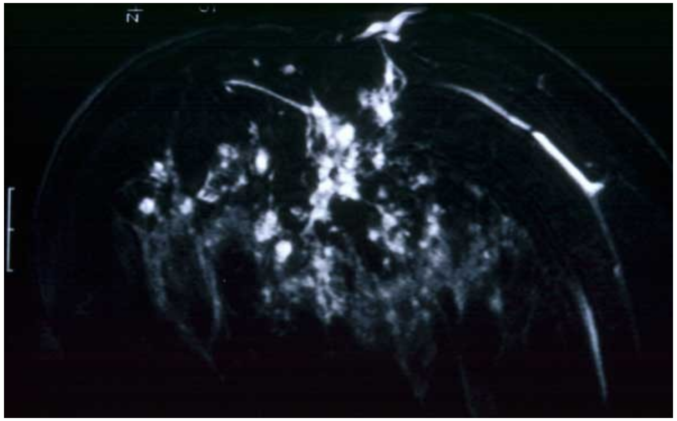
Wolfgang Bangerth

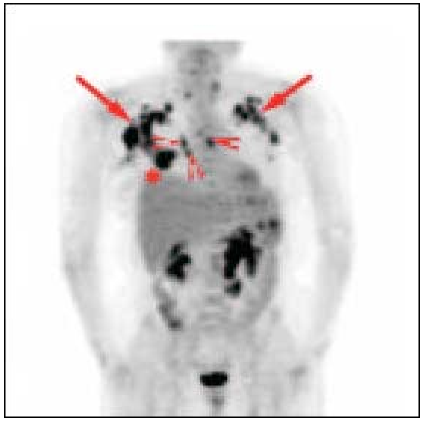
Department of Mathematics
Texas A&M University

Background

- Existing biomedical imaging techniques for tumor diagnosis do not show what we are interested in, or do so only with drawbacks:

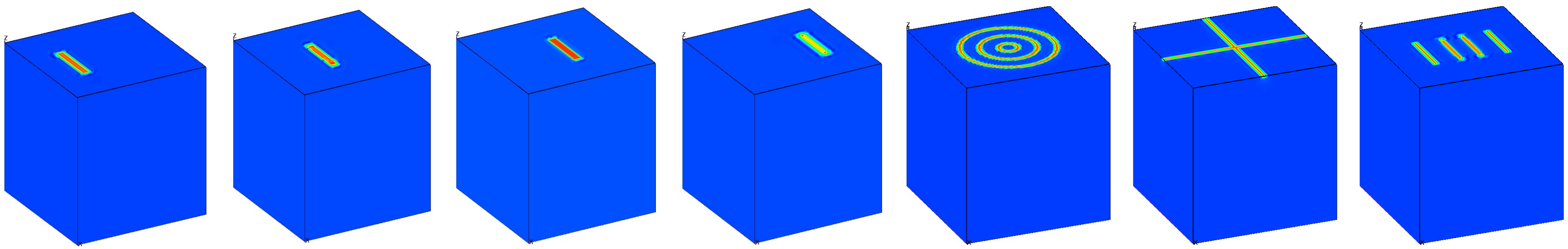




- What we need is a method that
  a) shows actual tumor cells, not secondary effects of tumors
  b) is harmless
  c) yields high-resolution images
  d) is mathematically stable (robust in the presence of noise)
  e) is fast
  f) is cheap

General Setting of Inverse Problems

- We have data from several experiments made on a subject:

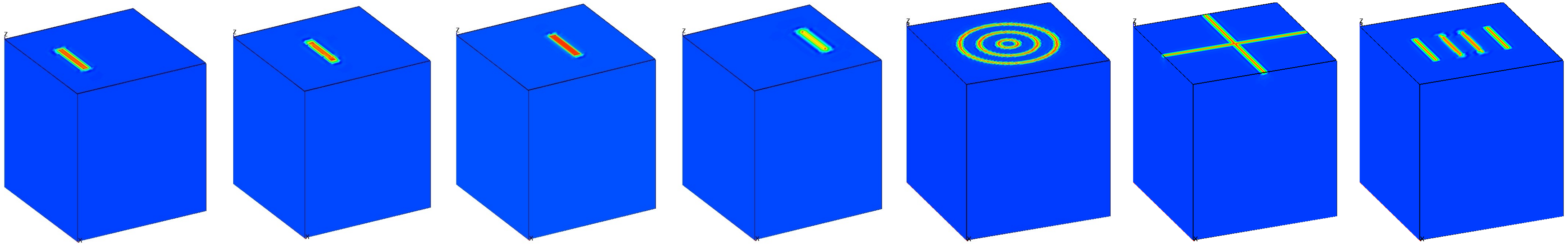


- We know how the system behaves when illuminated: We can predict the outcome of every experiment by a (possibly nonlinear) PDE:

$$A^i(q, u^i)(\varphi^i) = 0 \qquad \forall \varphi^i \qquad\qquad [e.g.\ A(q,u)(\varphi) = (q\nabla u, \nabla\varphi) - (f, \varphi)]$$

General Setting of Inverse Problems

- We have data from several experiments made on a subject:



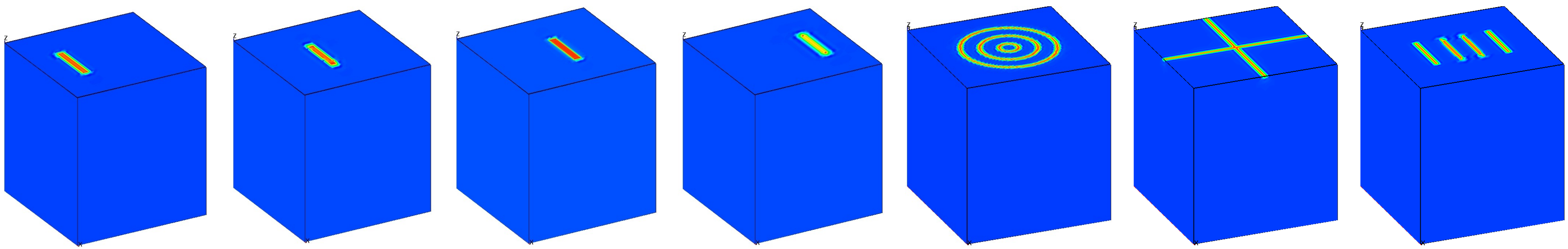
- We know how the system behaves when illuminated: We can predict the outcome of every experiment by a (possibly nonlinear) PDE.

  In reality, the equations and boundary conditions for the particular application I will show results for are

$$
\begin{aligned}
-\nabla\cdot D_u(q)\nabla u+\left(\frac{i\omega}{c}+k_u(q)\right)u &= 0 & \qquad 2\mathrm{D}_u(q)\frac{\partial u}{\partial n}+\gamma u &= S\\
-\nabla\cdot D_v(q)\nabla v+\left(\frac{i\omega}{c}+k_v(q)\right)v &= \beta q u & \qquad 2\mathrm{D}_v(q)\frac{\partial v}{\partial n}+\gamma v &= 0
\end{aligned}
$$

General Setting of Inverse Problems

- We have data from several experiments made on a subject:



- We know how the system behaves when illuminated: We can predict the outcome of every experiment by a (possibly nonlinear) PDE:

$$A^i(q,u^i)(\varphi^i)=0 \qquad \forall \varphi^i \qquad\qquad [e.g.\ A(q,u)(\varphi)=(q\nabla u,\nabla\varphi)-(f,\varphi)]$$

- All experiments depend on a common set *q(x)* of distributed parameters (e.g. absorption and scattering coefficients)

- **Goal:** Have efficient tools to identify *q(x)* from all these measurements!

Overview

- How do we formulate inverse problems?
- Why are inverse problems hard computationally?
- Nonlinear and linear solvers
- Inequality constraints and adaptive meshing
- Numerical examples for membrane displacement
- Numerical examples for refraction imaging
- Numerical example for optical tomography imaging of tumors
- Conclusions and outlook

General Setting of Inverse Problems

- **Goal:** Have efficient tools to identify *q(x)* from all measurements!

- **How:**

  1) We can predict measurements if we knew *q*

  2) Starting with an initial guess, vary *q* until our predicted measurements match the actual measurements best

  3) Compare measurements using an objective functional (e.g. a norm)

- **Consequences:**

  The inverse problem is posed as a minimization problem!
  The forward problem (i.e. the PDE) is only a side condition!

The Maths

A model problem: Consider a membrane of variable thickness *q(x)* subjected to a known force *f(x)*

- *z(x)* measured deflection
- *q(x)* unknown coefficient
- $u_q(x)$ expected (predicted) displacement for coefficient *q(x)*

- State equation: $-\nabla \cdot q \nabla u_q = f \qquad + B.C.$

Program: Determine the unknown coefficient by varying *q(x)* until we have found $u_q(x)$ that matches the observation *z(x)* best!

The Maths

Program: Determine the unknown coefficient by varying *q(x)* until we have found $u_q(x)$ that matches the observation *z(x)* best!

Formulation as a constrained minimization problem when multiple data sets are available:

$$
\begin{array}{ll}
\textit{minimize} & J(u,q) = \sum_{i=1}^{N} m^i (M^i u^i - z^i) + \beta r(q) \\
\textit{subject to} & A^i(q, u^i)(\cdot) = 0, \qquad i = 1 \ldots N
\end{array}
$$

Challenges

- Have a *mathematical* framework that allows for such inversion problems
- Have *efficient computational* techniques for each step in this framework:
  - nonlinear solvers
  - inner linear solvers for forward/adjoint problems

  - inclusion of bounds on parameters

  - efficient discretizations
  - error estimates and adaptivity for the forward/adjoint problems

  - regularization techniques
  - error estimates for the parameters

  - algorithms that run in parallel and within the allowed time

  - yield optimal accuracy of the reconstructed parameter map despite the ill-posedness of the problem

The Maths

Lagrangian formulation:

Set $x = \{u, \lambda, q\}$, introduce Lagrange function

$$L(x) = \frac{1}{2}\sum_{i=1}^{N} m^i(M^i u^i - z^i) + \beta r(q) + \sum_{i=1}^{N} A^i(q, u^i)(\lambda^i)$$

For example, for a single Laplace problem:

$$L(x) = \frac{1}{2}\|u - z\|^2 + \frac{\beta}{2}\|q\|^2 + (q\nabla u, \nabla\lambda) - (f, \lambda)$$

Optimum then stationary point of $L(x)$:

$$\nabla_x L(x)(y) = 0 \quad \forall y \longleftrightarrow \begin{aligned} (q\nabla\lambda, \nabla\psi) &= -(u - z, \psi) \\ (\beta q + \nabla u \cdot \nabla\lambda, \chi) &= 0 \\ (q\nabla u, \nabla\varphi) &= (f, \varphi) \end{aligned}$$

Solution of Optimality Condition

$$\nabla_x L(x)(y) = 0 \quad \forall y \longleftrightarrow \begin{aligned} (q\nabla\lambda, \nabla\psi) &= -(u-z, \psi) \\ (\beta q + \nabla u \cdot \nabla\lambda, \chi) &= 0 \\ (q\nabla u, \nabla\varphi) &= (f, \varphi) \end{aligned}$$

Solution: Newton-type iteration on nonlinear continuous problem

$$\nabla_x^2 L(x_k)(\delta x_k, y) = -\nabla_x L(x_k)(y) \quad \forall y,$$

- Discretize by FEM with different grids and ansatz spaces for u/λ (fine mesh, Q_p elements) and q (coarse mesh, DGQ_0 elements)
- Solve for Newton direction by Schur complement formulation
- Use line search techniques to determine step length
- If sufficient progress on one grid, then evaluate a posteriori error estimator and refine state/adjoint and parameter grids

Solution of Optimality Condition

Solution of discretized Newton steps:

- Matrix representation of

$$\nabla_x^2 L(x_k)(\delta x_k, y) = -\nabla_x L(x_k)(y) \quad \forall y,$$

  is

$$\begin{pmatrix} M & A & B \\ A^T & 0 & C \\ B^T & C^T & \beta R \end{pmatrix} \begin{pmatrix} \delta u \\ \delta \lambda \\ \delta q \end{pmatrix} = -\begin{pmatrix} F_u \\ F_\lambda \\ F_q \end{pmatrix}$$

- Can be *very* large (several 10^7), in particular with multiple experiments
- Symmetric but indefinite
- Extremely ill-conditioned: $\kappa = O(h^{-6})$ and up to 10^{13}!

Solution of Optimality Condition

Rather: Use Gauss-Newton method:

- Drop B

$$\begin{pmatrix} M & A & \cancel{B} \\ A^T & 0 & C \\ \cancel{B^T} & C^T & \beta R \end{pmatrix} \begin{pmatrix} \delta u \\ \delta \lambda \\ \delta q \end{pmatrix} = - \begin{pmatrix} F_u \\ F_\lambda \\ F_q \end{pmatrix}$$

- Consider Schur complement

$$\left(\beta R + C^T A^{-T} M A^{-1} C\right) \delta q = F_q - C^T A^{-1} \left(F_u - M A^{-1} F_\lambda\right)$$

$$A \delta u = F_\lambda - C \delta q$$

$$A^T \delta \lambda = F_u - M \delta u$$

- Much smaller systems (several 1000 to 10,000)
- Schur complement now positive definite
- Better condition number: $\kappa = O\left(h^{-4}\right)$ instead of $\kappa = O\left(h^{-6}\right)$

Solution of Optimality Condition

Main problem now: invert

$$S = \beta R + \sum_i C_i^T A_i^{-T} M_i A_i^{-1} C_i$$

for δq.

Challenges:

- S is still expensive to compute, and ill-conditioned
- Each multiplication with S requires $2N$ solutions of the underlying PDE: **May require several 10,000 to 100,000 PDE solutions!**
- Preconditioning, multigrid anyone?

Challenges

- Have a *mathematical* framework that allows for such inversion problems

- Have *efficient computational* techniques for each step in this framework:
  - nonlinear solvers
  - inner linear solvers for forward/adjoint problems

  - inclusion of bounds on parameters

  - efficient discretizations
  - error estimates and adaptivity for the forward/adjoint problems

  - regularization techniques
  - error estimates for the parameters

  - algorithms that run in parallel and within the allowed time

  - yield optimal accuracy of the reconstructed parameter map despite the ill-posedness of the problem

Inequality Constraints

Often constraints like

$$q_0(x) \le q(x) \le q_1(x)$$

are known. Including them can stabilize inversion, and make result physical.

Question: *How can we use them?*

Idea: After discretization, use an active-set-like strategy on parameters.

Inequality Constraints

Approach: Before each Newton step, determine those parameters that

- are already at their bounds
- that we expect to move out of the feasible region (using a heuristic)

Fix these parameters, i.e. enforce $\delta q_i = 0$ for them.

That means, solve

$$S \delta q = -J, \qquad s.t.\ X \delta q = 0$$

$$\downarrow$$

$$\hat{S} \delta q = -\hat{J}$$

with reduced matrix and r.h.s. with some rows/columns deleted.

This procedure is also simple to perform even if S not known explicitly, but only through matrix-vector product.

Mesh Refinement

At the end of each Gauss-Newton step:

- Check if we are still making sufficient progress on the present mesh
- If this is not the case, evaluate an error indicator and refine both the meshes on which we discretize the state/adjoint variables as well as the mesh for the parameter
- Derivation of error estimates is complicated by the ill-posedness of the problem (i.e. the lack of stability in the problem)
- Duality-based error estimates are a way out here:

$$\begin{aligned} J(x) - J(x_h) = & \frac{1}{2} \sum_K (\rho_u, \lambda - \lambda_h)_K + \textit{jump term} \\ & + (\rho_\lambda, u - u_h)_K + \textit{jump term} \\ & + (\rho_q, q - q_h)_K \\ & + R \end{aligned}$$

$$\begin{aligned} \rho_u &= f + \nabla \cdot q_h \nabla u_h \\ \rho_\lambda &= (u - z) + \nabla \cdot q_h \nabla \lambda_h \\ \rho_q &= \beta q + \nabla u_h \cdot \nabla \lambda_h \\ R &= -\frac{1}{12} (e_q \nabla e_u, \nabla e_\lambda) \end{aligned}$$

Numerical Examples

- *Steady-state membrane of variable stiffness, subjected to known (distributed) force:* Measure deflection everywhere, and recover stiffness parameter

$$N = 1 \qquad \begin{aligned} A(q,u)(\varphi) &= (q\nabla u, \nabla\varphi) - (f, \varphi) \\ m(Mu - z) &= \frac{1}{2}\|u - z\|_{\Omega}^2 \end{aligned}$$

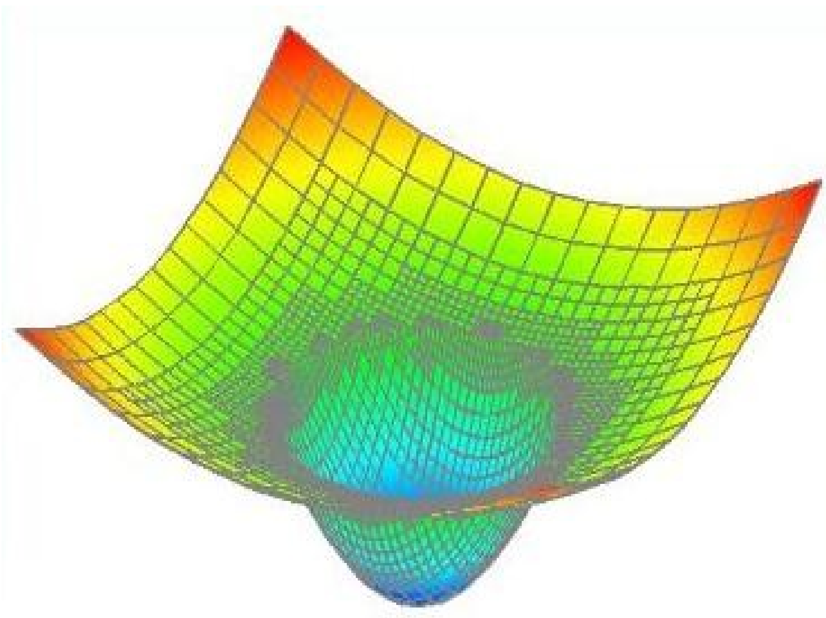
- *Helmholtz equation:* Excite material by injecting energy at one boundary, and measure at another boundary

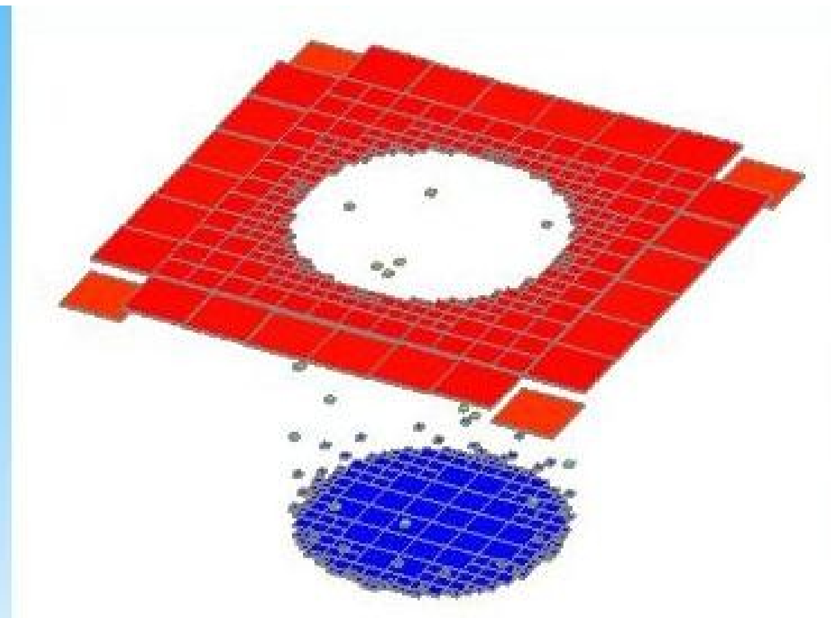
$$N \gg 1 \qquad \begin{aligned} A^i(q,u^i)(\varphi^i) &= (q\nabla u^i, \nabla\bar{\varphi}^i) - k_i^2(u^i, \bar{\varphi}^i) + b.c. \\ m^i(M^i u^i - z^i) &= \frac{1}{2}\|u^i - z^i\|_{\Sigma}^2, \qquad \Sigma \subset \partial\Omega \end{aligned}$$

- *Optical tomography:* Identify dye concentrations for cancer imaging using infrared light

Laplace example

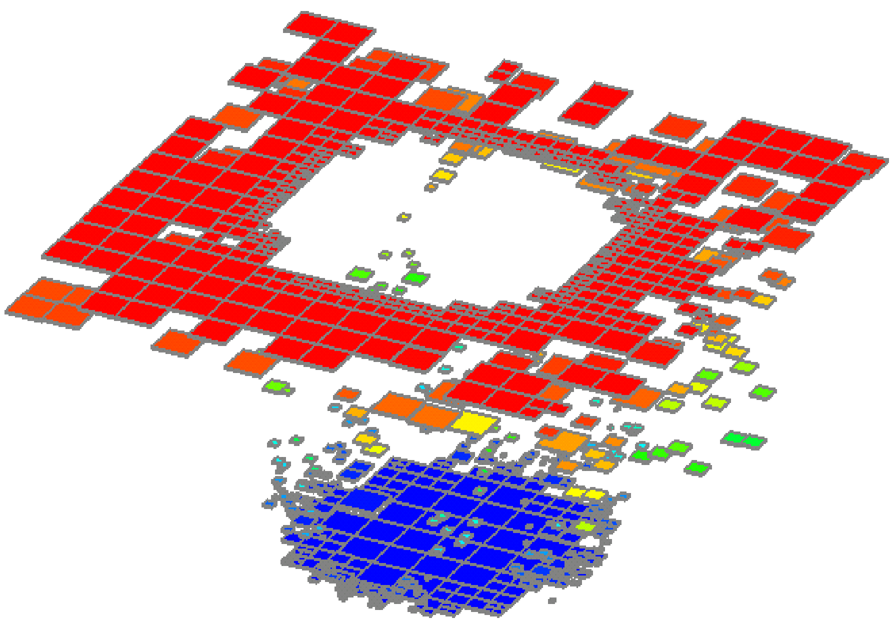
Exact solution generated from a discontinuous parameter:



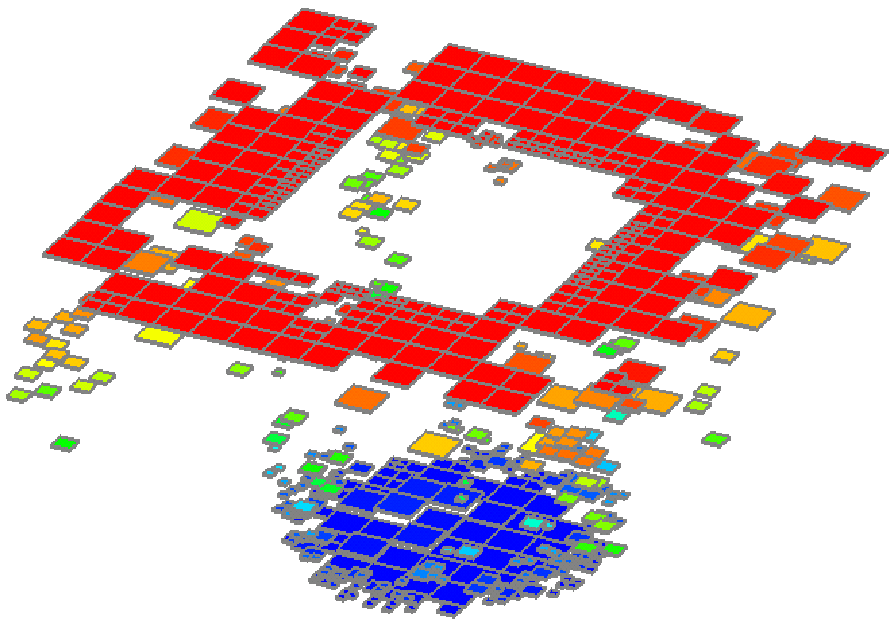


Laplace example

Same example, but this time with noise in the measurement:



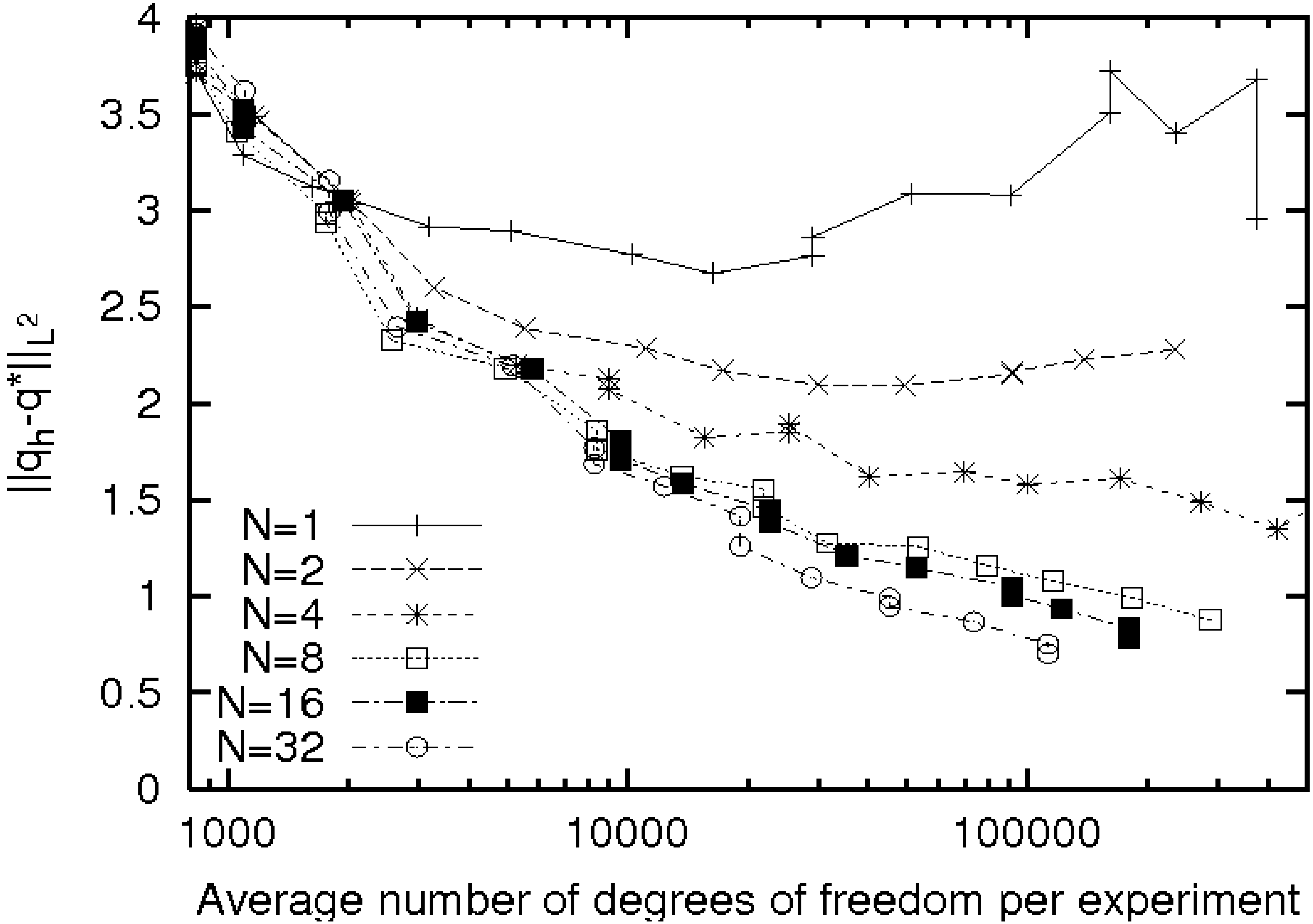
Recovered coefficient *q*
ε=1%



Recovered coefficient *q*
ε=2%

Laplace example

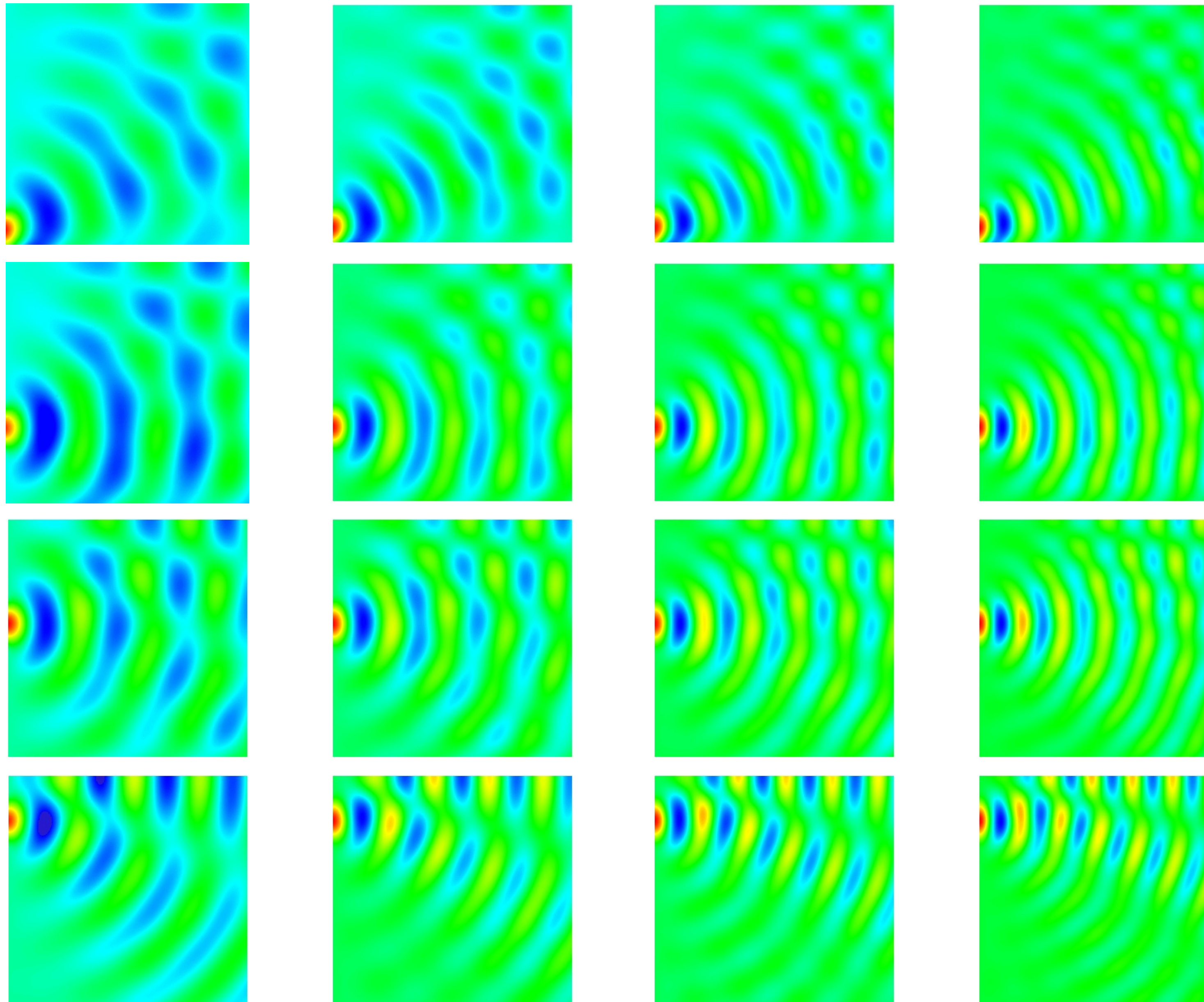
Results for different numbers of experiments



Transmission tomography example

4 different frequencies: k^2=25, 30, 35 40

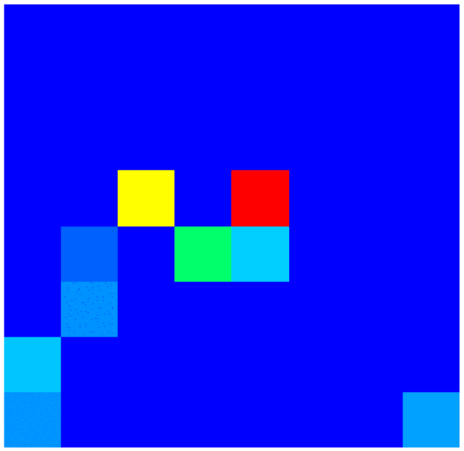
8 different source locations along left boundary



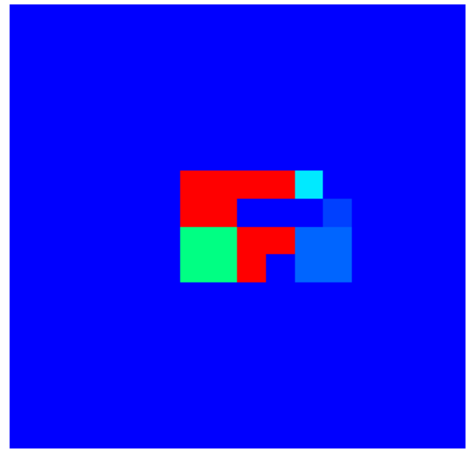
Total:
32 different
experiments

Transmission tomography example

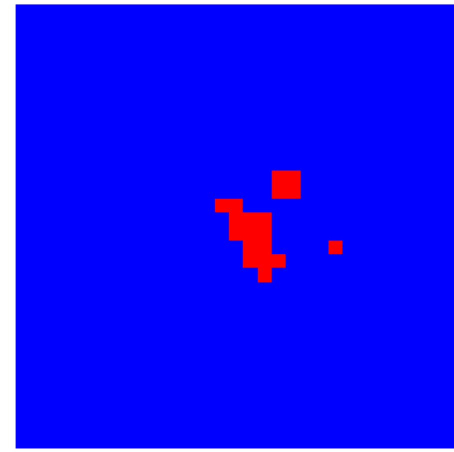
Example of a 2D reconstruction using sound transmission



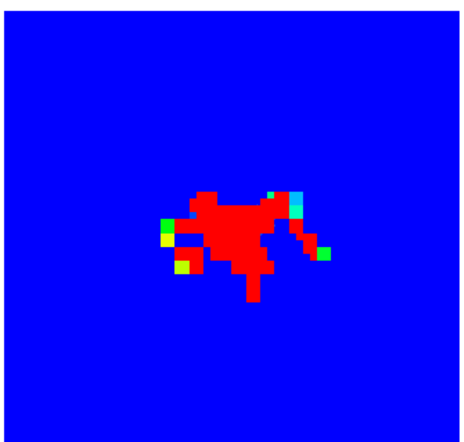
After 5 Newton steps, 64 parameters



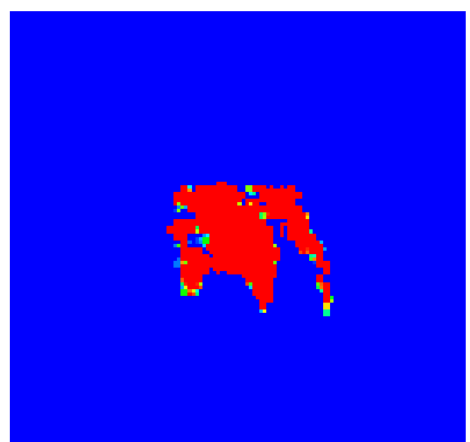
After 10 Newton steps, 61 parameters



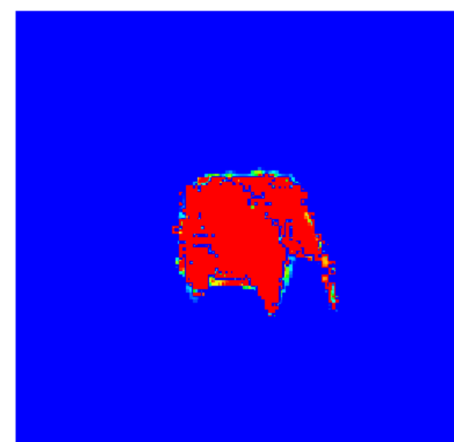
After 15 Newton steps, 118 parameters



After 20 Newton steps, 235 parameters



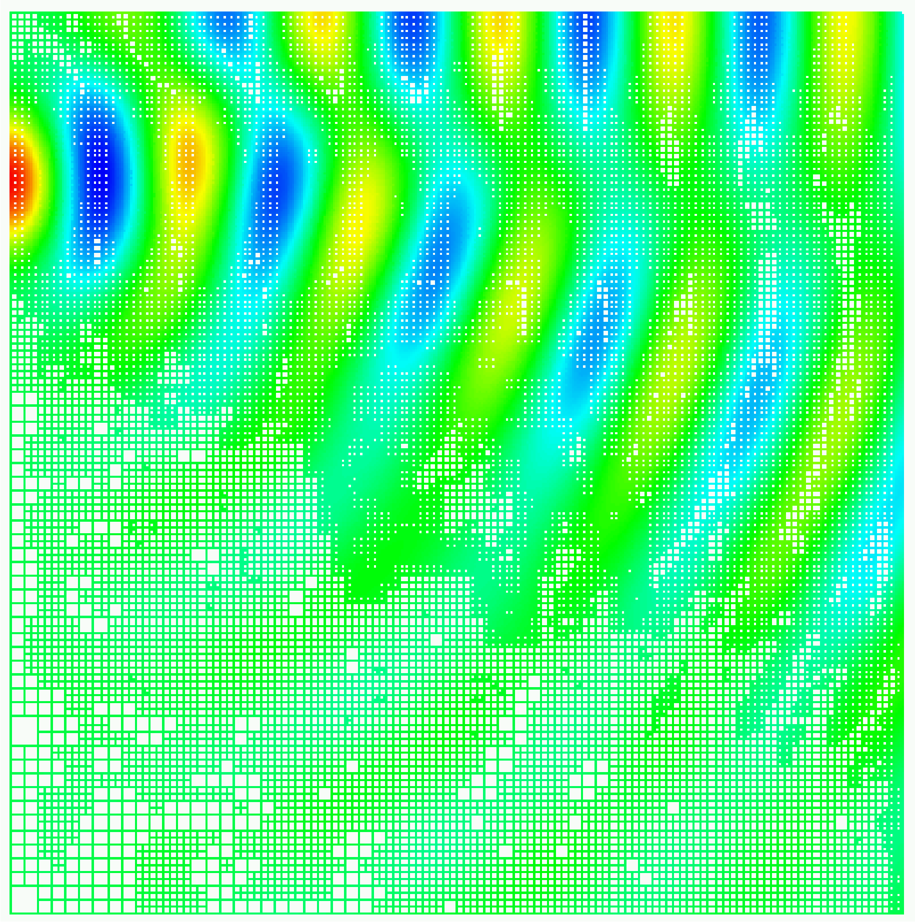
After 25 Newton steps, 1290 parameters



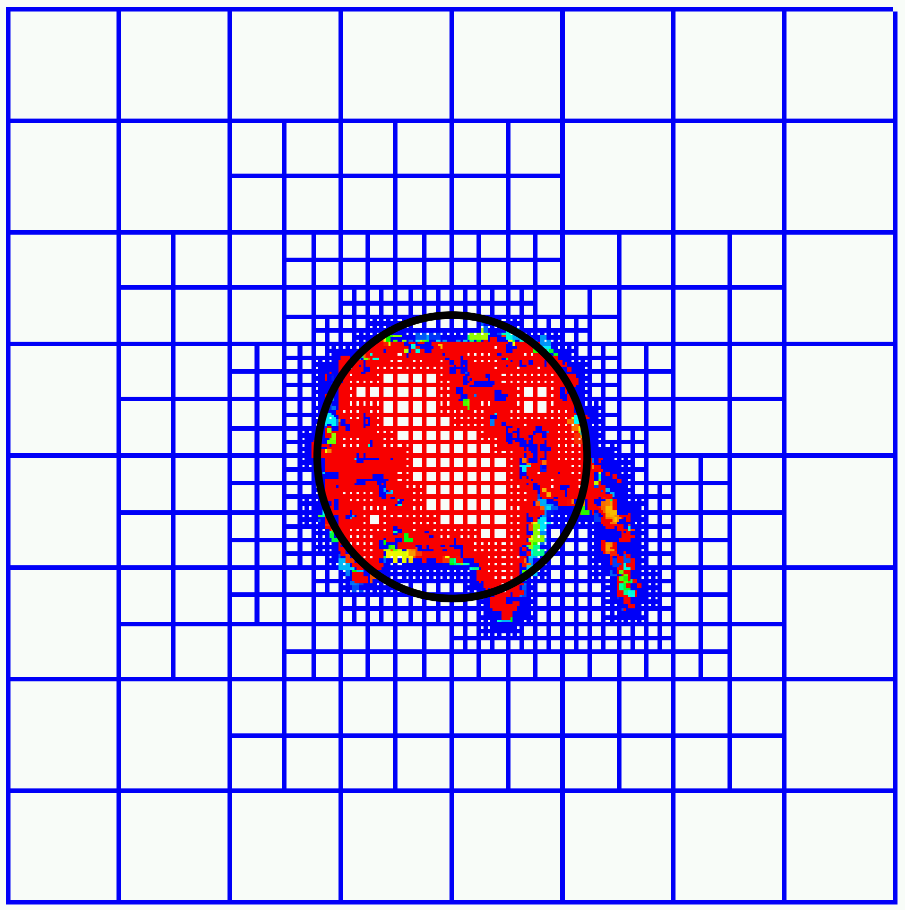
After 30 Newton steps, 6016 parameters

Transmission tomography example

Example of a 2D reconstruction using sound transmission



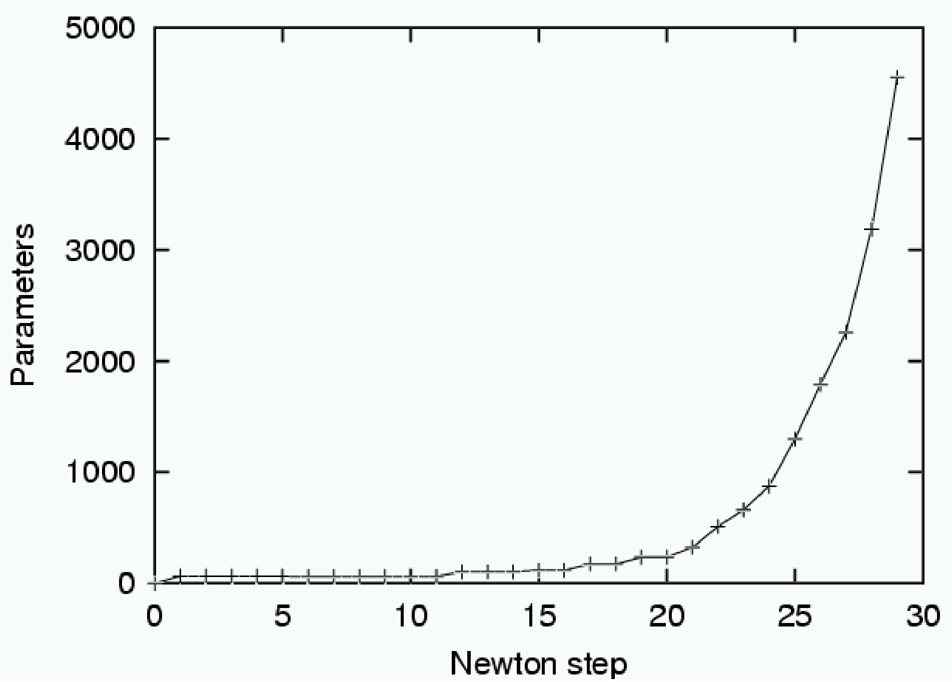
Grid for state/adjoint variables



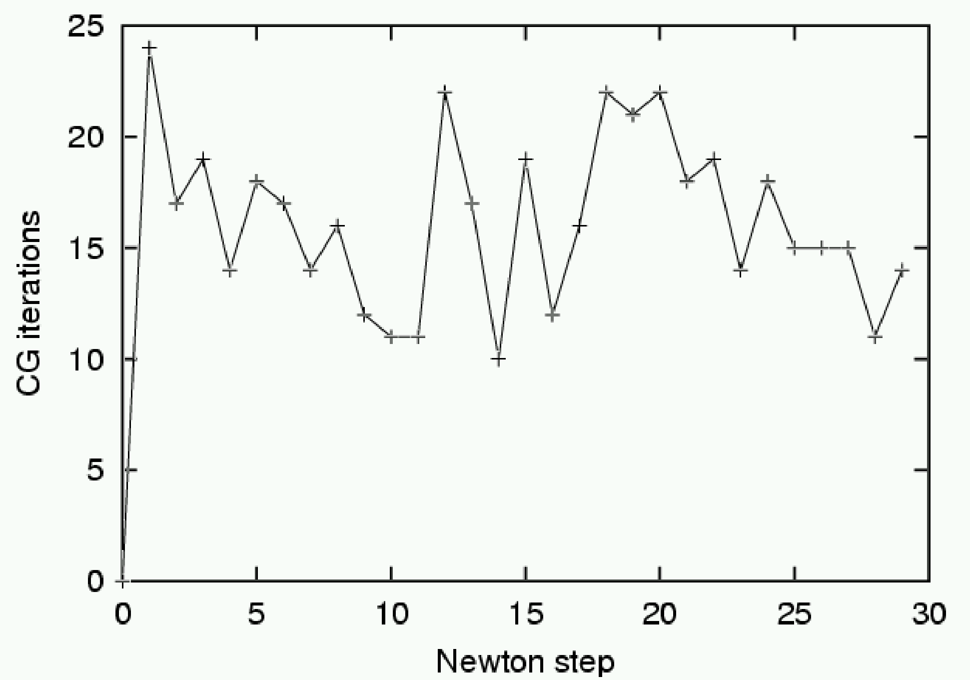
Grid for coefficient
(has 6016 cells, while a uniformly refined mesh of the same resolution would have 65k cells)

Transmission tomography example

Numbers of parameters



CG iterations per Newton step

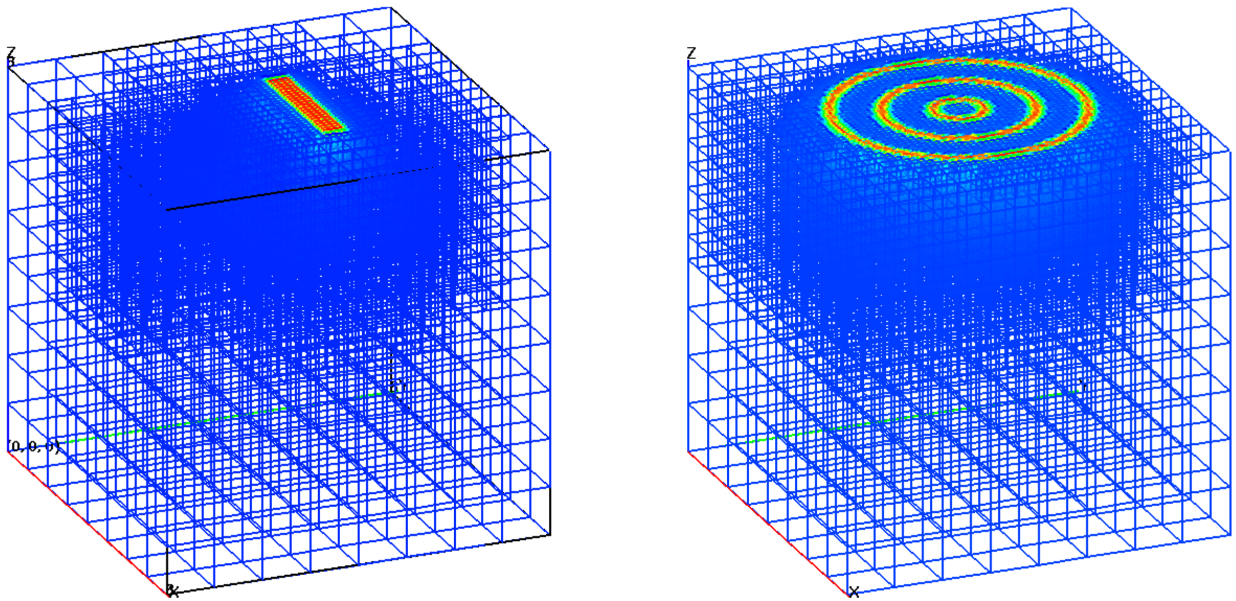


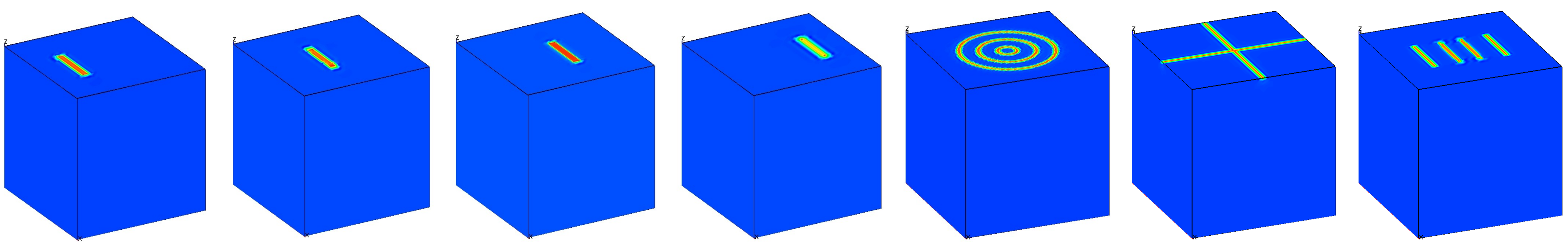
- Number of CG iterations *does not grow* with number of parameters!
- Per experiment approx. 1,000 solves
  total of 30*16*32*2 = 30,000 solves of the underlying PDE
- However, initial steps relatively cheap since systems small
- Yet, at end high resolution with >6,000 parameters

Optical tomography examples

Adaptive meshes for experimental cube:

- Illuminate a cube target with a number of different patterns
- Three targets at different depths

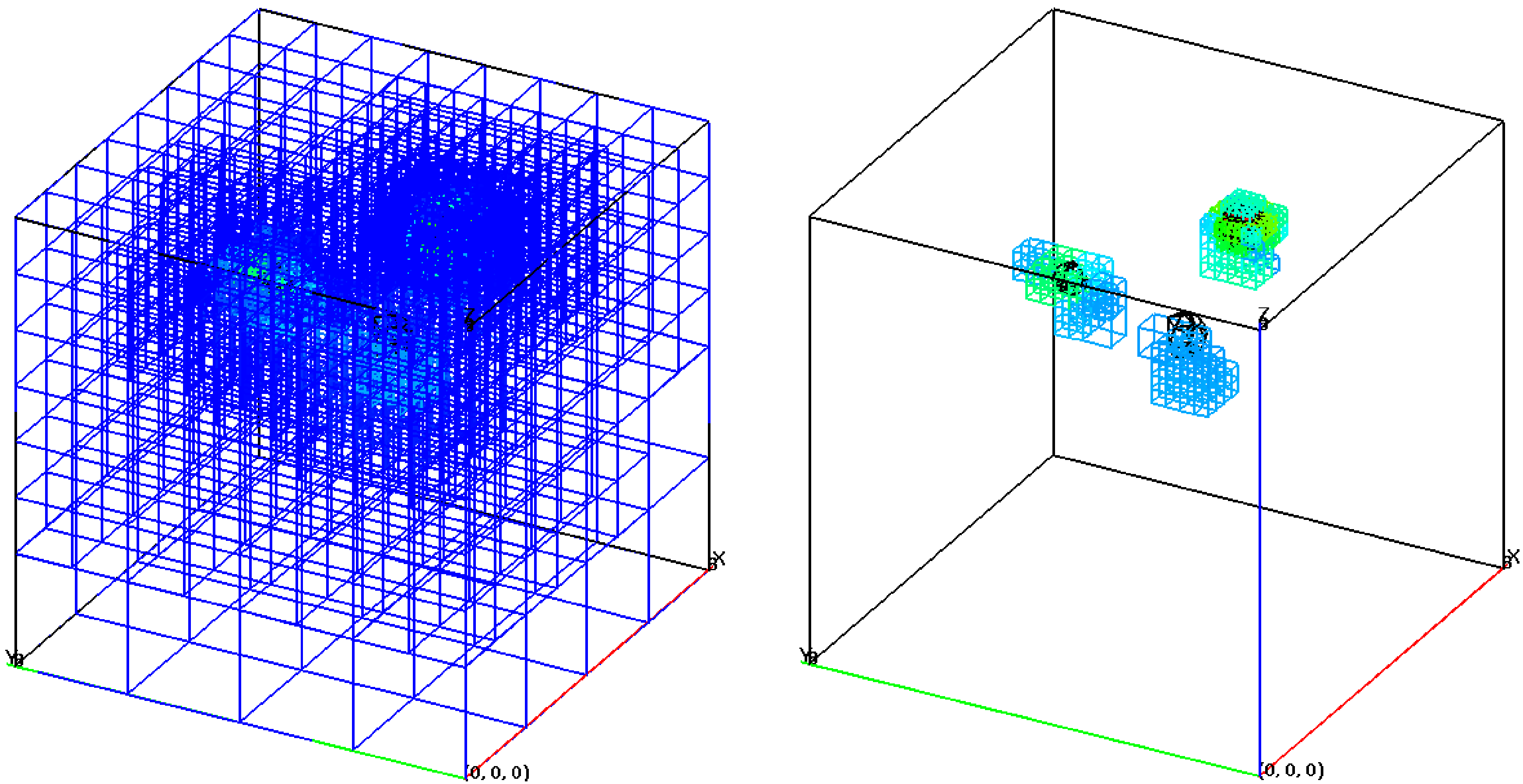




Optical tomography examples

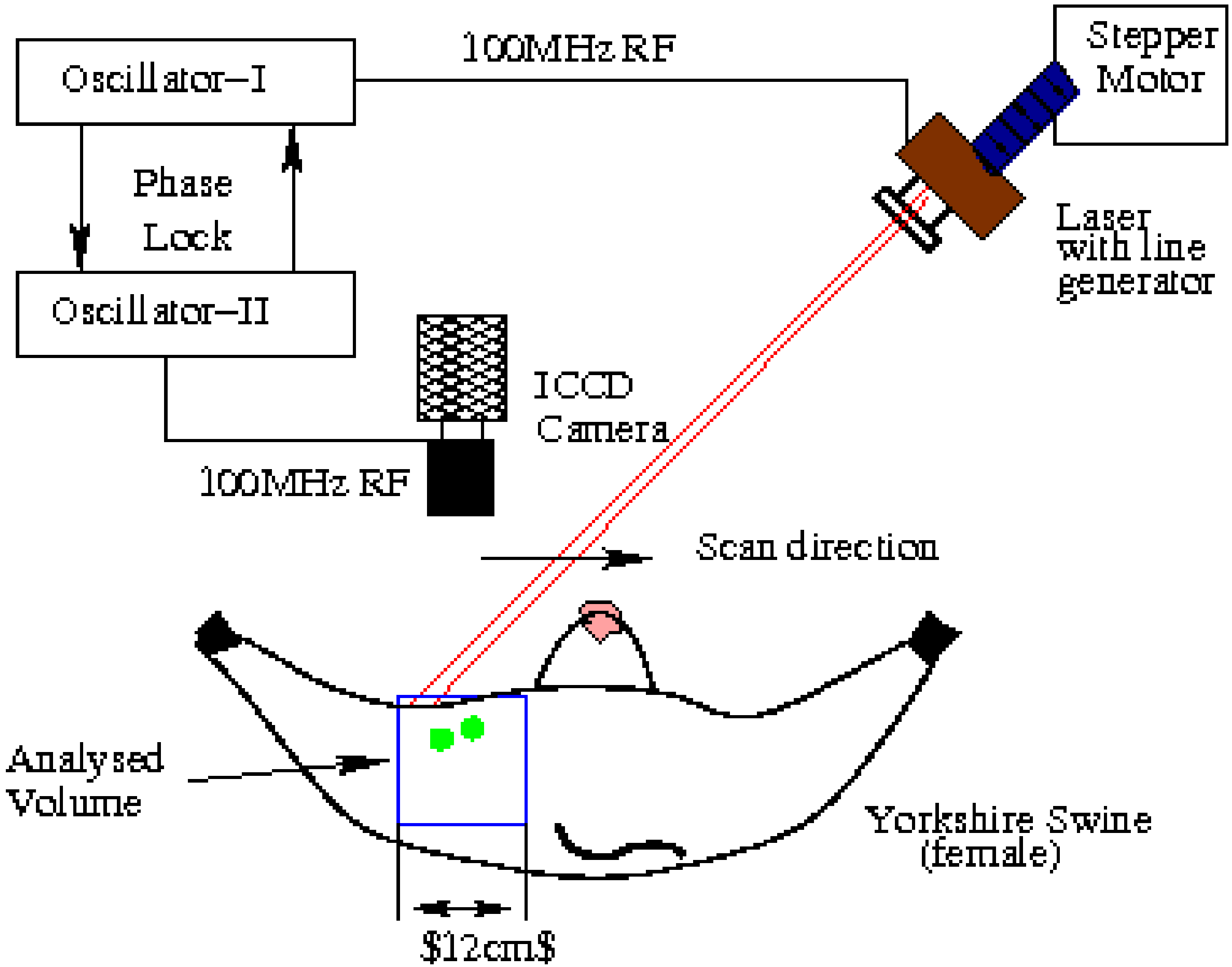
Adaptive meshes for experimental cube:

- Illuminate a cube target with a number of different patterns
- Three targets at different depths



Scanning the pig

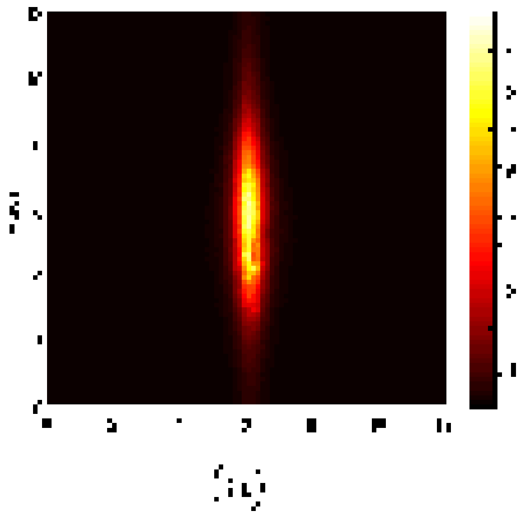
Experimental data obtained from a pig:



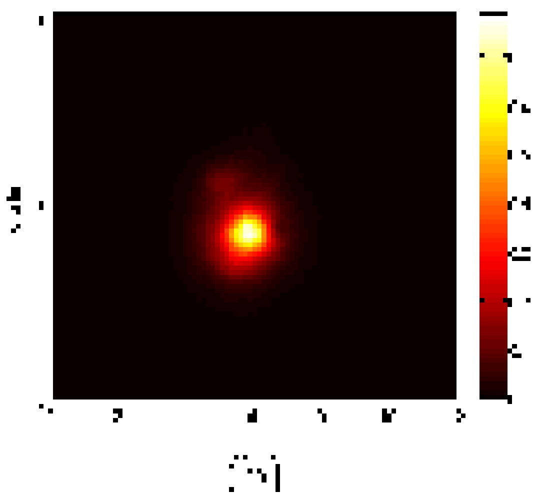
Experimental setup, a widened laser line scans the pig at 6 locations

Scanning the pig

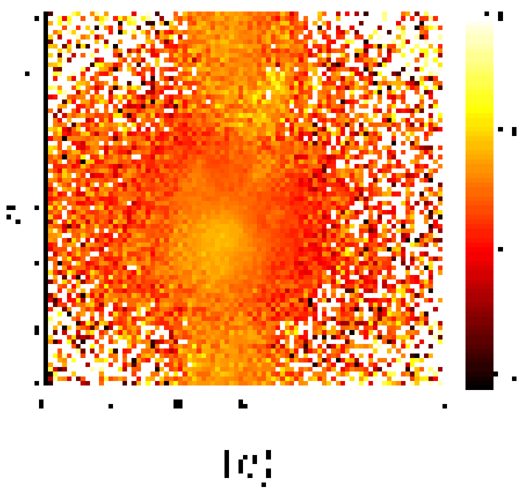
Experimental data obtained from a pig:



Illumination pattern

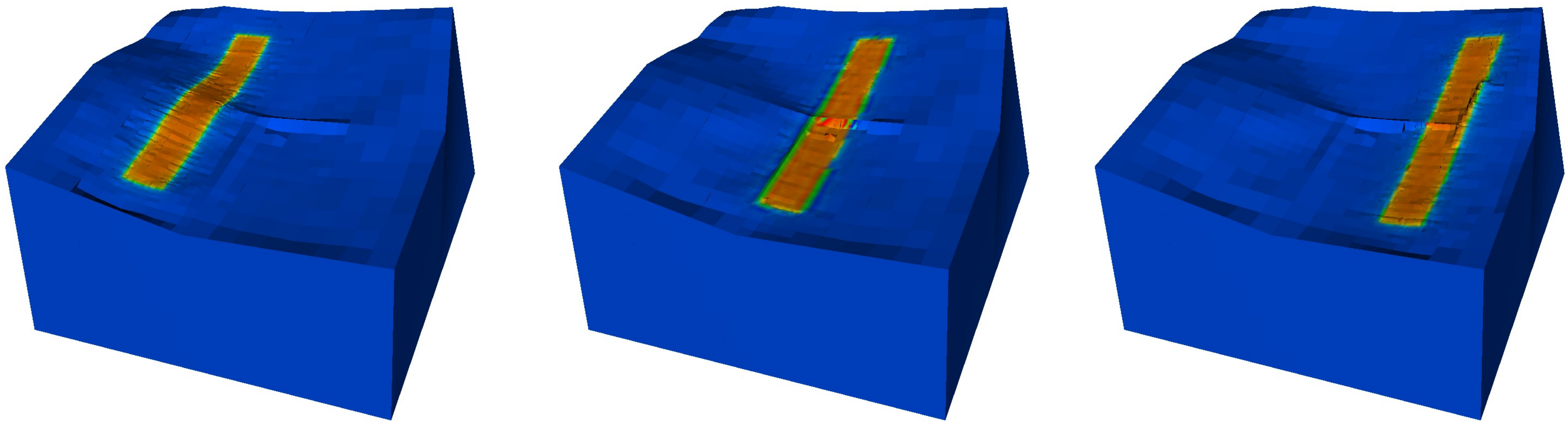


Fluorescent signal (amplitude)

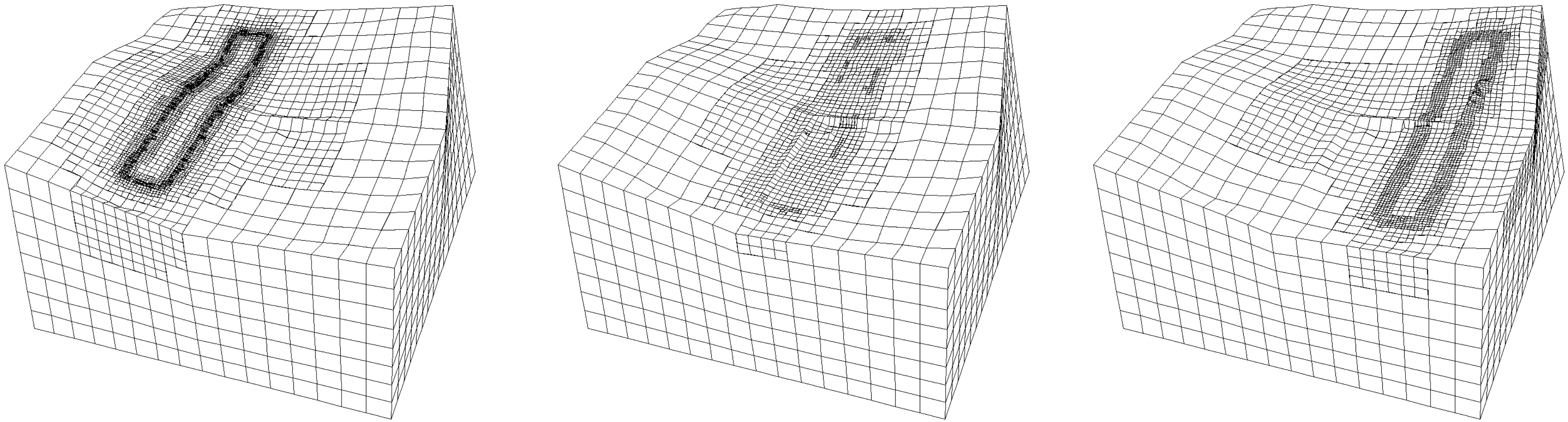


Fluorescent signal (complex phase)

Scanning the pig

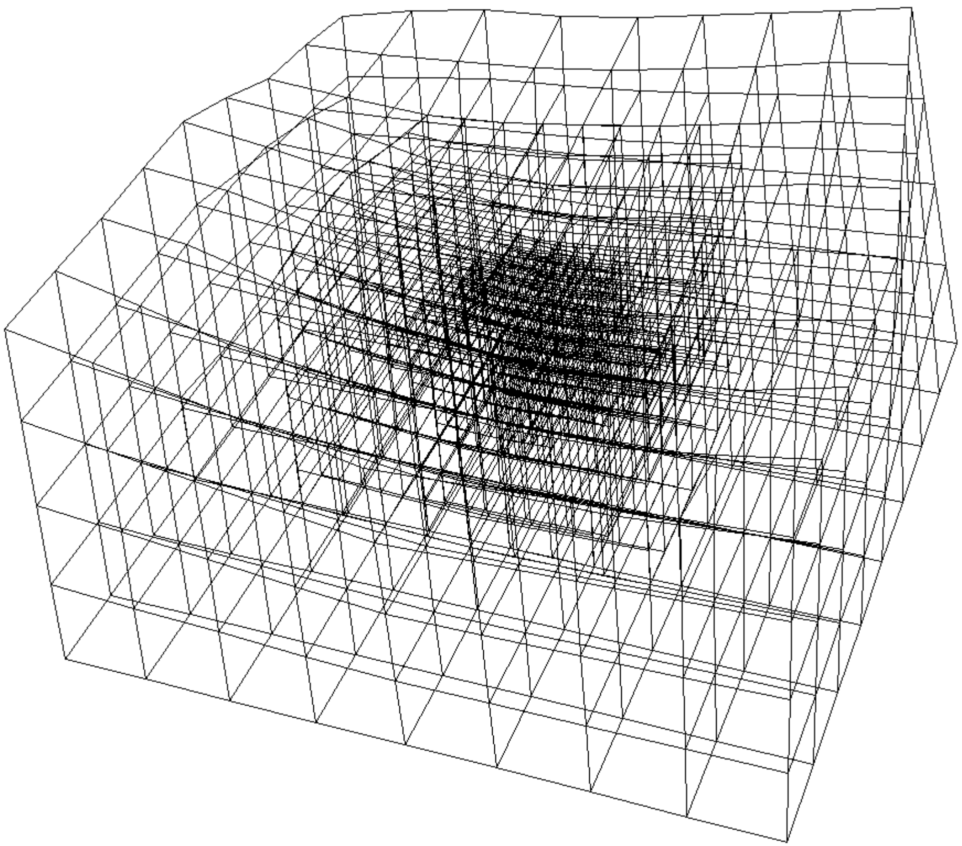


Solutions for various illumination positions

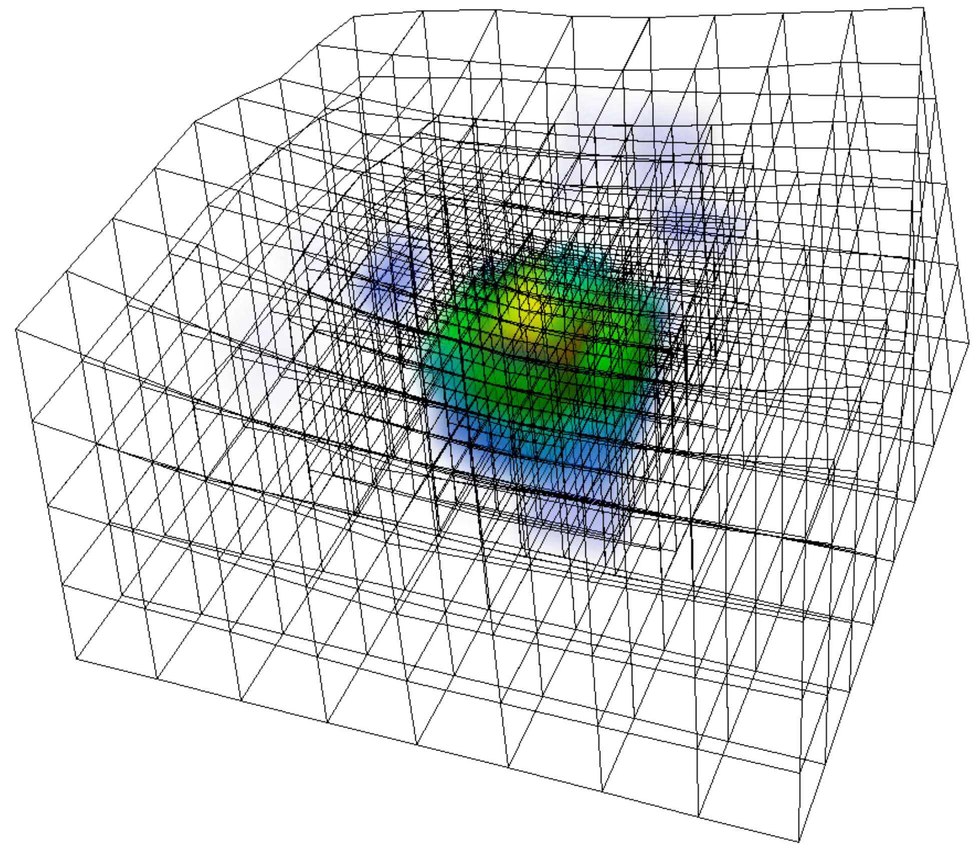


Meshes used for state and adjoint variables

Scanning the pig



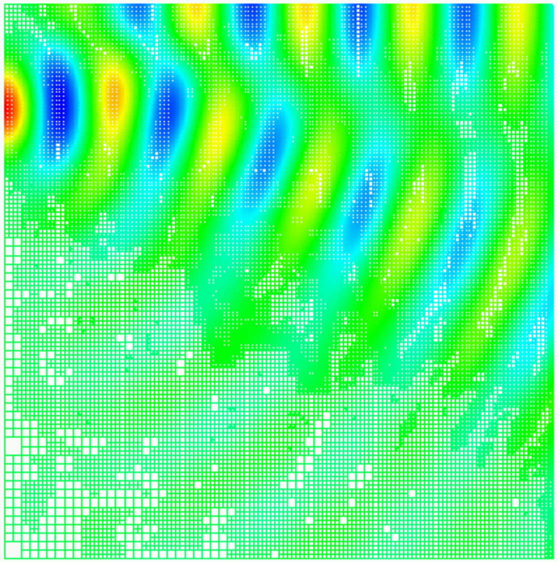
Mesh used to discretize the unknown parameter

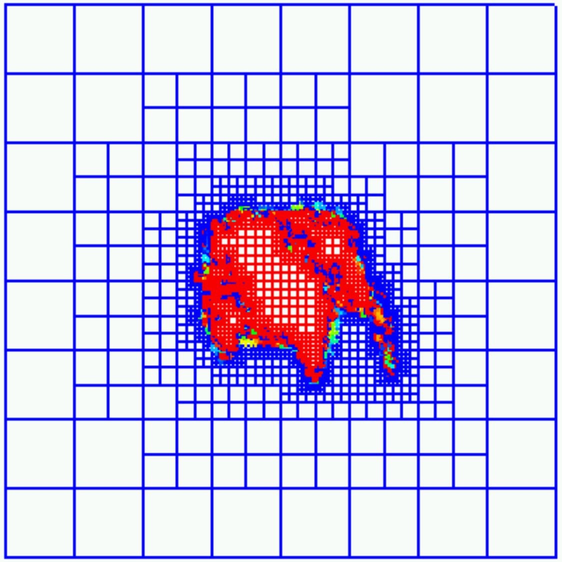


Reconstructed parameter (here: position of lymph node, later confirmed by a surgeon)

Results after 26 Newton iterations and approximately 10 minutes (~15,000 solutions of the underlying 3d PDE).

Summary



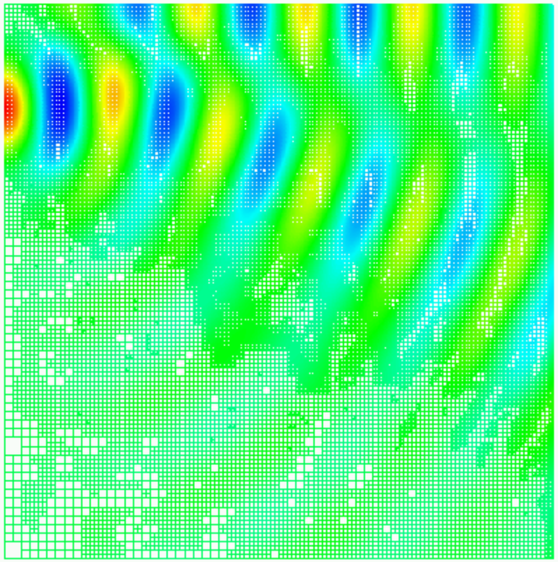


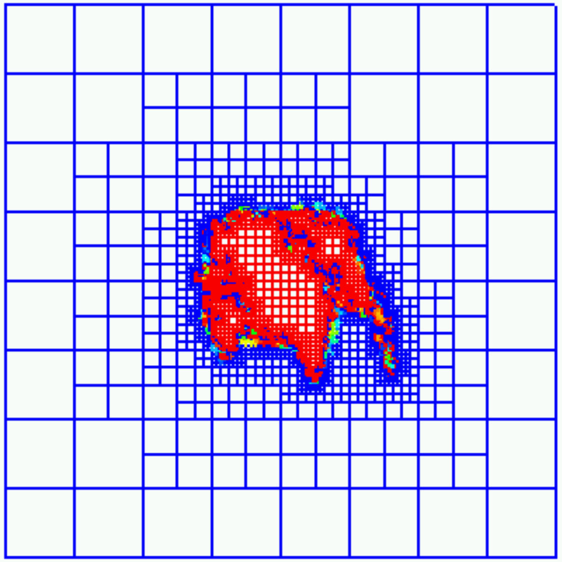
- Inverse problems are challenging numerically because the solution of a PDE is only a subproblem
- Because they are so eminently important in practice, we need (and have) efficient algorithms for
  - nonlinear and linear solvers
  - bound constraints
  - adaptive meshing techniques
- Using these methods, we can solve problems of a complexity that were intractable before
- For example, the resolution we achieve in optical tomography is one order of magnitude better than previously available in this field and much faster

Outlook and Open Questions

Mathematical questions on experiments:

- What are experimental setups that maximize the available information content?
- How can we quantify information content?
- How can we maximize it?





This leads into *experimental design* where we want to optimize the outcome of our inverse problem:

Optimize the result of an (already very expensive) optimization problem!

(Requires ~10^9 PDE solutions)

Collaborators

None of this would have been possible without:

- Amit Joshi, Eva Sevick-Muraca (Baylor College of Medicine, Houston)
- K. Hwang, J. Rasmussen, R. Sharma, W. Wang, ... (Baylor)
- G. Kanschat (Texas A&M), R. Hartmann (German DLR), O. Kayser-Herold (Harvard), ...
- G. Carey, M. Anderson (UTexas)

- A postdoc fellowship, support from NSF, NIH, DoE