Optical tomography: seeing inside the body

Kenneth M. Hanson

Los Alamos National Laboratory

This presentation available under http://home.lanl.gov/kmh/

Overview of presentation

- General problem of inversion of complex simulations
- Introduction to optical tomography
- Modeling of propagation of IR photons in tissue as diffusion process
- Simulation of diffusion process by finite-difference method - *"the forward problem"*
- Reconstruction of optical properties using adjoint differentiation - *"the inverse problem"*
- Examples of IR tomographic reconstructions
- Other applications of general technique

Acknowledgements

- Optical tomography
  - Suhail Saquib, Greg Cunningham
  - Andreas Hielscher, Alexander Klose, David Catarious
- Bayes Inference Engine
  - Greg Cunningham
  - Xavier Battle, Bob McKee
- Applications to hydrodynamics
  - Rudy Henninger, Maria Rightley
- General discussions
  - C. Thacker, J. Skilling, S. Gull, R. Silver, J. Gee
  - R. Giering, P. Maudlin, L. Margolin, B. Travis

Inversion of complex simulations

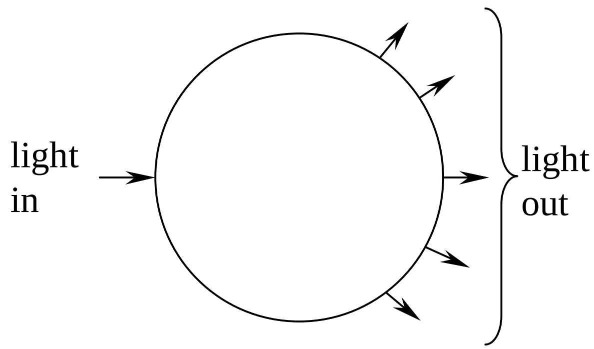
- There are BIG problems that
  - require complex numerical simulations to describe phenomena
  - are nonlinear in nature
  - one would like to fit to data, that is, solve the inverse problem
- Approximations are typically made in forward simulation to facilitate the solution of the inverse problem
  - perturbation methods (Born approximation)
  - truncated basis-function expansion
  - linearization of the problem
  - degradation of the spatial resolution
- Advanced methods are needed to invert large numerical simulations

Inversion of complex simulations

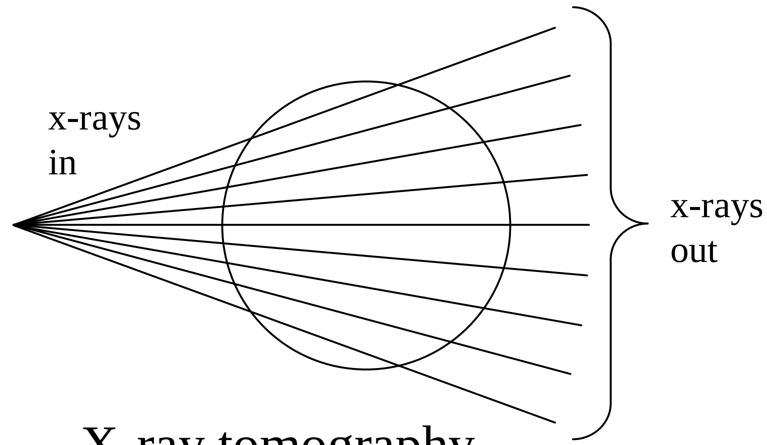
- Advanced techniques are required to cope with large data structures and numerous parameters
  - Optimization
    - gradient-based quasi-Newton methods (e.g., CG, BFGS)
    - adjoint differentiation for efficient calculation of gradients
    - multiscale methods for controlling optimization process
  - Bayesian methods
    - overcome ill posedness of inversion through use of prior knowledge
    - Markov chain Monte Carlo to characterize uncertainties
  - Appropriate higher-order models
    - Markov random fields
    - deformable geometrical models
    - but also consider lowest order, elemental representations

Optical tomography - general idea

- Shine light on tissue sample; measure light out



Optical tomography



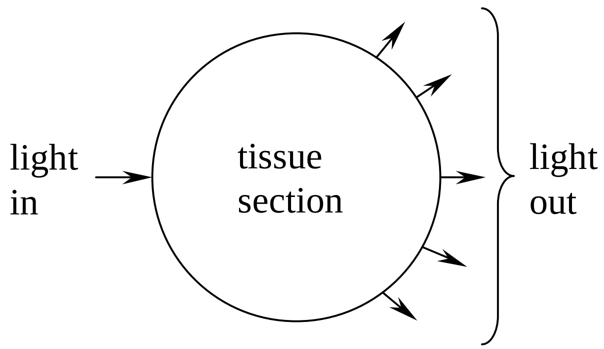
X-ray tomography

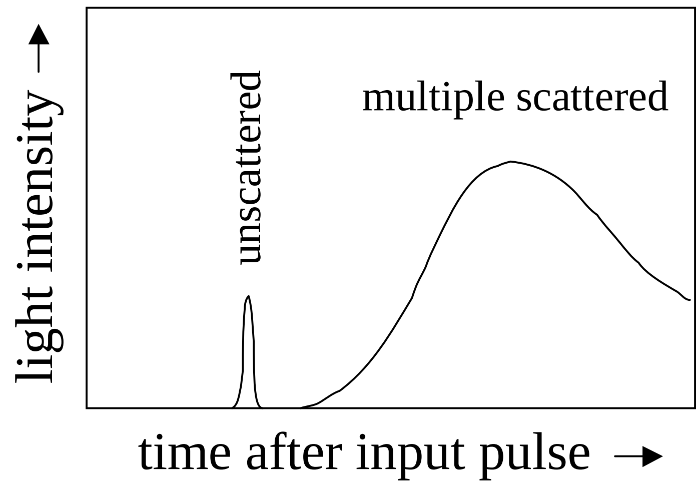
- Similar to x-ray computed tomography, so:
  - Can one actually do optical tomography?
  - What are best operating conditions?
  - What are imaging properties and diagnostic uses?

Physics of propagation of light in tissue

- Basic processes are scattering and absorption of photons
  - absorption in tissue is minimal in *infrared* range
  - IR photons can actually pass through bone
  - for soft tissue, $\mu_{\mathrm{scat}} \approx$ 1-10 cm^{-1}, $\mu_{abs} \approx$ 0.1 cm^{-1}
- Transport equation generally applies
- Diffusion equation often good approximation
  - valid when scattering is isotropic and without energy loss

Proposed experimental scheme



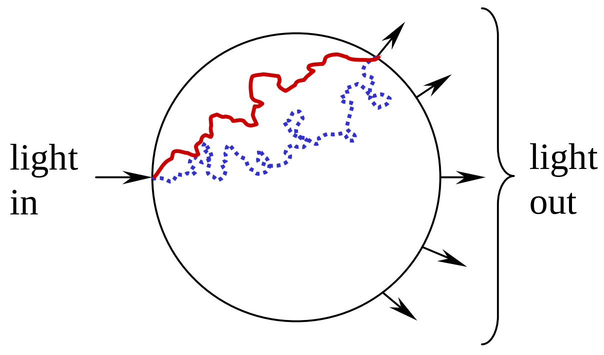


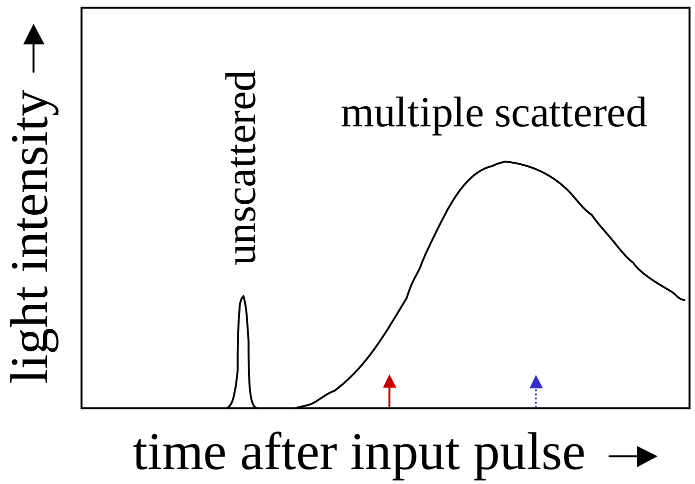
- Shine IR light pulse on tissue sample at several positions
- For each input pulse, measure at several output positions the light intensity vs. time with time resolution << 1 ns.
  - prompt, unscattered photons; few survive thick sections
  - multiply scattered photons; meander or diffuse through section

Alternative experimental schemes

- Numerous types of measurements of the light transmitted through tissue sample are possible:
  - pulsed input; measure full time dependence distribution (delta-function response)
  - pulsed input; measure average time <t> (first moment of time distribution)
  - modulated input; measure amplitude and phase of modulated output intensity (Fourier transform of delta-function response)
  - constant input; measure amplitude of output (integral of delta-function response)

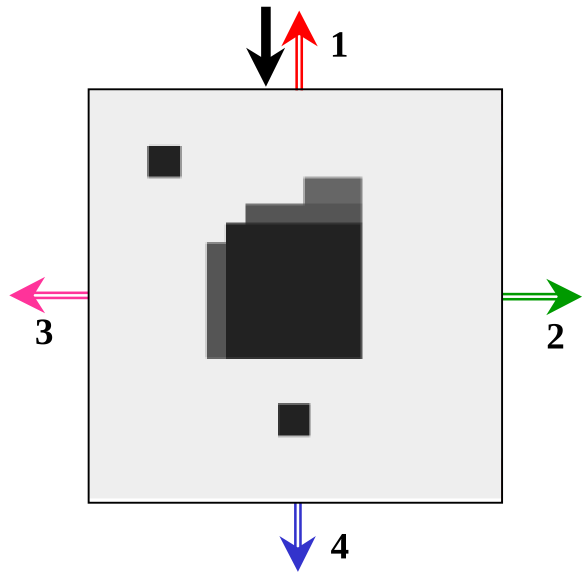
Modeling of process



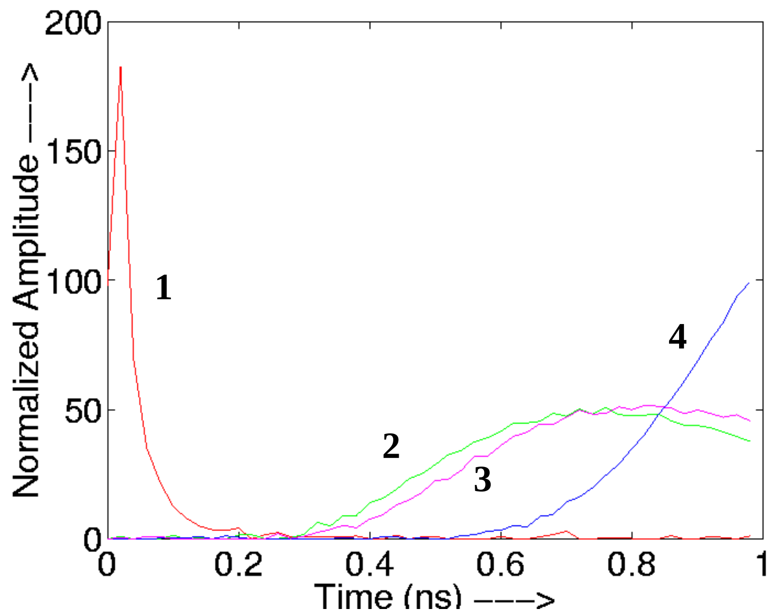


- IR light photons in broad, retarded peak literally "diffuse" by multiple scattering from source to detector
  – time is equivalent to distance traveled
  – diffusion equation models these multiply-scattered photons
  – these photons do not follow straight lines

Example - simulation of light diffusion



$0.7 < D < 1.4$ cm^{2}ns^{-1} (μ_a=0.1 cm^{-1})



– for assumed distribution of diffusion coefficients (left)
– predict time-dependent output at four locations (right)
– reconstruction problem - determine image on left from data on right

Reconstruction problem

- Determine tissue properties from measurements
  - diffusion coefficient $D(x,y)$ and absorption coefficient $\mu_a(x,y)$, as a function of position - therefore, many unknowns
- Many problems must be overcome
  - photon paths depend on properties to be reconstructed
  - hence, inverse problem is nonlinear and difficult
  - measurements can only be calculated numerically;
    no analytic expression for measurements in terms of D and μ_a
  - gradients are desired for speedy gradient-based optimization
    - needed with respect to (wrt) many unknowns
    - analytic gradients are not available
    - numerical gradients by perturbation would be time consuming

Diffusion equation

- Infrared light diffuses through tissue and bone
- Partial differential equation describes diffusion process
  - *U(x,y,t)* is intensity of diffused light (no angular dependence)

$$\frac{\partial U}{\partial t} = \frac{\partial}{\partial x}\left[D \frac{\partial U}{\partial x} \right] + \frac{\partial}{\partial y}\left[D \frac{\partial U}{\partial y} \right] - c\mu_{abs} U + S$$

  - where *D(x,y)* is position-dependent diffusion coefficient,
    $\mu_{abs}(x,y)$ is the linear absorption coefficient,
    c is the speed of light,
    S(x,y,t) is a source term;

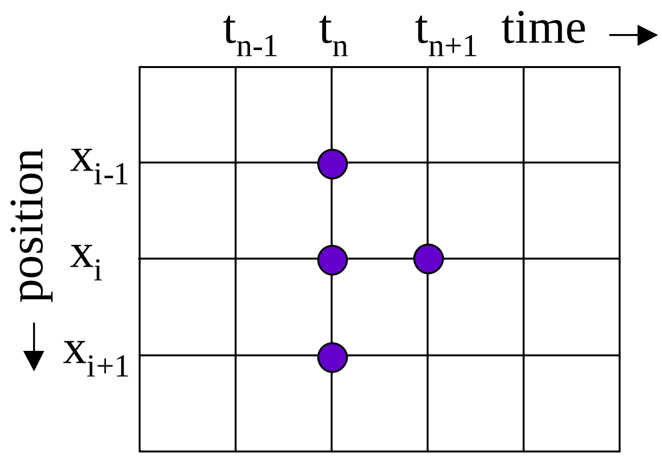
$D = c[3(\mu_{abs} + \mu'_{scat})]^{-1}$ (μ'_{scat} = effective scattering coefficient)

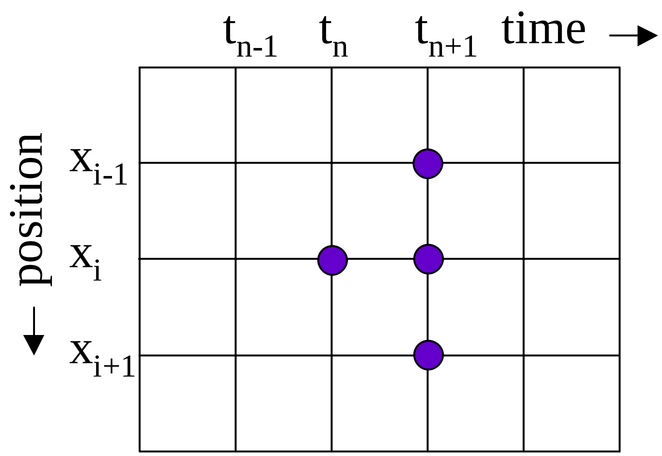
Method of finite differences

- Approximate derivatives by finite differences
  - wrt time: $$\frac{\partial U}{\partial t} \Rightarrow \frac{\Delta U}{\Delta t} = \frac{U_{i,n+1} - U_{i,n}}{\Delta t}$$
  - wrt position: $$\frac{\partial^2 U}{\partial x^2} \Rightarrow \frac{U_{i+1,n} - 2U_{i,n} + U_{i-1,n}}{(\Delta x)^2}$$
- Differential equation then becomes a set of linear equations to be solved to obtain time-step update
  - calculate time evolution, starting with initial conditions
- Question: at what time should second derivative wrt position be calculated, *n* or *n+1*?

Calculation of finite differences

- For diffusion equation, need
  - temporal first order derivative
  - spatial second order derivative
- Explicit technique
  - for step from time *n* to *n+1*, evaluate spatial derivative at *n*
  - **unstable** for moderate time steps
- Implicit technique
  - for step from time *n* to *n+1*, evaluate spatial derivative at *n+1*
  - inherently **stable**





Explicit method

- Evaluating position derivatives at *n*

$$\frac{U_{i,n+1} - U_{i,n}}{\Delta t} = D_i \frac{U_{i+1,n} - 2U_{i,n} + U_{i-1,n}}{(\Delta x)^2} - c\mu_i U_{i,n} + S_{i,n}$$

  - for clarity, ignore position dependence of *D* and *y* coord.
  - yields set of linear equations:

$$\mathbf{U}_{n+1} = \mathbf{B}\mathbf{U}_n + b\mathbf{S}_n \quad (b \text{ is a scalar constant})$$

- $\mathbf{U}_{n+1}$ at new time $n+1$ is given **explicitly** in terms of state at previous $\mathbf{U}_n$
- Easy to calculate time steps (just matrix multiplication)
- Unfortunately, inherently **unstable** for moderate Δt

Implicit method

- Evaluating position derivatives at time *n+1*

$$\frac{U_{i,n+1} - U_{i,n}}{\Delta t} = D_i \frac{U_{i+1,n+1} - 2U_{i,n+1} + U_{i-1,n+1}}{(\Delta x)^2} - c\mu_i U_{i,n+1} + \tfrac{1}{2}(S_{i,n+1} + S_{i,n})$$

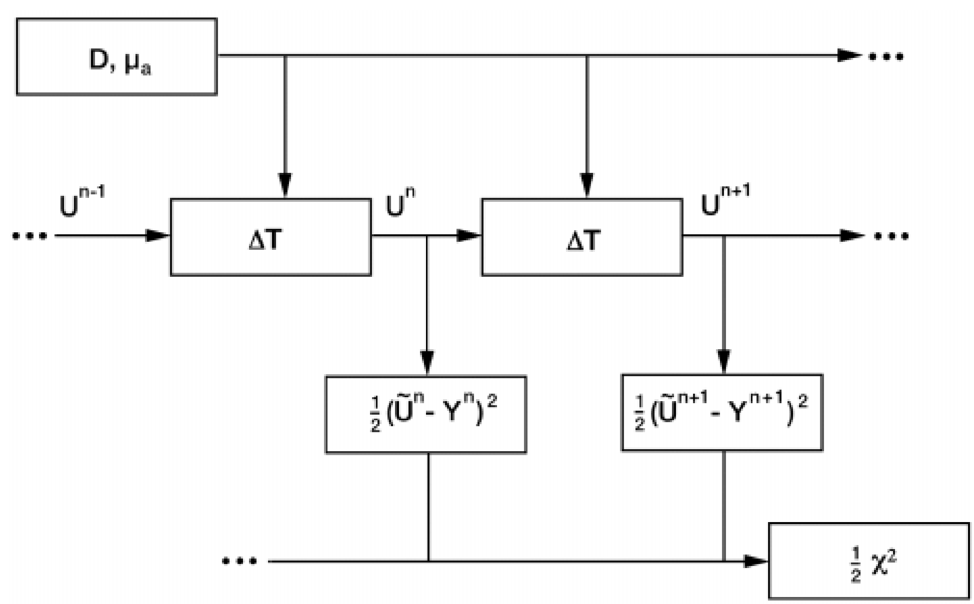
  - for clarity, ignore position dependence of *D* and *y* coord.
  - yields set of linear equations:

$$\mathbf{AU}_{n+1} = \mathbf{U}_n + a\,\mathbf{S}_n \quad (a \text{ is a scalar constant})$$

- $\mathbf{U}_{n+1}$ at new time *n+1* is given **implicitly** in terms of state at previous time $\mathbf{U}_n$
- Must solve set of linear eqs. to calculate time steps
- Inherently **stable** for moderate Δt

Finite-difference calculation

- Data-flow diagram shows calculation of time-dependent measurements by finite-difference simulation
- Calculation marches through time steps Δt
  - new state $\mathbf{U}_{n+1}$ depends only on previous state $\mathbf{U}_n$



Inversion of forward calculation

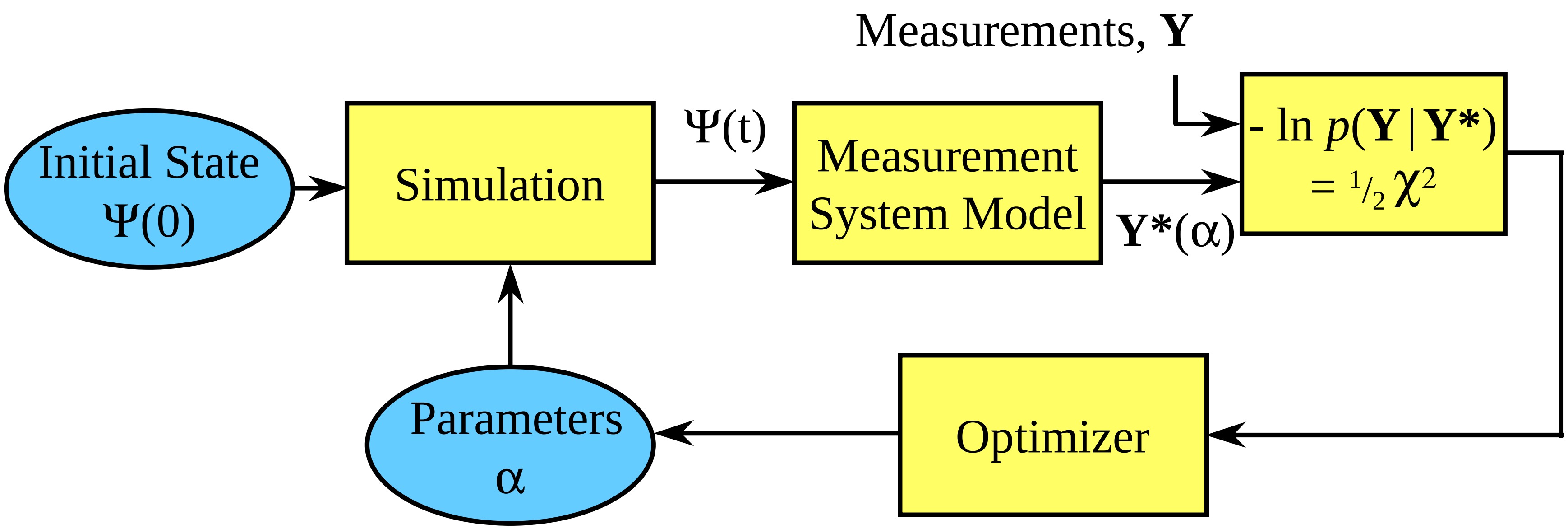
- To find parameters $\alpha = (D, \mu_a)$, minimize minus-log-likelihood of data:

$$\phi(\alpha) = -\ln p(\mathbf{Y} \mid \alpha) = \tfrac{1}{2} \sum_m \frac{(Y_m - Y_m^*)^2}{\sigma_m^2} = \tfrac{1}{2} \chi^2$$

  – where Y_m is the mth measurement,
    Y_m^* its predicted value (= $U_{s,n}$ at appropriate s and n),
    σ_m is rms noise in measurement
  – measurements are at fixed position, but at all times

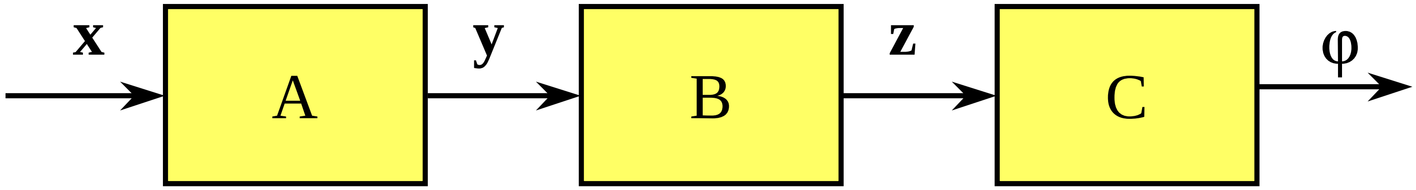
- Problems for inverting diffusion process
  – inversion may be ill posed, a theoretical issue
  – have only numerical solution of forward simulation, so calculation of gradient poses practical problem

Parameter estimation by fitting data



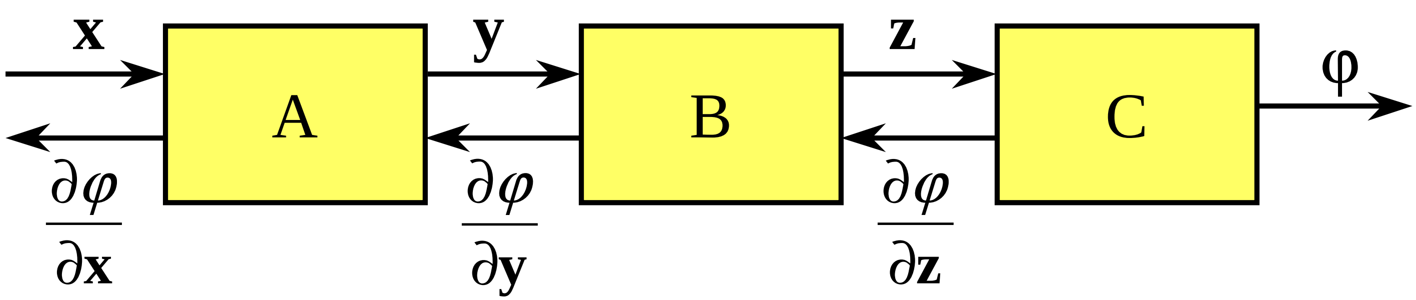
- Diagram describes general approach (analytical and computational)
- Find parameters (vector α) that minimize -ln $p(\mathbf{Y}|\ \mathbf{Y}^*(\alpha))$
- Result is maximum likelihood estimate for α
  - also known as minimum-chi-squared or least-squares solution
- Optimization process is accelerated by using gradient-based algorithms; therefore need gradients of simulation and measurement processes

Differentiation of sequence of transformations



- Data-flow diagram shows sequence of transformations A->B->C that converts data structure **x** to **y** to **z** and then scalar φ
- Desire derivatives of φ wrt all components of **x**, *assuming* φ is *differentiable*
- Chain rule applies: $$\frac{\partial \varphi}{\partial x_i} = \sum_{j,k} \frac{\partial y_j}{\partial x_i} \frac{\partial z_k}{\partial y_j} \frac{\partial \varphi}{\partial z_k}$$
- Two choices for summation order; the one that reverses data flow is preferable, because it avoids large intermediate matrices of derivatives

Adjoint Differentiation In Code Technique ADICT

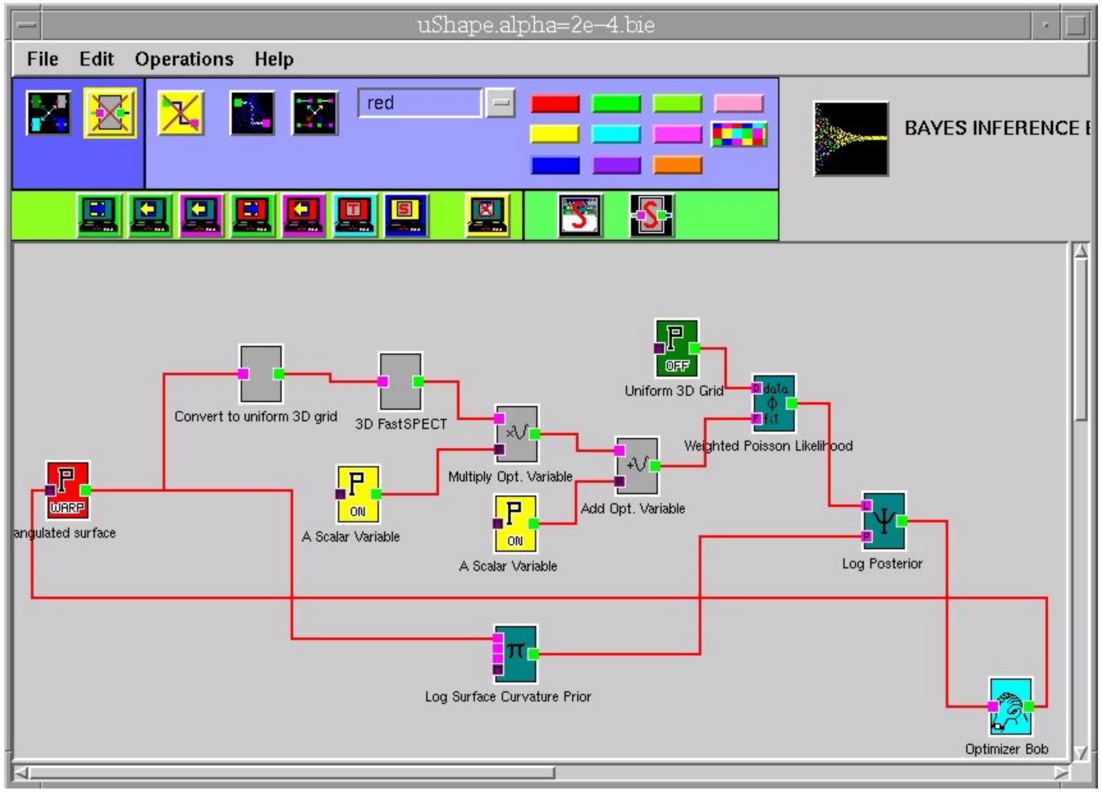


- For sequence of transformations that converts data structure **x** to scalar φ
- Derivatives $\frac{\partial \varphi}{\partial \mathbf{x}}$ are efficiently calculated in the reverse (adjoint) direction
- Code-based approach: logic of adjoint code is based explicitly on the forward code or on derivatives of the forward algorithm
- **Not** based on the theoretical eqs., which forward calc. only approximate
- Only assumption is that φ is a **differentiable function** of **x**
- CPU time to compute **all** derivatives is comparable to forward calculation

Adjoint Differentiation Crucial to BIE Success

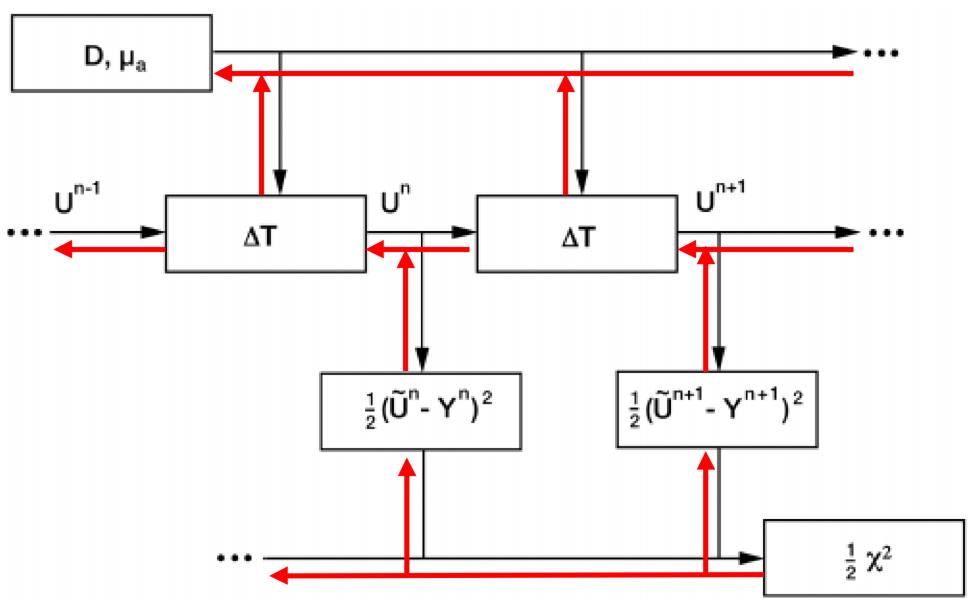
- Bayes Inference Engine (BIE) created in DX-3
  - modeling tool for interpreting radiographs

BIE programmed by creating data-flow diagram, shown here for a 3D reconstruction problem



Adjoint differentiation in diffusion calculation

- Adjoint differentiation calculation precisely reverses direction of forward calculation
- Each forward data structure has an associated derivative
  – where $\mathbf{U}_n$ propagates forward, $\dfrac{\partial\varphi}{\partial\mathbf{U}_n}$ goes backward $(\varphi = \frac{1}{2}\chi^2)$



Comments about diffusion problem

- Algorithm used to solve forward problem was chosen without regard to inversion process
  - adjoint differentiation typically places no requirement on simulation method
- Simplifying aspects of diffusion problem:
  - update operation depends only on parameters $D(x,y)$ and $\mu_{abs}(x,y)$ (time independent)
  - adjoint derivatives do not depend on state of system $\mathbf{U}_n$
  - no need to save $\mathbf{U}_n$ during forward calculation

Automatic differentiation

- Several tools exist for automatically differentiating codes (only available for FORTRAN77)
  - TAMC (R. Giering, JPL, prev. MIT & MPI-Meteorology)
    - operates in both forward and reverse directions
    - works for large codes; follows ADICT principle
  - GRESS (Hordewel, et al., ORNL)
    - operates in both forward and adjoint directions
    - can not compute gradients wrt many parameters for large calcs.
    - for adjoint, stores derivatives for each line of the forward code
  - ADIFOR (Bischof, Griewank, et al., ANL)
    - only operates in forward direction
    - can not compute gradients wrt many parameters

Bayesian approach to inversion

- Inverse problems are often ill posed, meaning there is no unique solution
- Bayesian formalism overcomes ill posedness by introducing prior information through Bayes law:

$$\ln p(\alpha \mid \mathbf{Y}) = \ln p(\mathbf{Y} \mid \alpha) + \ln p(\alpha) + C$$

  – where $p(\alpha|\mathbf{Y})$ = posterior probability of the parameters α,
    $p(\mathbf{Y}|\alpha)$ = likelihood of the data,
    $p(\alpha)$ = prior probability of the parameters α
- Bayesian posterior $p(\alpha|\mathbf{Y})$ describes uncertainty in inferred parameters

Prior based on Markov Random Field model

- MRF can control local behavior of an intensity field
- Minus-log-prior given by (considering only *D*)

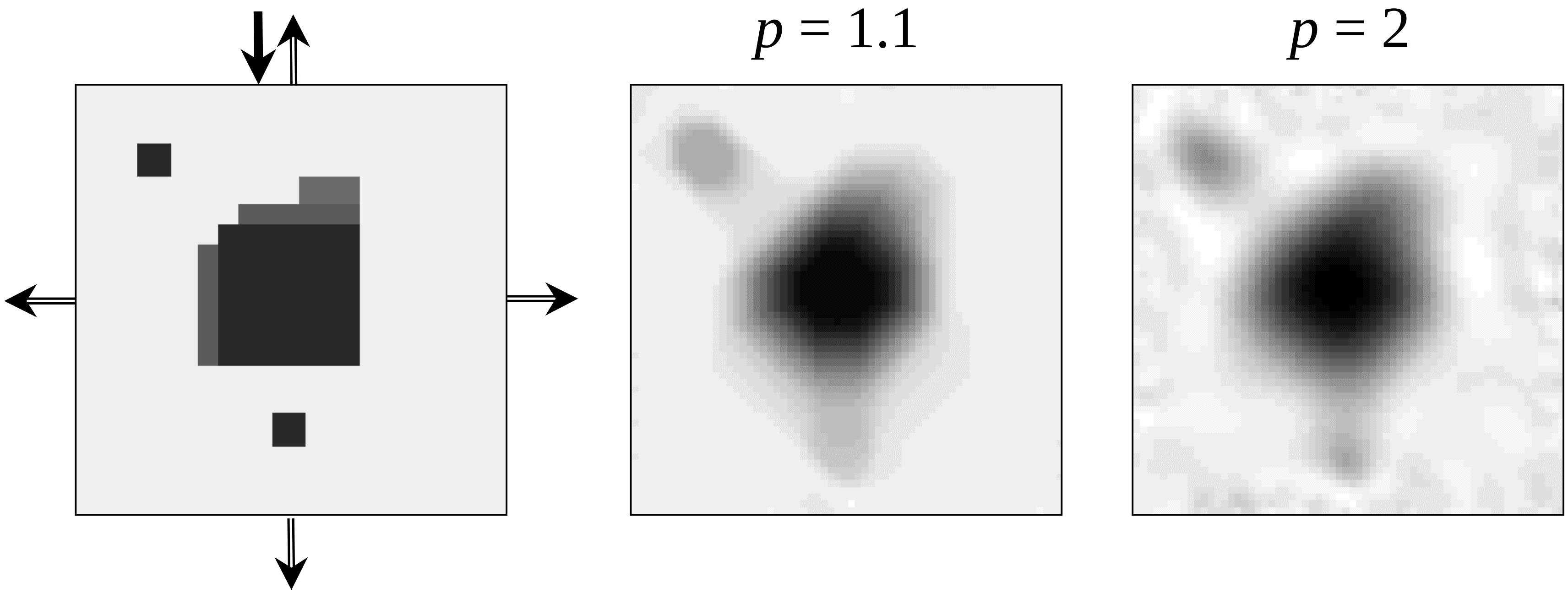
$$-\ln p(\mathbf{D}) = \beta \sum_i | D_i - \overline{D}_i |^p$$

  - where D_i = diffusion coefficient at *i*th pixel,
    $\overline{D}_i$ = *D* averaged over a neighborhood of *i*th pixel
  - this is added to minus-log likelihood ($\chi^2/2$)
- The exponent *p* controls shape of penalty function
  - $p = 2$ (standard) excessively penalizes large fluctuations
  - $p \cong 1$ results in better reconstructions
- Parameter β conveniently determined for MRF model

Examples

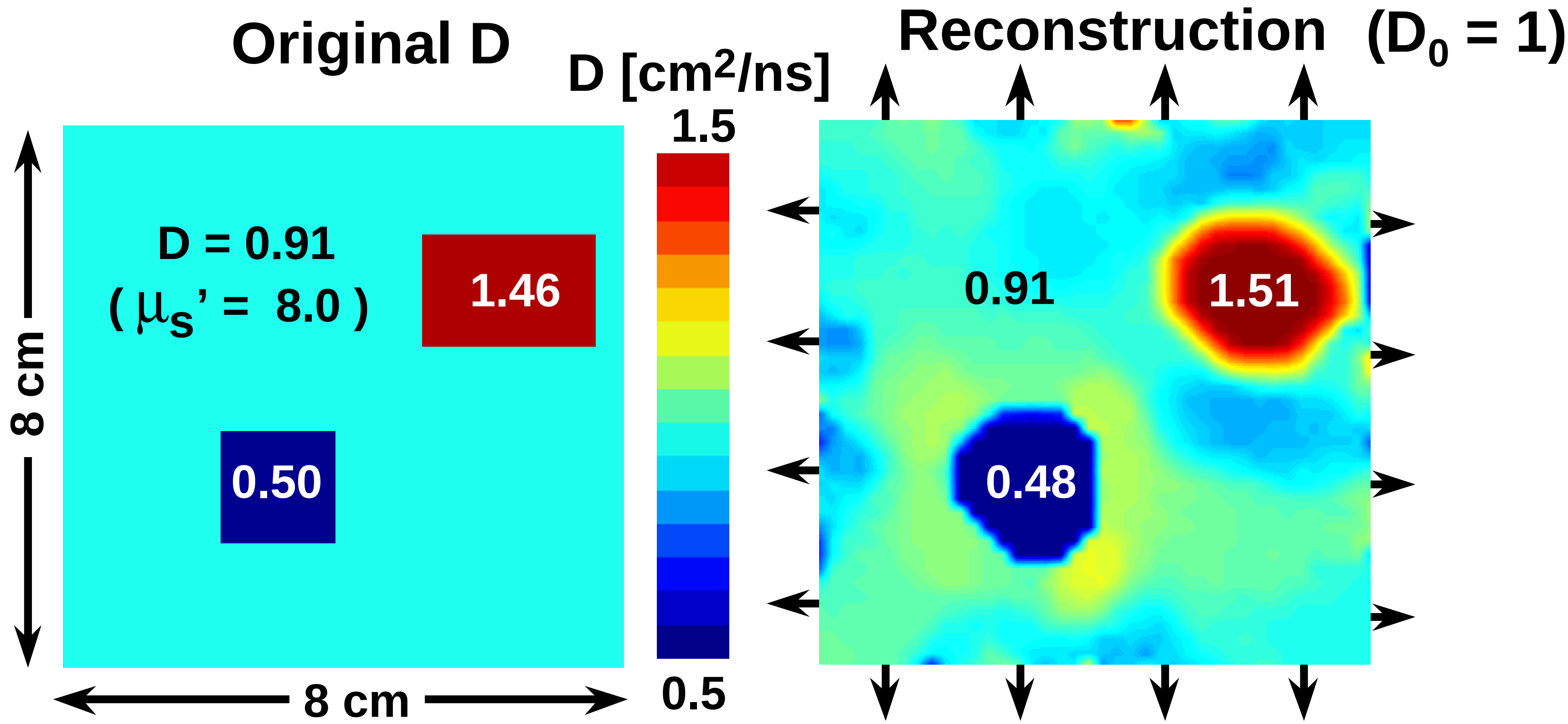
- Initial project - reconstruct $D(x,y)$ for simple phantom
  - Saquib, Hanson, Cunningham (LANL)
- Extension to simultaneously obtain $D(x,y)$ and $\mu_a(x,y)$; simulations relevant to human tissue
  - Hielscher, Klose, Catarious, Hanson (LANL)
- 3D reconstruction; applications to hypothetical diagnostic cases
  - brain, ventricular bleeding, and arthritis in finger joints
  - Hielscher, Klose (SUNY - Brooklyn), Hanson (LANL), Beuthan (FU Berlin)

Reconstruction of simple phantom



- Measurements
  - section is $(6.4\text{cm})^2$, $0.7 < D < 1.4\ \text{cm}^2\text{ns}^{-1}$ ($\mu_{abs} = 0.1\ \text{cm}^{-1}$)
  - 4 input pulse locations (middle of each side)
  - 4 detector locations; intensity measured every 50 ps for 1 ns
- Reconstructions on 64 x 64 grid from noisy data (rmsn = 3%)
  - conjugate-gradient optimization algorithm

Simultaneously determine D and μ_a

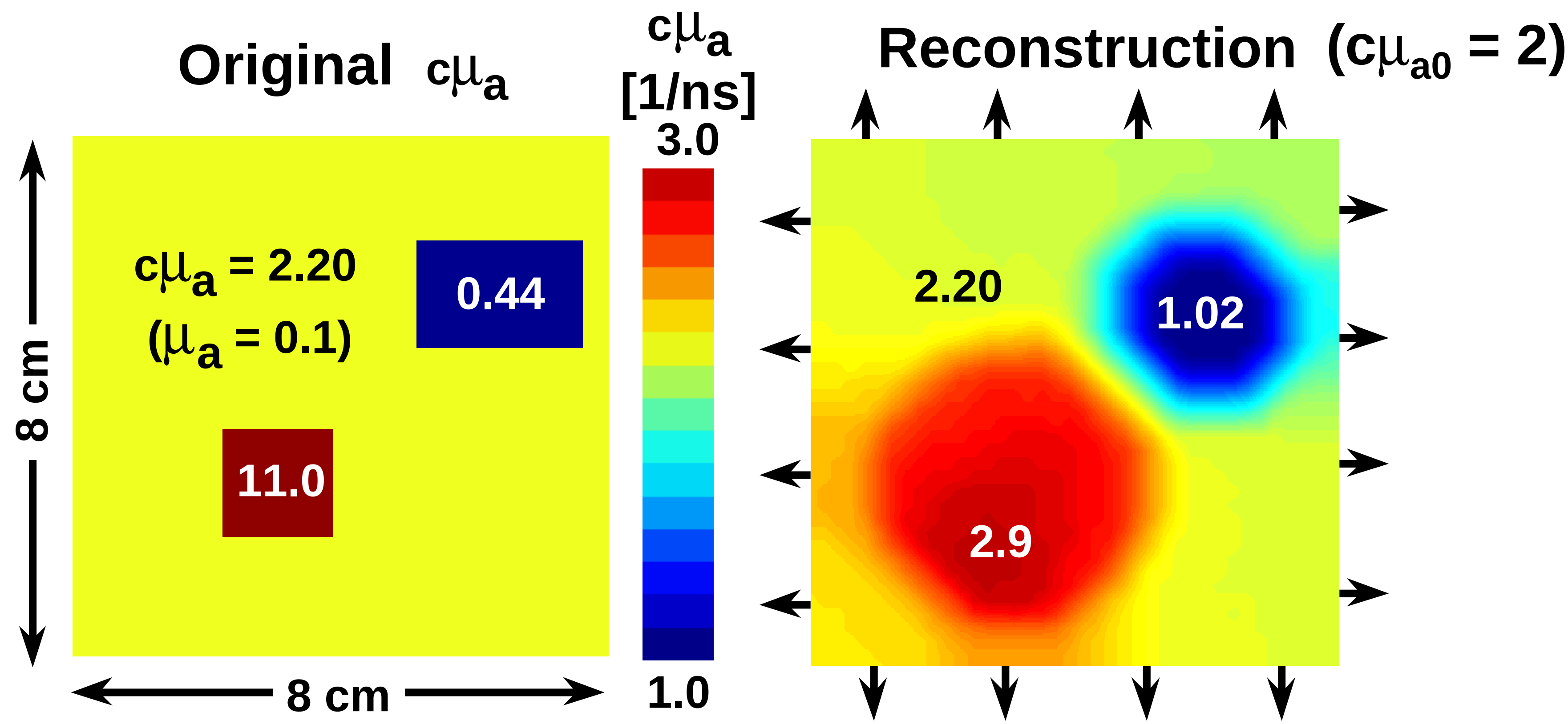


40x40 = 1600 pixels, 2 parameters
16 source and 15 detector positions
50 time steps

27 iterations ~ 8 min

derivatives by finite differences would take 1600x2x16x27 = 1,382,400 forward calculations

Simultaneously determine D and μ_a



40x40 = 1600 pixels, 2 parameters
16 source and 15 detector positions
50 time steps

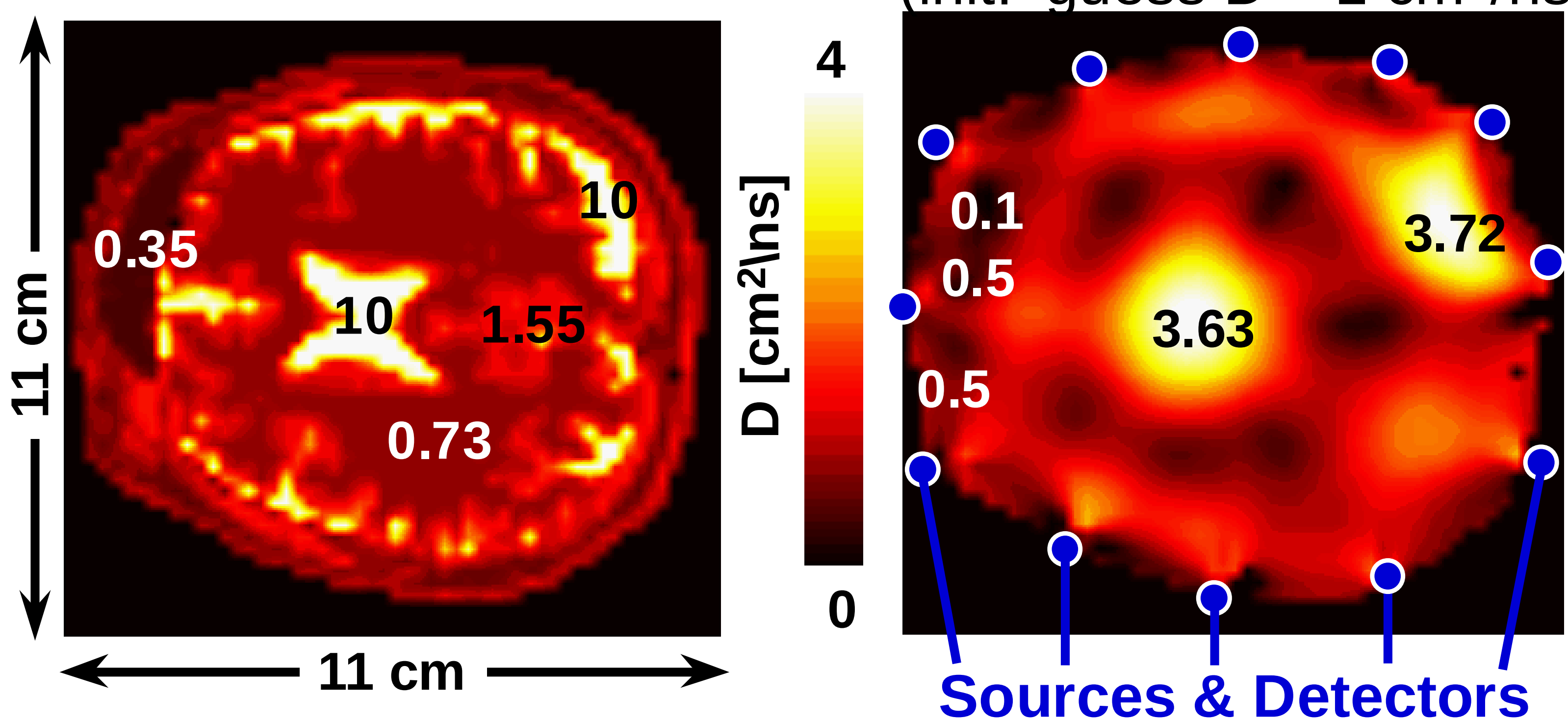
27 iterations ~ 8 min

derivatives by finite differences would take 1600x2x16x27 = 1,382,400 forward calculations

Reconstruction of Infants' Brain I

Original MRI data

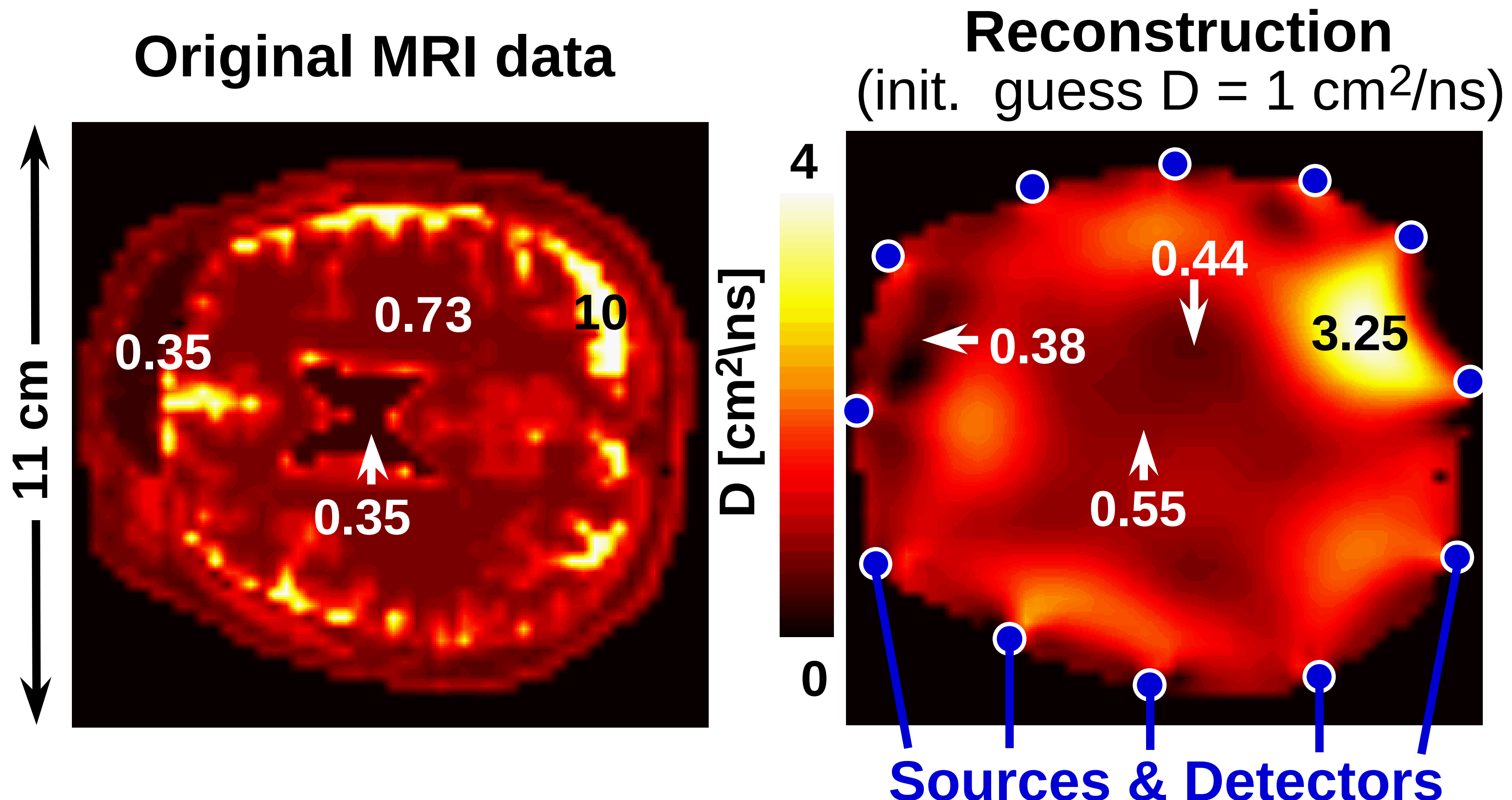
Reconstruction
(init. guess D = 1 cm^2/ns)



hematoma (left side);
cerebrospinal fluid pocket
(upper right)

(60 iterations ~ 70 min)

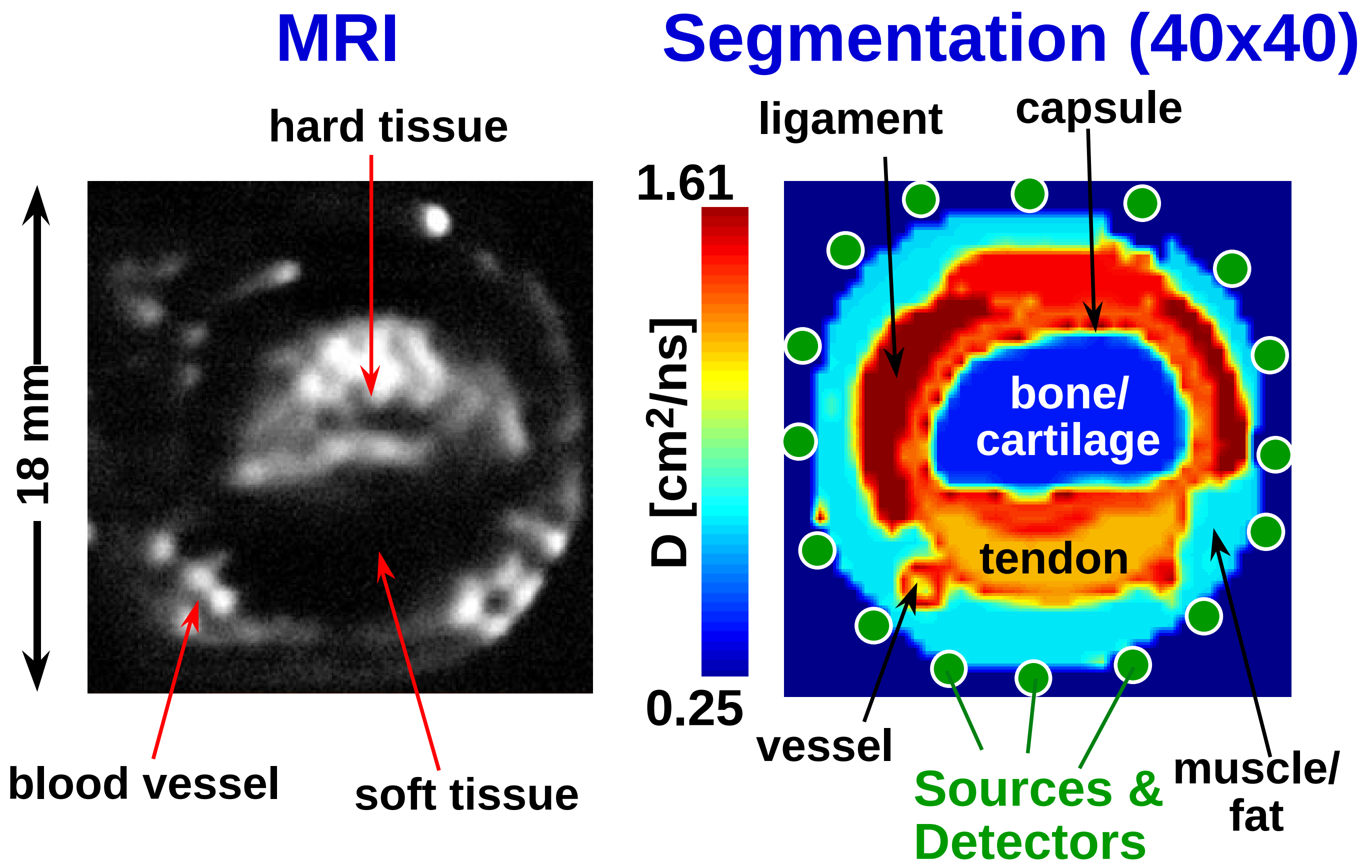
Reconstruction of Infants' Brain II



blood-filled ventricle (occurs in 15-30% of all preterm infants)

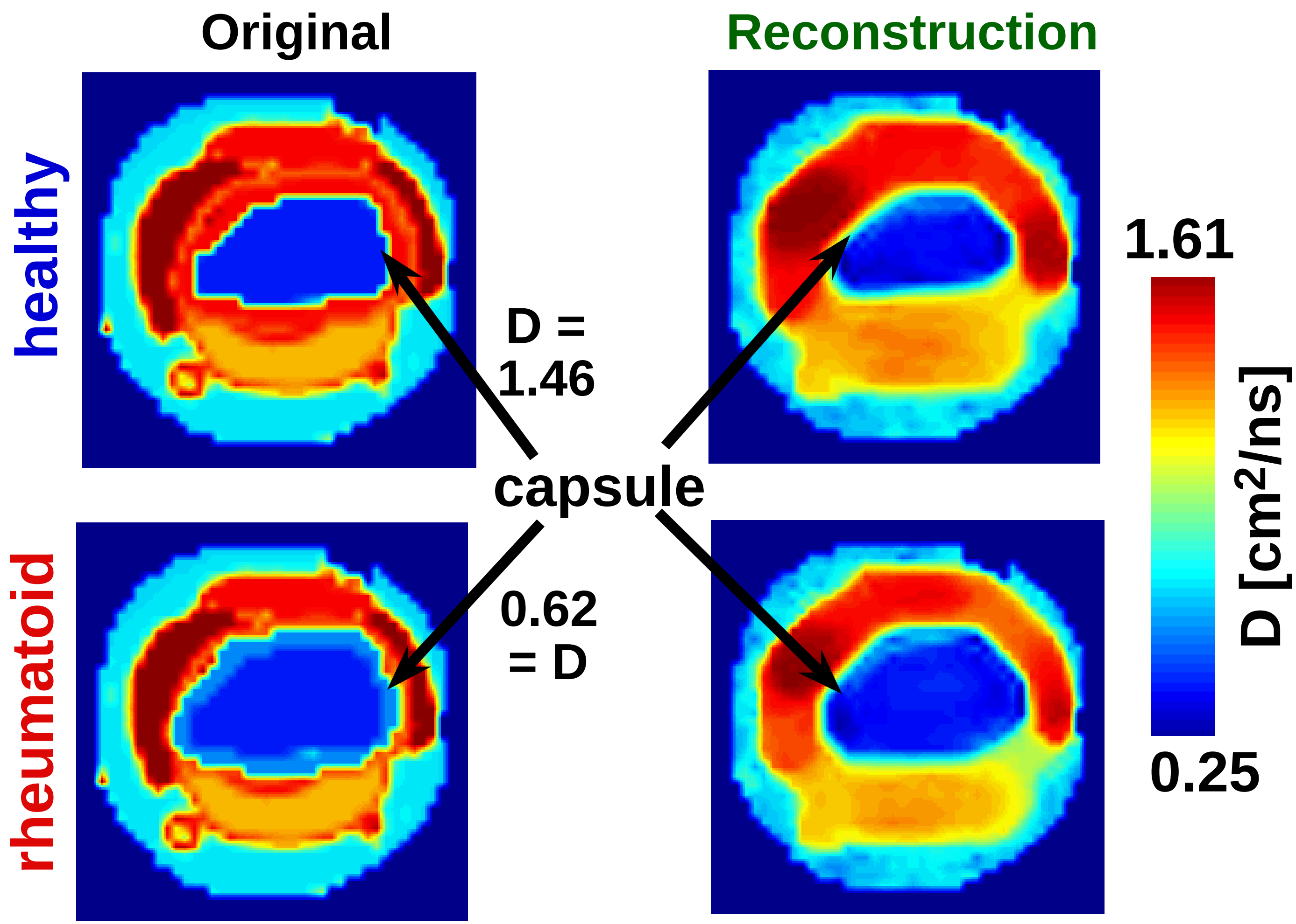
(60 iterations ~ 70 min)

Create Optical Image of Finger Joint

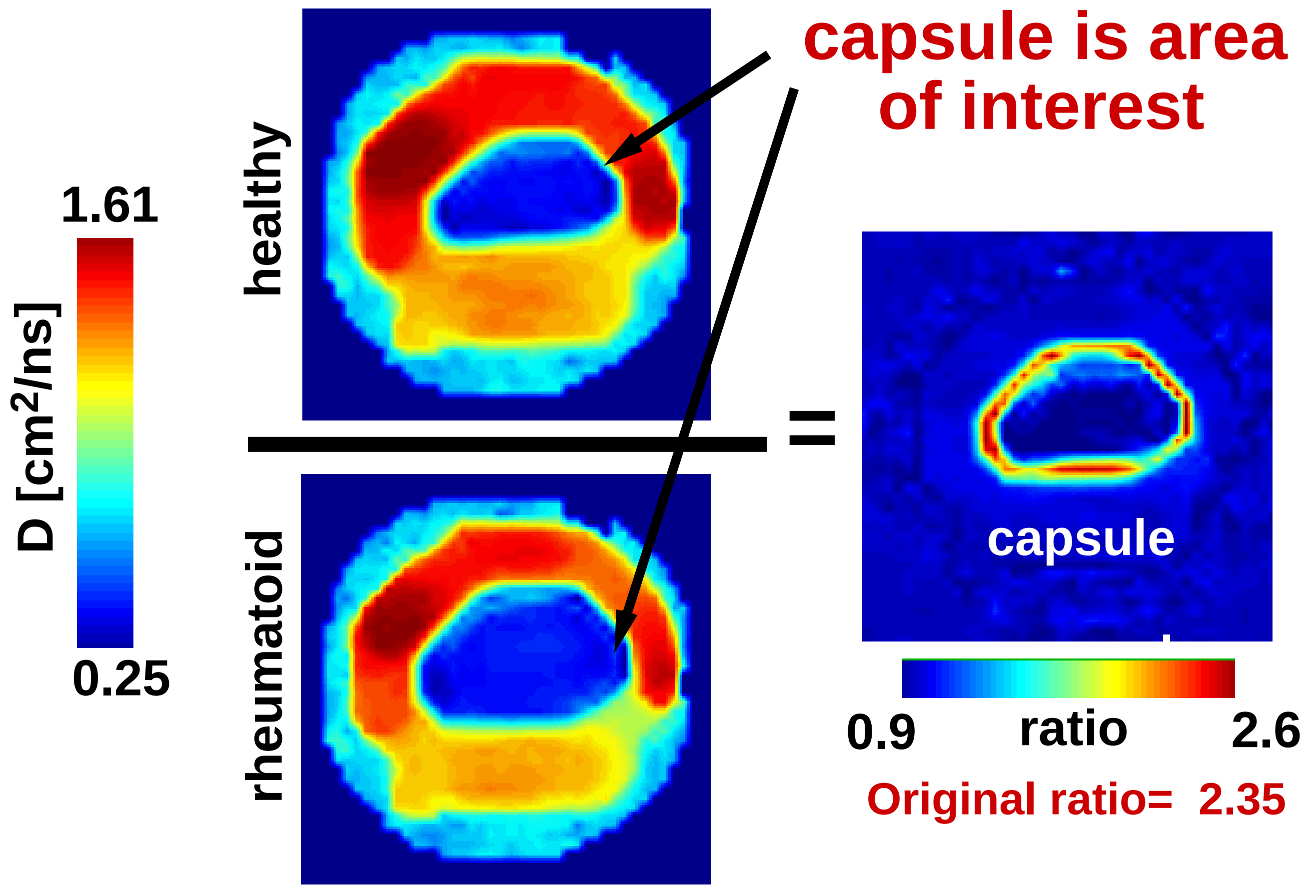


Simulated data obtained from optical image for source and detector positions shown

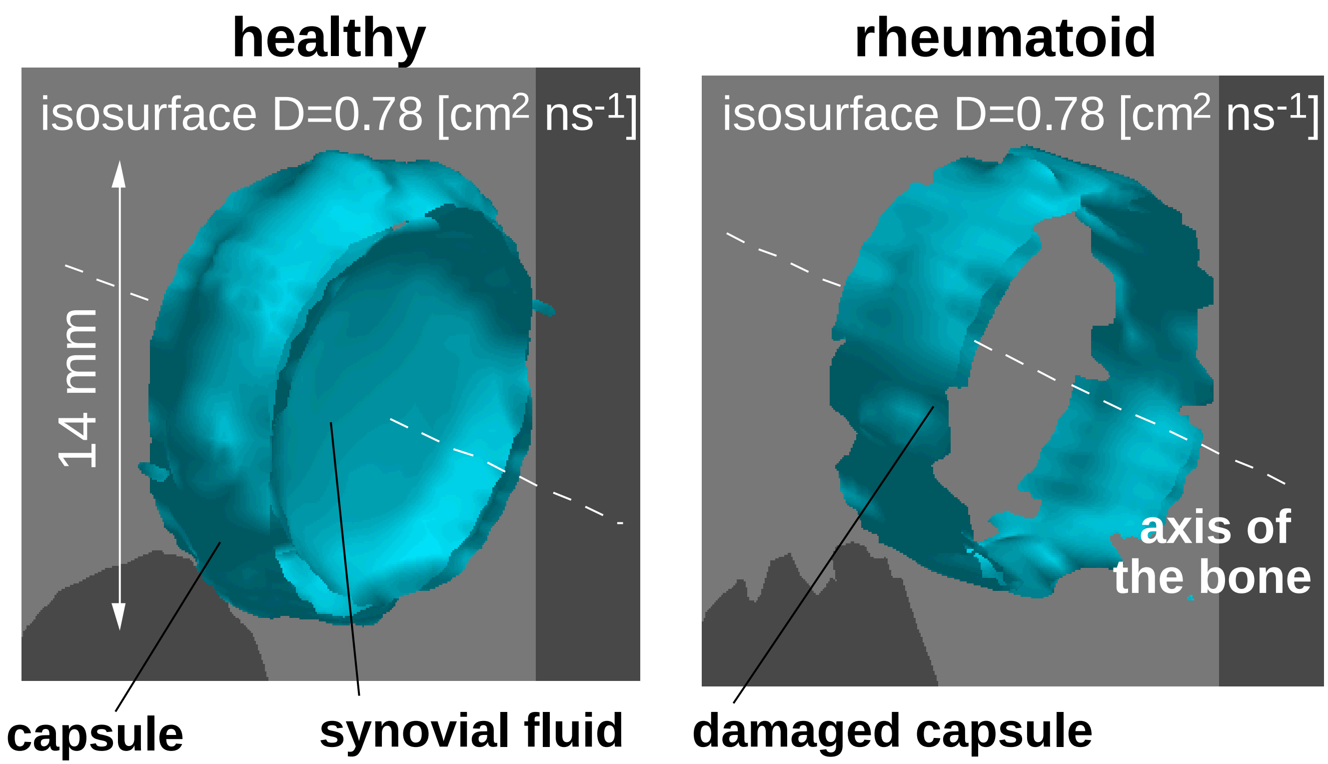
Reconstruction of Capsule



Reconstruction of Capsule



3D Reconstruction of Capsule



3D volume: 9 x 30 x 30 = 8100 voxels
8 sources x 8 detectors x 4 layers

Further applications

- Applications under development
  - inversion of transport equation (Alex Klose, FU Berlin)
  - oceanographics (Ralf Giering, JPL)
  - hydrodynamics (Rudy Henninger, LANL)
- General approach would be useful in
  - reconstruction: imaging through refractive media (seismology, medical and NDE ultrasound), . . .
  - matching large-scale simulations to data: atmosphere and ocean models, fluid dynamics, hydrodynamics
  - optimization in large engineering design problems: optical lens, geometry of integrated circuits, aerodynamic shape, engines

Potential extensions of adjoint differentiation

- Higher order derivatives
  - $\frac{\partial^2 \varphi}{\partial x_i \partial x_j}$ requires 2 forward and 2 adjoint calculations
  - large intermediate matrices => restrict to $\sum_j \frac{\partial^2 \varphi}{\partial x_i \partial x_j} x_j$
- Incorporate derivatives into data structures
  - with each variable vector $\mathbf{x}$, associate $\delta\mathbf{x}$ and $\partial / \partial\mathbf{x}$
  - for each transformation $f(\mathbf{x})$, associate $\frac{\partial f}{\partial \mathbf{x}}$ with capabilities for forward and adjoint propagation
  - useful in symbolic languages, such as Maple (S. Gull)
  - facilitated in object-oriented setting (Bayes Inference Engine)
- Construct new programming paradigm based on these composite data structures in OO environment
  - view computer code as establishing links between transforms

Bibliography

- “Inversion based on computational simulations,” K. M. Hanson, et al., in *Maximum Entropy and Bayesian Methods*, pp. 121-135 (Kluwer, 1998)
- “Model-based image reconstruction from time-resolved diffusion data,” S. S. Saquib, et al., *Proc. SPIE* **3034**, pp. 369-380 (1997)
- “Gradient-based iterative image reconstruction scheme for time-resolved optical tomography,” A. H. Hielscher, et al., *IEEE Trans. Med. Imag.* **18**, pp. 262-271 (1999)
- “Two- and three-dimensional optical tomography of finger joints for diagnostics of rheumatoid arthritis,” A. D. Klose, et al., *Proc. SPIE* **3566** (1998)

Find these under author’s home page http://home.lanl.gov/kmh/