# Image reconstruction for MRI: to FFT or not?

Jeffrey A. Fessler

EECS Department
The University of Michigan

Mar. 16, 2006
ECE Department, Michigan State University

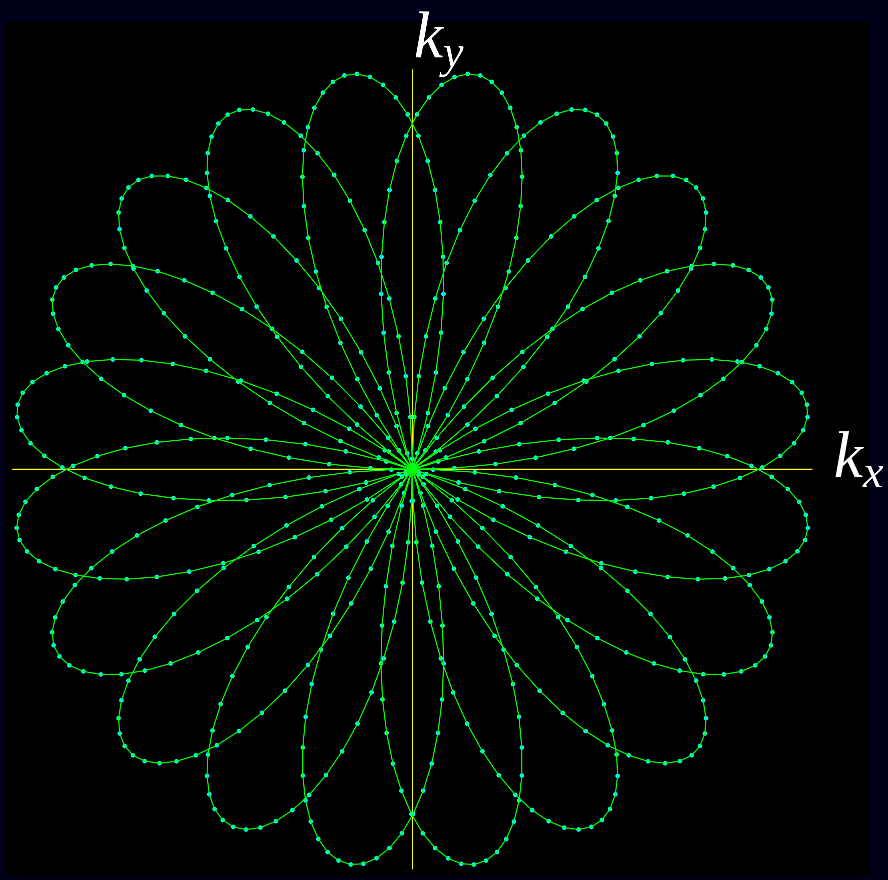
Acknowledgements: Doug Noll, Brad Sutton,
Chunyu Yip, Will Grissom

# Outline

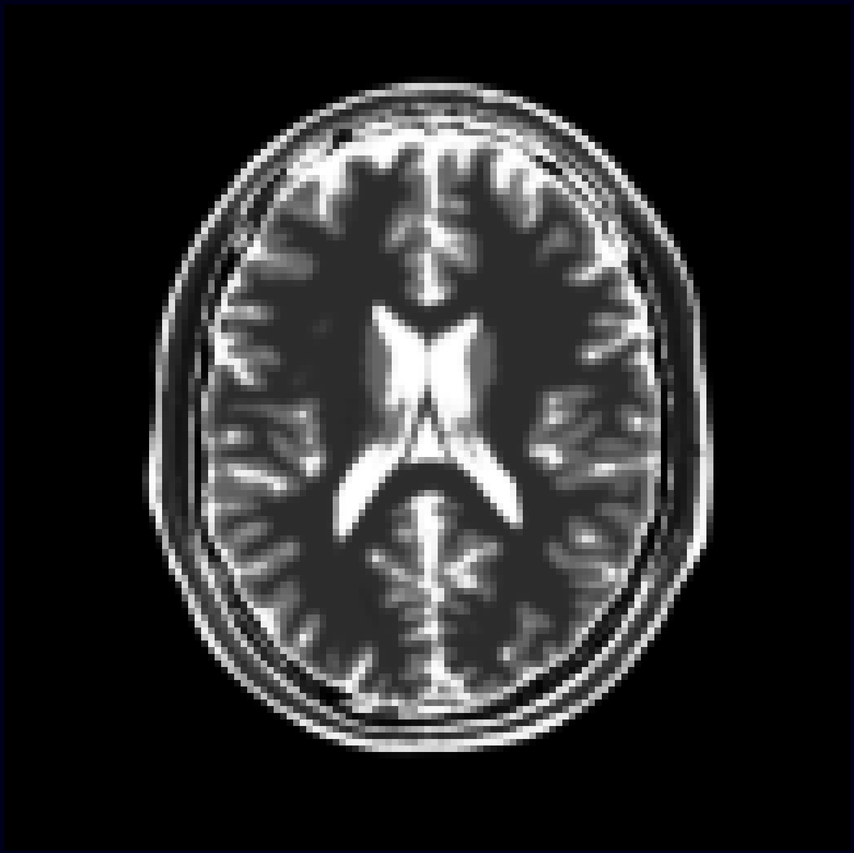
- MR image reconstruction
- Model-based reconstruction
- Iterations and computation (NUFFT etc.)
- Regularization approach
- Examples

# MR Image Reconstruction

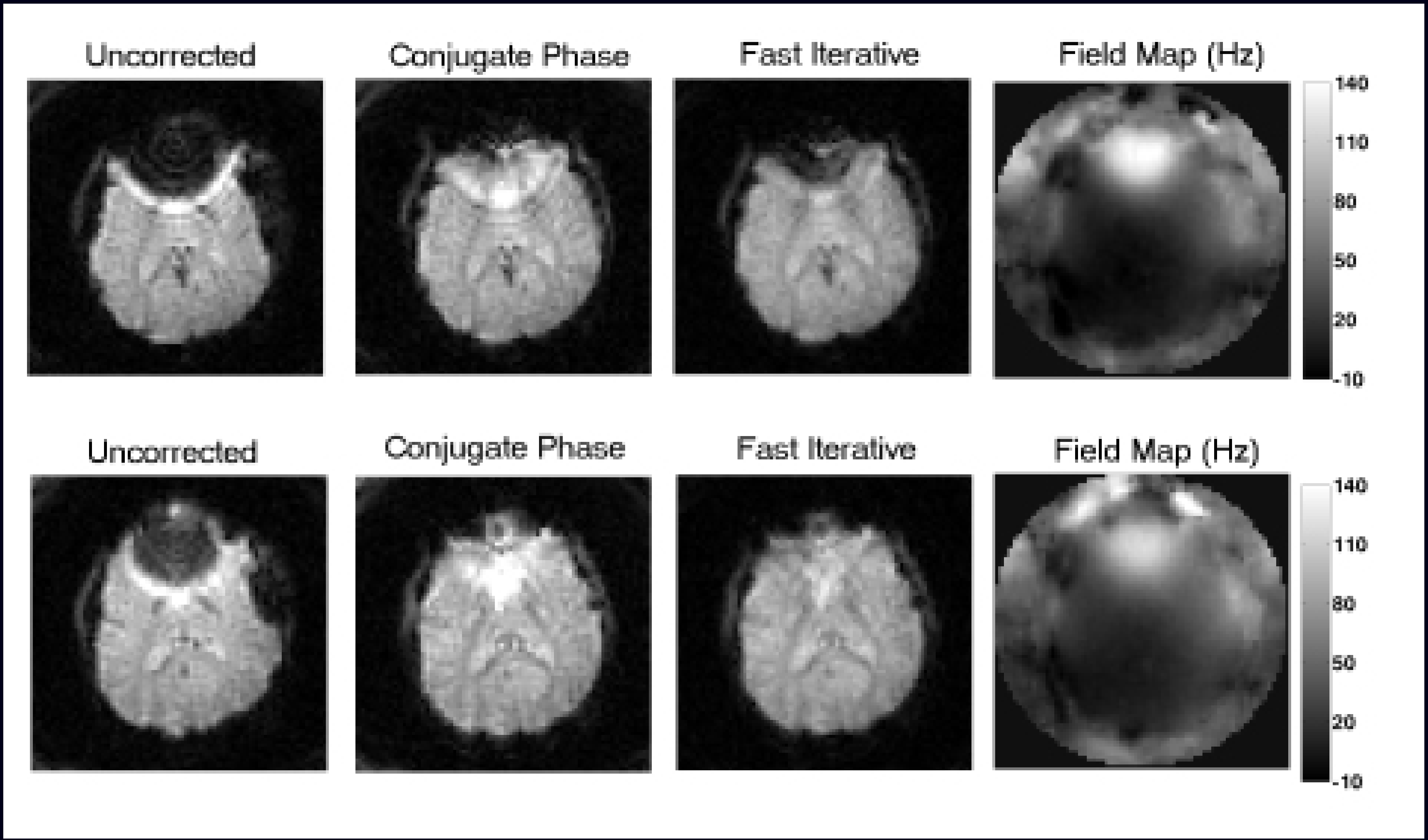
“k-space” image



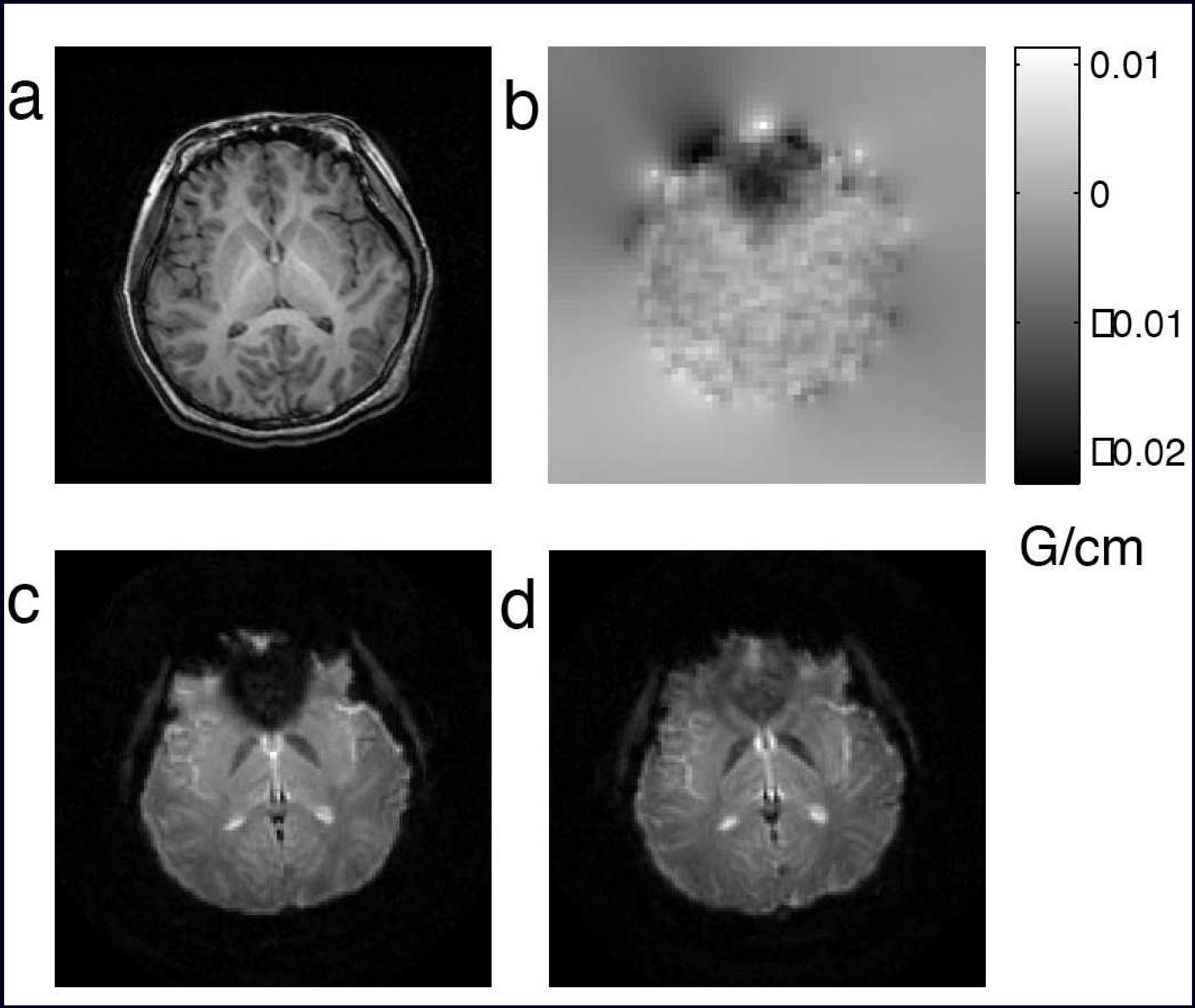

$\Longrightarrow$



# Example: Iterative Reconstruction under $\Delta B_0$



# Example: Iterative Pulse Sequence Design



# Textbook MRI Measurement Model

Ignoring *lots* of things:

$$y_i = s(t_i) + \text{noise}_i, \qquad i = 1, \ldots, N$$

$$s(t) = \int f(\vec{r})\, \mathrm{e}^{-\imath 2\pi \vec{k}(t) \cdot \vec{r}} \, \mathrm{d}\vec{r},$$

where $\vec{r}$ denotes spatial position, and
$\vec{k}(t)$ denotes the "k-space trajectory" of the MR pulse sequence, determined by user-controllable magnetic field gradients.

$\mathrm{e}^{-\imath 2\pi \vec{k}(t) \cdot \vec{r}}$ provides spatial information $\Longrightarrow$ Nobel Prize

- MRI measurements are (roughly) samples of the Fourier transform $F(\vec{k})$ of the object's transverse magnetization $f(\vec{r})$.
- Basic image reconstruction problem:
  recover $f(\vec{r})$ from measurements $\{y_i\}_{i=1}^N$.

Inherently under-determined (ill posed) problem
$\Longrightarrow$ no canonical solution.

# Image Reconstruction Strategies

The unknown object $f(\vec{r})$ is a continuous-space function, but the recorded measurements $\boldsymbol{y} = (y_1, \ldots, y_N)$ are finite.

Options?

- Continuous-discrete formulation using many-to-one linear model:
  $$\boldsymbol{y} = \mathcal{A} f + \boldsymbol{\varepsilon}.$$
  Minimum norm solution (*cf.* "natural pixels"):
  $$\min_{\hat{f}} \left\| \hat{f} \right\| \text{ subject to } \boldsymbol{y} = \mathcal{A} \hat{f}$$
  $$\hat{f} = \mathcal{A}^* (\mathcal{A}\mathcal{A}^*)^{-1} \boldsymbol{y} = \textstyle\sum_{i=1}^N c_i \, \mathrm{e}^{-\imath 2\pi \vec{k}(t) \cdot \vec{r}}, \text{ where } \mathcal{A}\mathcal{A}^* \boldsymbol{c} = \boldsymbol{y}.$$
- Discrete-discrete formulation
  Assume parametric model for object:
  $$f(\vec{r}) = \sum_{j=1}^{M} f_j \, b_j(\vec{r}).$$

- Continuous-continuous formulation

  Pretend that a continuum of measurements are available:

$$F(\vec{k}) = \int f(\vec{r})\, \mathrm{e}^{-\imath 2\pi \vec{k}\cdot\vec{r}}\, \mathrm{d}\vec{r},$$

  vs samples $y_i = F(\vec{k}_i) + \varepsilon_i$.

  The "solution" is an inverse Fourier transform:

$$f(\vec{r}) = \int F(\vec{k})\, \mathrm{e}^{\imath 2\pi \vec{k}\cdot\vec{r}}\, \mathrm{d}\vec{k}.$$

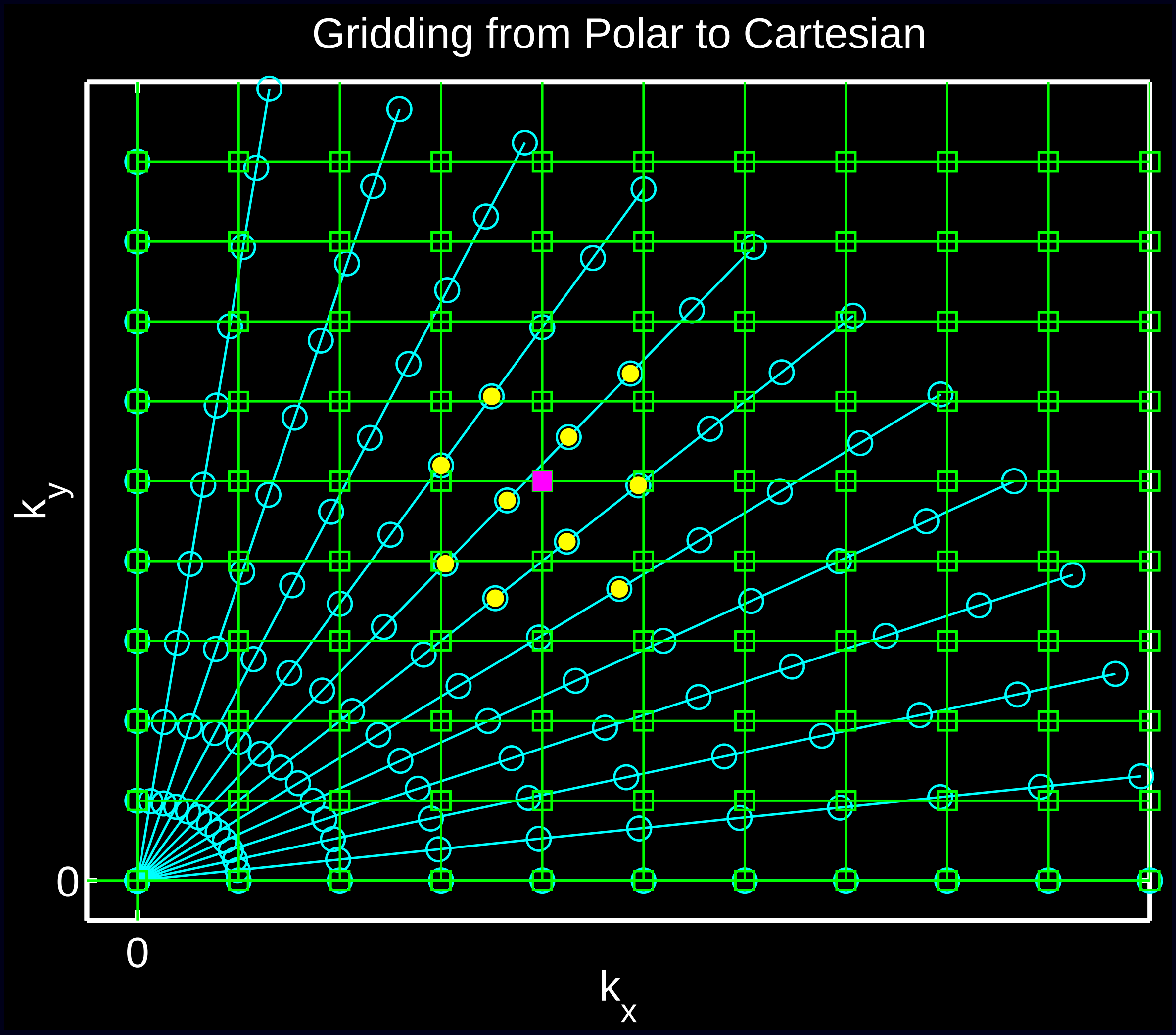
  Now discretize the integral solution (two approximations!):

$$\hat{f}(\vec{r}) = \sum_{i=1}^{N} F(\vec{k}_i)\, \mathrm{e}^{\imath 2\pi \vec{k}\cdot\vec{r}}\, w_i \approx \sum_{i=1}^{N} y_i\, \mathrm{e}^{\imath 2\pi \vec{k}\cdot\vec{r}}\, w_i,$$

  where $w_i$ values are "sampling density compensation factors."
  Numerous methods for choosing $w_i$ value in the literature.

# Conventional MR Image Reconstruction

1. Interpolate measurements onto rectilinear grid ("gridding")
2. Apply inverse FFT to estimate samples of $f(\vec{r})$



# Limitations of Gridding Reconstruction

1. Artifacts/inaccuracies due to interpolation
2. Contention about sample density "weighting"
3. Oversimplifications of Fourier transform signal model:
   - Magnetic field inhomogeneity
   - Magnetization decay ($T_2$)
   - Eddy currents
   - ...
4. Sensitivity encoding ?
5. ...

# Model-Based Image Reconstruction

MR signal equation with more complete physics:

$$s(t) = \int f(\vec{r})\, s_{\text{coil}}(\vec{r})\, \mathrm{e}^{-\imath\,\omega(\vec{r})\,t}\, \mathrm{e}^{-R_2^*(\vec{r})\,t}\, \mathrm{e}^{-\imath 2\pi \vec{k}(t)\cdot\vec{r}}\, \mathrm{d}\vec{r}$$

$$y_i = s(t_i) + \text{noise}_i, \qquad i = 1, \ldots, N$$

- $s_{\text{coil}}(\vec{r})$ Receive-coil sensitivity pattern(s) (for SENSE)
- $\omega(\vec{r})$ Off-resonance frequency map
  (due to field inhomogeneity / magnetic susceptibility)
- $R_2^*(\vec{r})$ Relaxation map

Other factors (?)

- Eddy current effects; in $\vec{k}(t)$
- Concomitant gradient terms
- Chemical shift
- Motion

Goal?

(it depends)

# Field Inhomogeneity-Corrected Reconstruction

$$s(t) = \int f(\vec{r})\, s_{\text{coil}}(\vec{r})\, e^{-\imath \omega(\vec{r}) t}\, e^{-R_2^*(\vec{r})\, t}\, e^{-\imath 2\pi \vec{k}(t) \cdot \vec{r}}\, d\vec{r}$$

Goal: reconstruct $f(\vec{r})$ given field map $\omega(\vec{r})$.
(Assume all other terms are known or unimportant.)

## Motivation

Essential for functional MRI of brain regions near sinus cavities!

(Sutton *et al.*, ISMRM 2001; T-MI 2003)

# Sensitivity-Encoded (SENSE) Reconstruction

$$s(t) = \int f(\vec{r}) s_{\text{coil}}(\vec{r})\, \mathrm{e}^{-\imath\,\omega(\vec{r})\,t}\, \mathrm{e}^{-R_2^*(\vec{r})\,t}\, \mathrm{e}^{-\imath 2\pi \vec{k}(t)\cdot\vec{r}}\, \mathrm{d}\vec{r}$$

Goal: reconstruct $f(\vec{r})$ given sensitivity maps $s_{\text{coil}}(\vec{r})$.
(Assume all other terms are known or unimportant.)

Can combine SENSE with field inhomogeneity correction “easily.”

(Sutton *et al.*, ISMRM 2001, Olafsson *et al.*, ISBI 2006)

# Joint Estimation of Image and Field-Map

$$s(t) = \int f(\vec{r})\, s_{\mathrm{coil}}(\vec{r})\, \mathrm{e}^{-\imath \omega(\vec{r}) t}\, \mathrm{e}^{-R_2^*(\vec{r})\, t}\, \mathrm{e}^{-\imath 2\pi \vec{k}(t) \cdot \vec{r}}\, d\vec{r}$$

Goal: estimate *both* the image $f(\vec{r})$ and the field map $\omega(\vec{r})$
(Assume all other terms are known or unimportant.)

Analogy:
joint estimation of emission image and attenuation map in PET.

(Sutton *et al.*, ISMRM Workshop, 2001; ISBI 2002; ISMRM 2002; ISMRM 2003; MRM 2004)

# The Kitchen Sink

$$s(t) = \int f(\vec{r})\, s_{\text{coil}}(\vec{r})\, \mathrm{e}^{-\imath\omega(\vec{r})t}\, \mathrm{e}^{-R_2^*(\vec{r})\,t}\, \mathrm{e}^{-\imath 2\pi \vec{k}(t)\cdot\vec{r}}\, \mathrm{d}\vec{r}$$

Goal: estimate image $f(\vec{r})$, field map $\omega(\vec{r})$, *and* relaxation map $R_2^*(\vec{r})$

Requires "suitable" k-space trajectory.

(Sutton *et al.*, ISMRM 2002; Twieg, MRM, 2003)

# Estimation of Dynamic Maps

$$s(t) = \int f(\vec{r})\, s_{\text{coil}}(\vec{r})\, \mathrm{e}^{-\imath\omega(\vec{r})t}\, \mathrm{e}^{-R_2^*(\vec{r})t}\, \mathrm{e}^{-\imath 2\pi \vec{k}(t)\cdot\vec{r}}\, \mathrm{d}\vec{r}$$

Goal: estimate dynamic field map $\omega(\vec{r})$ and “BOLD effect” $R_2^*(\vec{r})$ given baseline image $f(\vec{r})$ in fMRI.

Motion...

# Back to Basic Signal Model

$$s(t) = \int f(\vec{r})\, e^{-\imath 2\pi \vec{k}(t)\cdot\vec{r}}\, d\vec{r}$$

Goal: reconstruct $f(\vec{r})$ from $\boldsymbol{y} = (y_1, \ldots, y_N)$, where $y_i = s(t_i) + \varepsilon_i$.

Series expansion of unknown object:

$$f(\vec{r}) \approx \sum_{j=1}^{M} f_j\, b(\vec{r} - \vec{r}_j) \longleftarrow \text{usually 2D rect functions.}$$

$$y_i \approx \int \left[\sum_{j=1}^{M} f_j\, b(\vec{r} - \vec{r}_j)\right] e^{-\imath 2\pi \vec{k}(t_i)\cdot\vec{r}}\, d\vec{r} = \sum_{j=1}^{M} \left[\int b(\vec{r} - \vec{r}_j)\, e^{-\imath 2\pi \vec{k}(t_i)\cdot\vec{r}}\, d\vec{r}\right] f_j$$

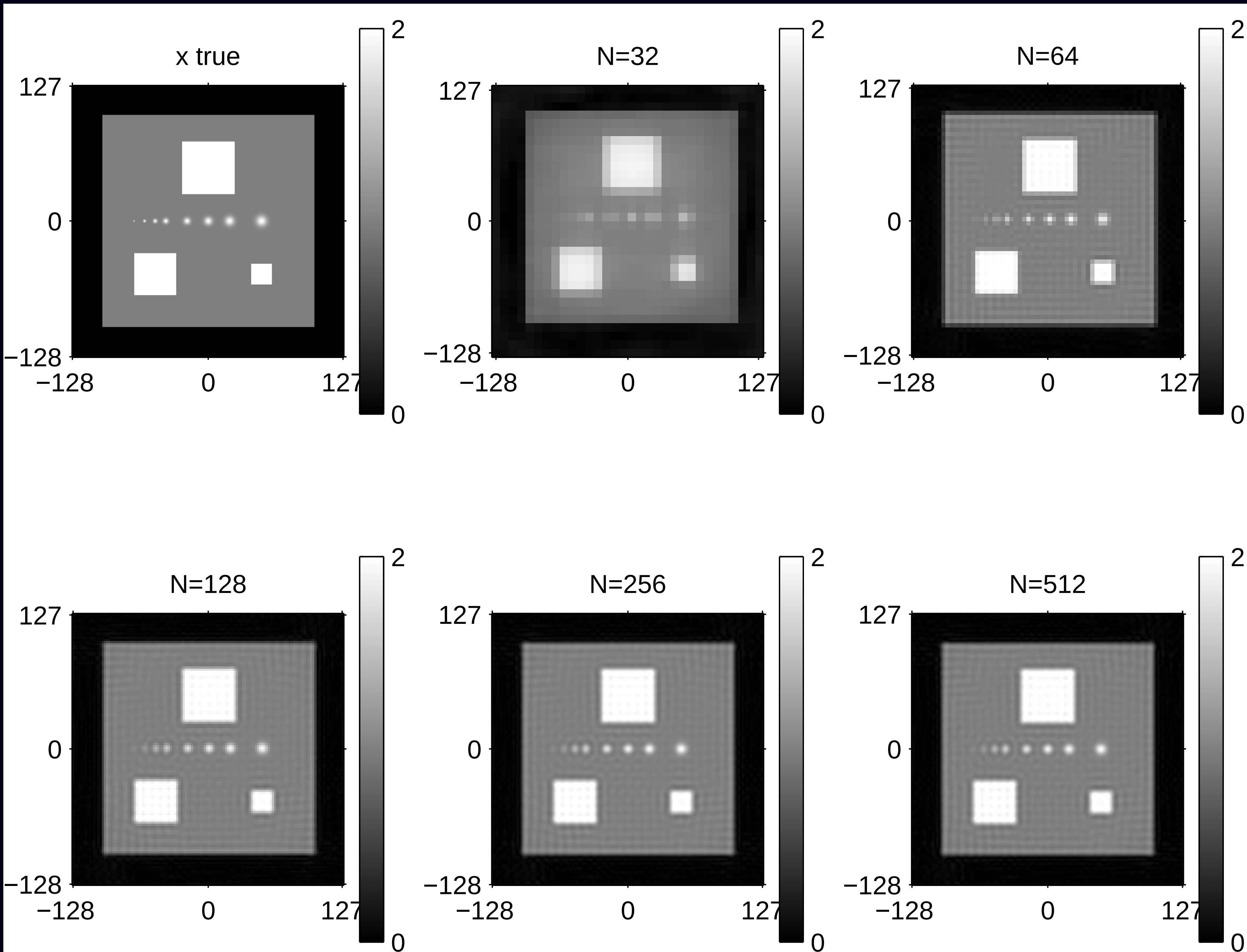
$$= \sum_{j=1}^{M} a_{ij} f_j, \qquad a_{ij} = B(\vec{k}(t_i))\, e^{-\imath 2\pi \vec{k}(t_i)\cdot\vec{r}_j}, \qquad b(\vec{r}) \overset{\text{FT}}{\Longleftrightarrow} B(\vec{k}).$$

Discrete-discrete measurement model with system matrix $\boldsymbol{A} = \{a_{ij}\}$:

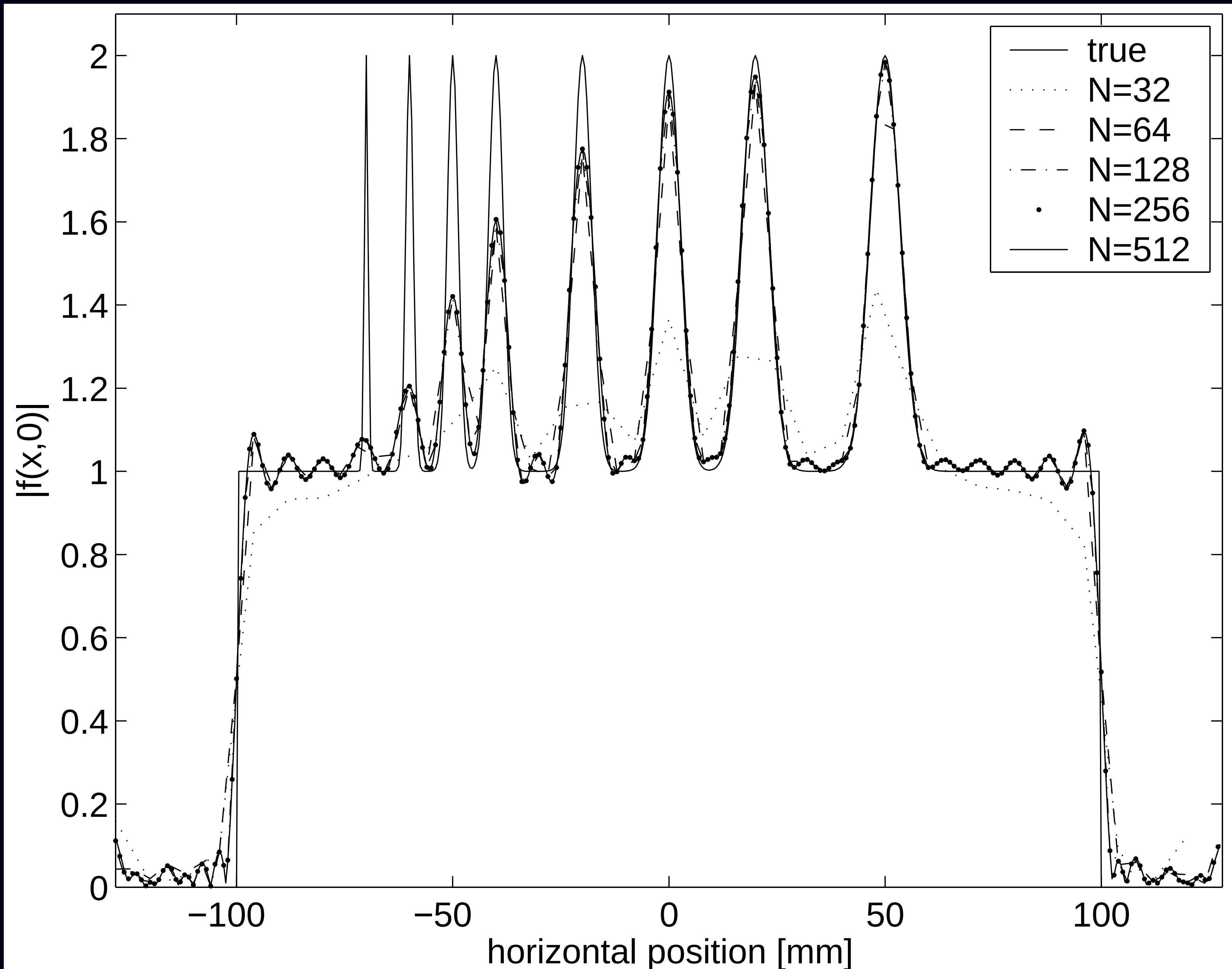
$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{f} + \boldsymbol{\varepsilon}.$$

Goal: estimate coefficients (pixel values) $\boldsymbol{f} = (f_1, \ldots, f_M)$ from $\boldsymbol{y}$.

# Small Pixel Size Does Not Matter



# Profiles



true
N=32
N=64
N=128
N=256
N=512
|f(x,0)|
horizontal position [mm]
−100
−50
0
50
100
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2

# Regularized Least-Squares Estimation

$$\hat{\boldsymbol{f}} = \arg\min_{\boldsymbol{f} \in \mathbb{C}^M} \Psi(\boldsymbol{f}), \qquad \Psi(\boldsymbol{f}) = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{f}\|^2 + \alpha R(\boldsymbol{f})$$

- data fit term $\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{f}\|^2$
  corresponds to negative log-likelihood of Gaussian distribution
- regularizing roughness penalty term $R(\boldsymbol{f})$ controls noise

$$R(\boldsymbol{f}) \approx \int \|\nabla f\|^2 \, \mathrm{d}\vec{r}$$

- regularization parameter $\alpha > 0$
  controls tradeoff between spatial resolution and noise
  (Fessler & Rogers, IEEE T-IP, 1996)
- Equivalent to Bayesian MAP estimation with prior $\propto \mathrm{e}^{-\alpha R(\boldsymbol{f})}$

Quadratic regularization $R(\boldsymbol{f}) = \|\boldsymbol{C}\boldsymbol{f}\|^2$ leads to closed-form solution:

$$\hat{\boldsymbol{f}} = \left[\boldsymbol{A}'\boldsymbol{A} + \alpha \boldsymbol{C}'\boldsymbol{C}\right]^{-1} \boldsymbol{A}'\boldsymbol{y}$$

(a formula of limited practical use)

# Iterative Minimization by Conjugate Gradients

Choose initial guess $\boldsymbol{f}^{(0)}$ (*e.g.*, fast conjugate phase / gridding).
Iteration (unregularized):

$$
\begin{aligned}
&\boldsymbol{g}^{(n)} = \nabla\Psi\left(\boldsymbol{f}^{(n)}\right) = \boldsymbol{A}'(\boldsymbol{A}\boldsymbol{f}^{(n)} - \boldsymbol{y}) && \text{gradient} \\
&\boldsymbol{p}^{(n)} = \boldsymbol{P}\boldsymbol{g}^{(n)} && \text{precondition} \\
&\gamma_n = \begin{cases} 0, & n = 0 \\ \dfrac{\langle \boldsymbol{g}^{(n)}, \boldsymbol{p}^{(n)} \rangle}{\langle \boldsymbol{g}^{(n-1)}, \boldsymbol{p}^{(n-1)} \rangle}, & n > 0 \end{cases} && \\
&\boldsymbol{d}^{(n)} = -\boldsymbol{p}^{(n)} + \gamma_n \boldsymbol{d}^{(n-1)} && \text{search direction} \\
&\boldsymbol{v}^{(n)} = \boldsymbol{A}\boldsymbol{d}^{(n)} && \\
&\alpha_n = \langle \boldsymbol{d}^{(n)}, -\boldsymbol{g}^{(n)} \rangle / \langle \boldsymbol{A}\boldsymbol{f}^{(n)}, \boldsymbol{A}\boldsymbol{f}^{(n)} \rangle && \text{step size} \\
&\boldsymbol{f}^{(n+1)} = \boldsymbol{f}^{(n)} + \alpha_n \boldsymbol{d}^{(n)} && \text{update}
\end{aligned}
$$

Bottlenecks: computing $\boldsymbol{A}\boldsymbol{f}$ and $\boldsymbol{A}'\boldsymbol{y}$.

- $\boldsymbol{A}$ is too large to store explicitly (not sparse)
- Even if $\boldsymbol{A}$ were stored, directly computing $\boldsymbol{A}\boldsymbol{f}$ is $O(NM)$ per iteration, whereas FFT is only $O(N \log N)$.

# Computing $Af$ Rapidly

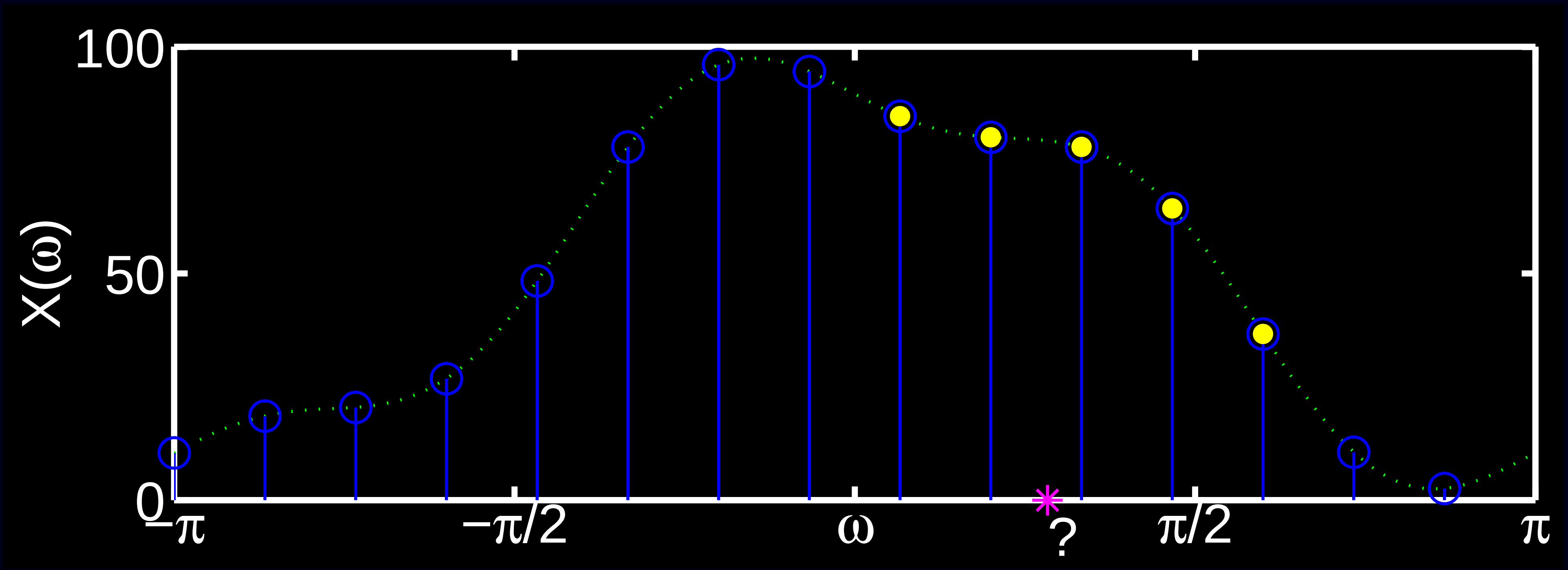
$$[\boldsymbol{A f}]_i = \sum_{j=1}^{M} a_{ij} f_j = B(\vec{k}(t_i)) \sum_{j=1}^{M} \mathrm{e}^{-\imath 2\pi \vec{k}(t_i) \cdot \vec{r}_j} f_j, \qquad i = 1, \ldots, N$$

- Pixel locations $\{\vec{r}_j\}$ are uniformly spaced
- k-space locations $\left\{\vec{k}(t_i)\right\}$ are unequally spaced

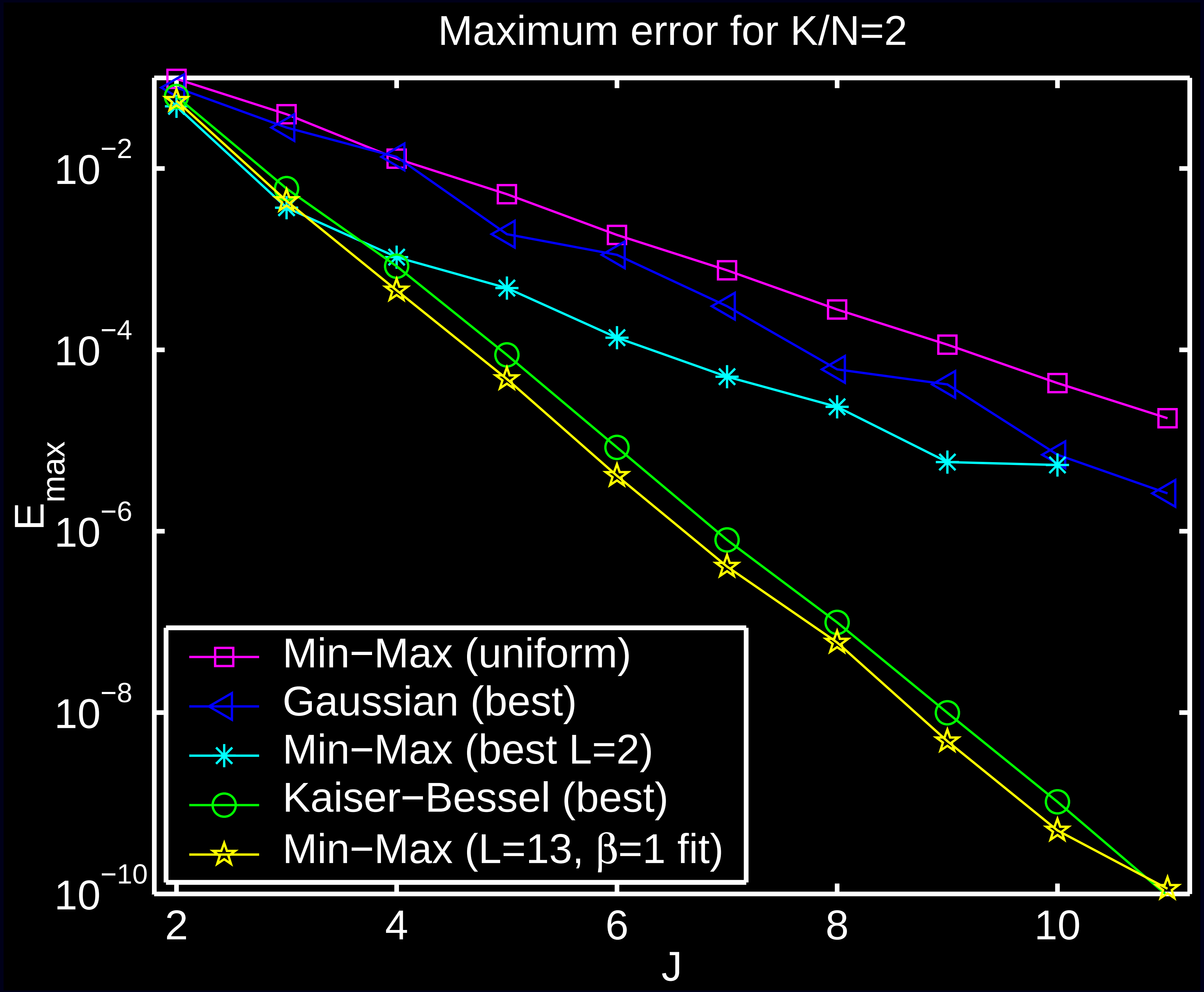
$\Longrightarrow$ needs nonuniform fast Fourier transform (NUFFT)

# NUFFT (Type 2)

- Compute over-sampled FFT of equally-spaced signal samples
- Interpolate onto desired unequally-spaced frequency locations
- Dutt & Rokhlin, SIAM JSC, 1993, Gaussian bell interpolator
- Fessler & Sutton, IEEE T-SP, 2003, min-max interpolator
  and min-max optimized Kaiser-Bessel interpolator.
  NUFFT toolbox: `http://www.eecs.umich.edu/~fessler/code`



# Worst-Case NUFFT Interpolation Error



# Further Acceleration using Toeplitz Matrices

Cost-function gradient:

$$\begin{aligned} \boldsymbol{g}^{(n)} &= \boldsymbol{A}'(\boldsymbol{A}\boldsymbol{f}^{(n)} - \boldsymbol{y}) \\ &= \boldsymbol{T}\boldsymbol{f}^{(n)} - \boldsymbol{b}, \end{aligned}$$

where

$$\boldsymbol{T} \triangleq \boldsymbol{A}'\boldsymbol{A}, \qquad \boldsymbol{b} \triangleq \boldsymbol{A}'\boldsymbol{y}.$$

In the absence of field inhomogeneity, the matrix $\boldsymbol{T}$ is Toeplitz.
Computing $\boldsymbol{T}\boldsymbol{f}^{(n)}$ requires an ordinary ($2\times$ over-sampled) FFT.

Precomputing the first column of $\boldsymbol{T}$ and $\boldsymbol{b}$ requires a couple NUFFTs.
(Wajer, ISMRM 2001, Eggers ISMRM 2002, Liu ISMRM 2005)

In the presence of field inhomogeneity, the matrix $\boldsymbol{T}$ is not Toeplitz.
But accurate approximations are feasible.
(Fessler *et al.*, IEEE T-SP, Sep. 2005, brain imaging special issue)

# Field inhomogeneity?

Combine NUFFT with min-max temporal interpolator
(Sutton *et al.*, IEEE T-MI, 2003)
(forward version of "time segmentation", Noll, T-MI, 1991)

Recall:

$$s(t) = \int f(\vec{r})\, \mathrm{e}^{-\imath \omega(\vec{r}) t}\, \mathrm{e}^{-\imath 2\pi \vec{k}(t) \cdot \vec{r}}\, \mathrm{d}\vec{r}$$

Temporal interpolation approximation (aka "time segmentation"):

$$\mathrm{e}^{-\imath\, \omega(\vec{r})\, t} \approx \sum_{l=1}^{L} a_l(t)\, \mathrm{e}^{-\imath\, \omega(\vec{r})\, \tau_l}$$

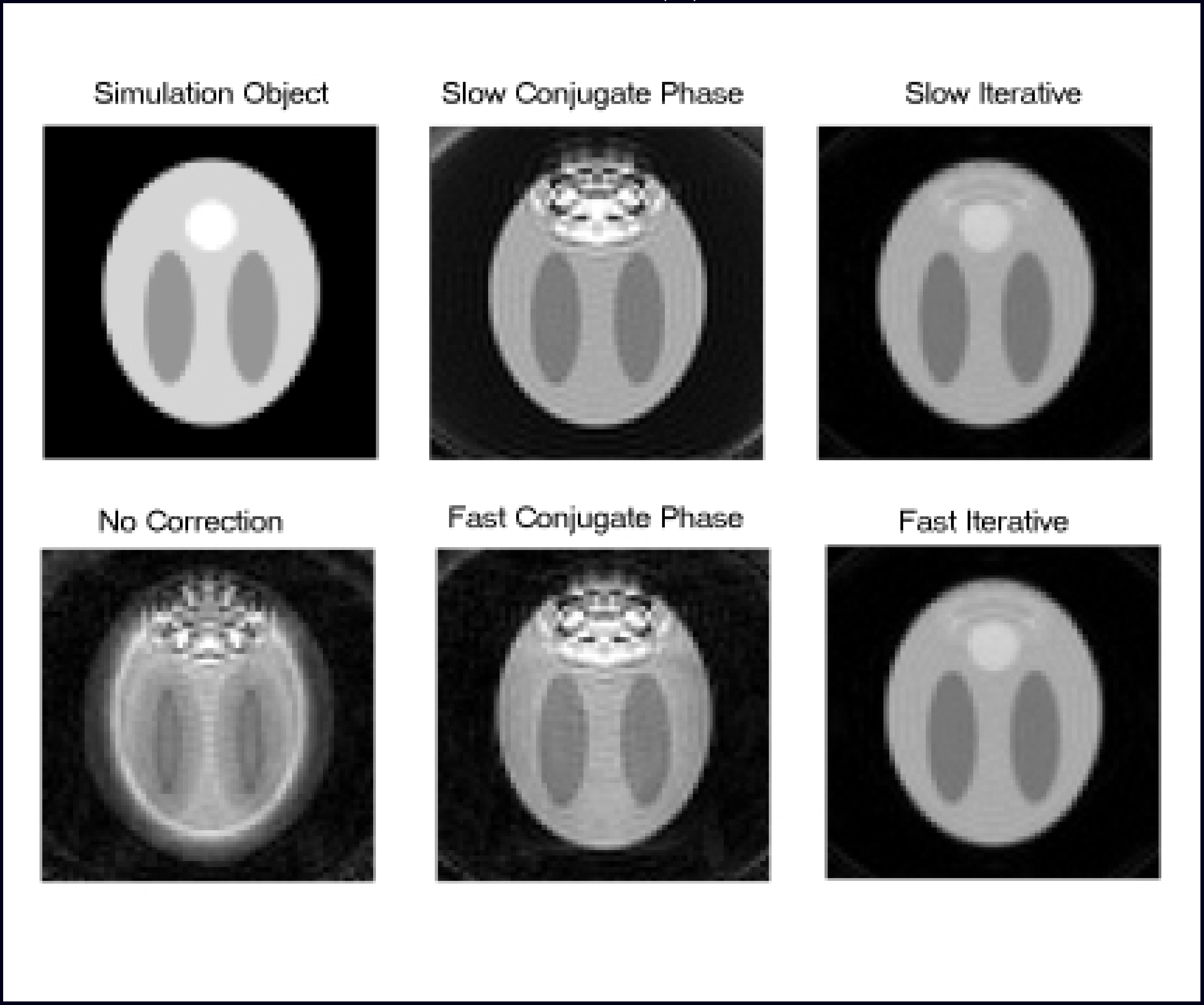
for min-max optimized temporal interpolation functions $\{a_l(\cdot)\}_{l=1}^{L}$.

$$s(t) \approx \sum_{l=1}^{L} a_l(t) \int \left[ f(\vec{r})\, \mathrm{e}^{-\imath\, \omega(\vec{r})\, \tau_l} \right] \mathrm{e}^{-\imath 2\pi \vec{k}(t) \cdot \vec{r}}\, \mathrm{d}\vec{r}$$

Linear combination of $L$ NUFFT calls.

# Field Corrected Reconstruction Example

Simulation using known field map $\omega(\vec{r})$.

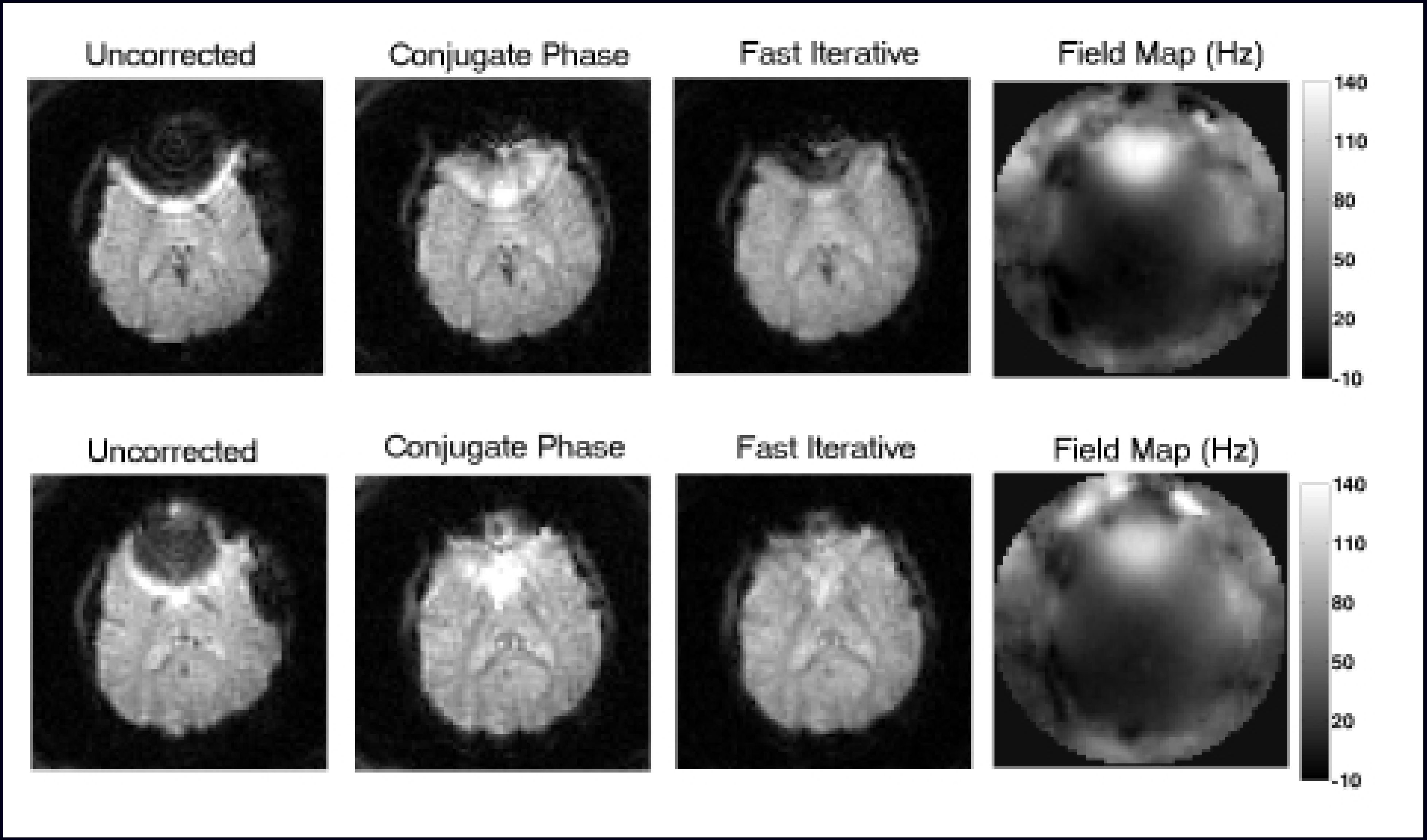


# Simulation Quantitative Comparison

- Computation time?
- NRMSE between $\hat{f}$ and $f^{\text{true}}$?

| Reconstruction Method | Time (s) | NRMSE | NRMSE |
|---|---|---|---|
| | | complex | magnitude |
| No Correction | 0.06 | 1.35 | 0.22 |
| Full Conjugate Phase | 4.07 | 0.31 | 0.19 |
| Fast Conjugate Phase | 0.33 | 0.32 | 0.19 |
| Fast Iterative (10 iters) | 2.20 | 0.04 | 0.04 |
| Exact Iterative (10 iters) | 128.16 | 0.04 | 0.04 |

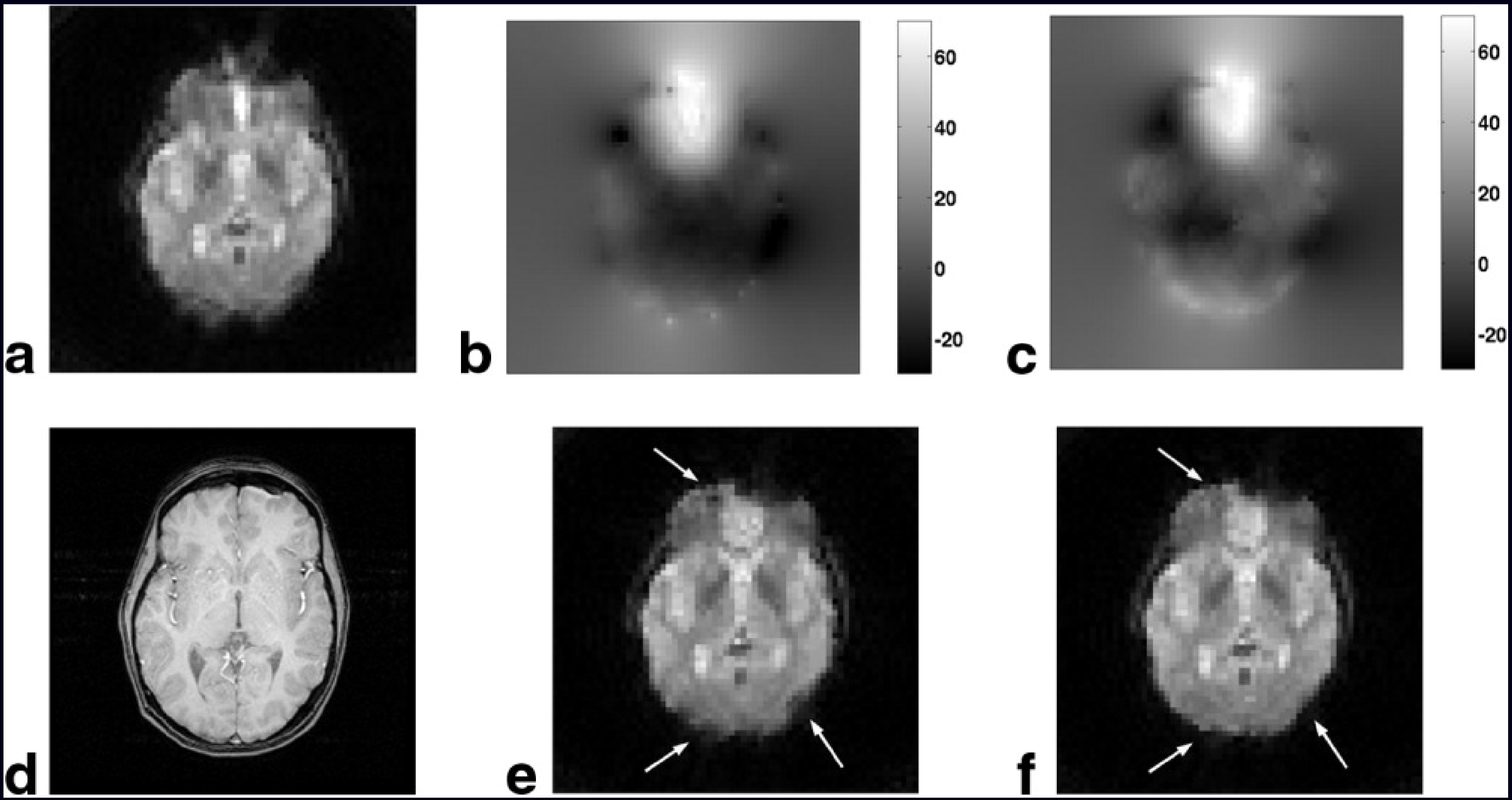
# Human Data: Field Correction



# Joint Field-Map / Image Reconstruction

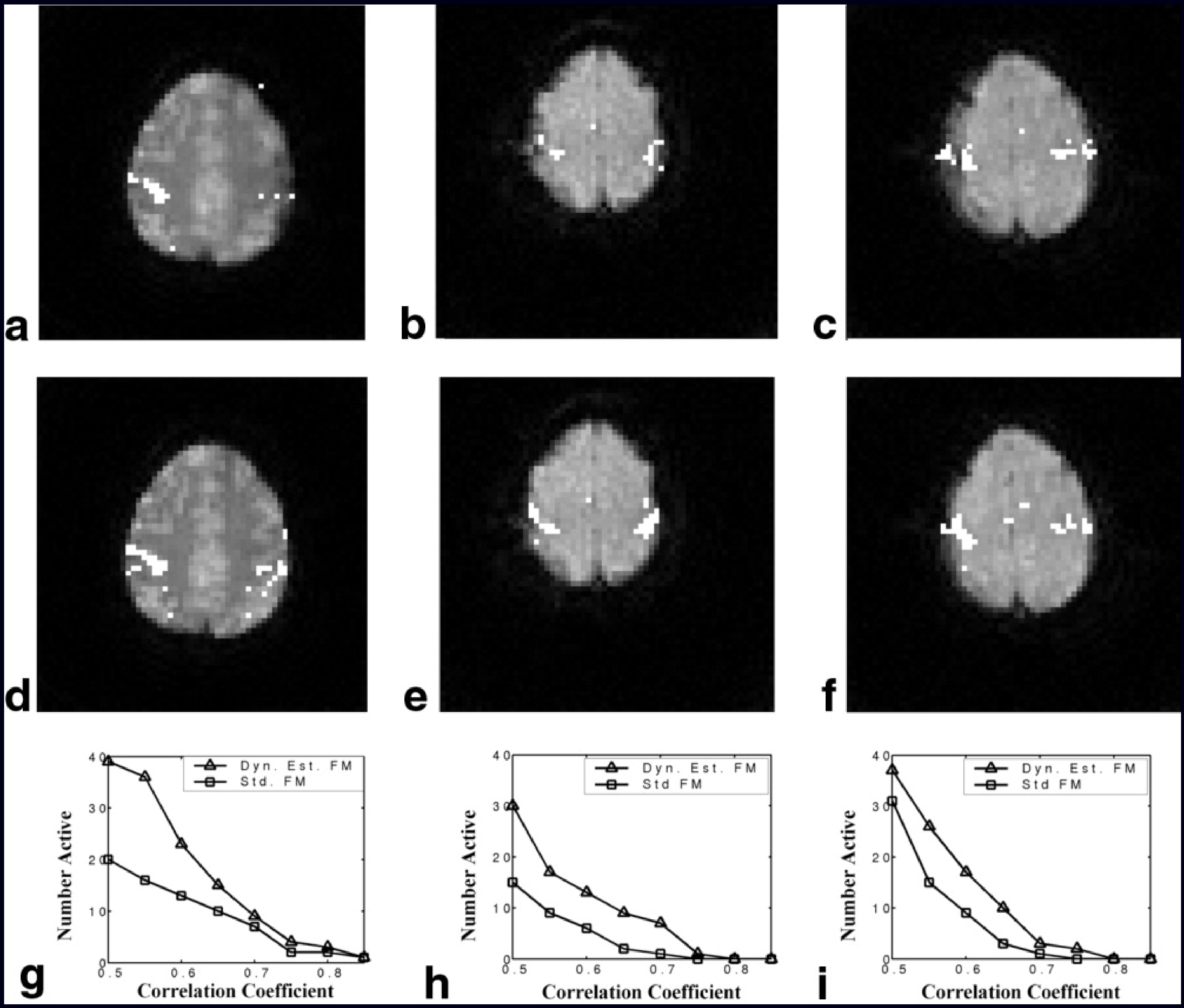
Dynamic field mapping using spiral-in / spiral-out sequence

(Sutton *et al.*, MRM, 2004).




(a) uncorr., (b) std. map, (c) joint map, (d) T1 ref, (e) using std, (f) using joint.

# Activation Results: Static vs Dynamic Field Maps



Functional results for the two reconstructions for 3 human subjects.

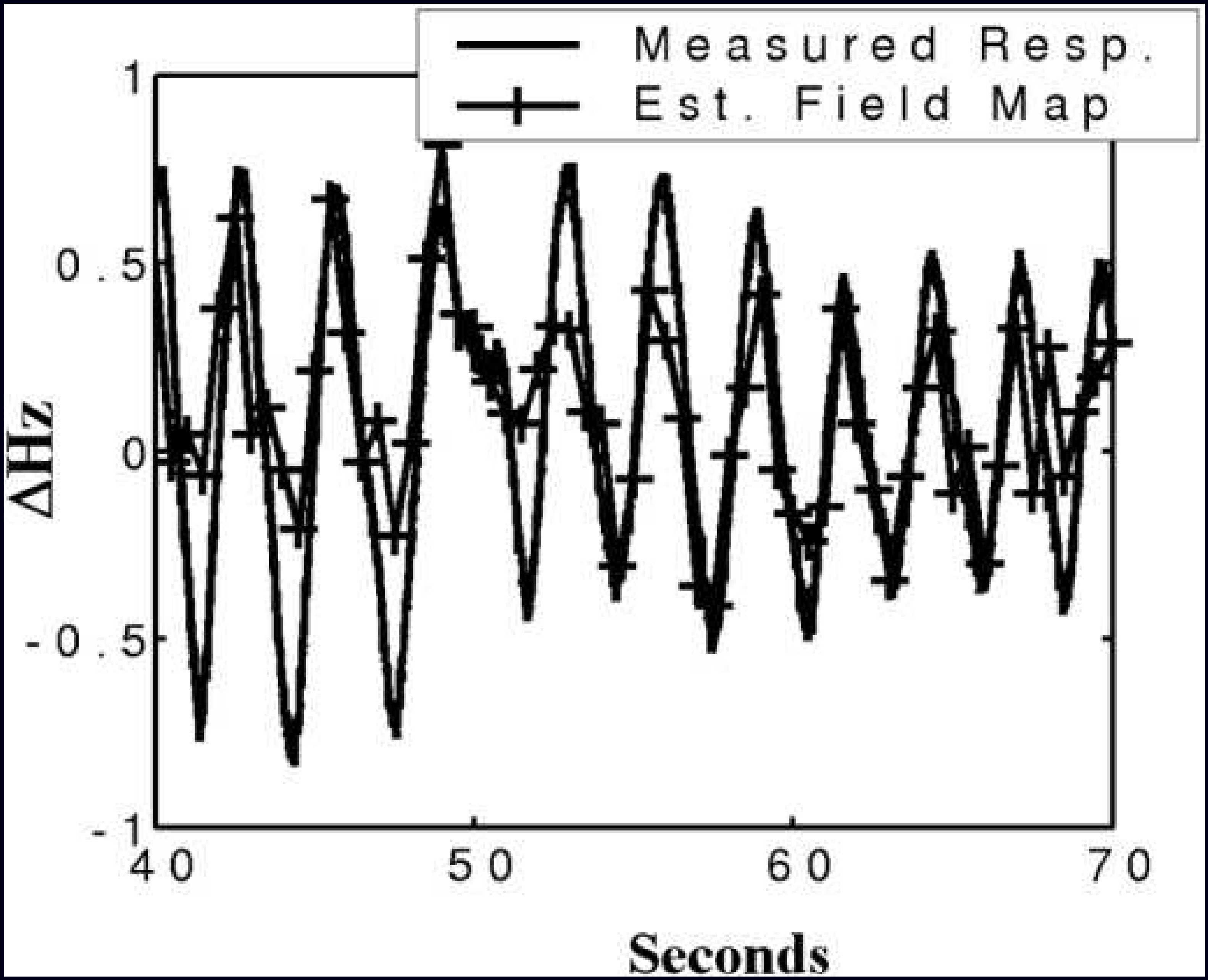
Reconstruction using the standard field map
for (a) subject 1, (b) subject 2, and (c) subject 3.

Reconstruction using the jointly estimated field map
for (d) subject 1, (e) subject 2, and (f) subject 3.

Number of pixels with correlation coefficients higher than thresholds
for (g) subject 1, (h) subject 2, and (i) subject 3.

Take home message: dynamic field mapping is possible, using iterative reconstruction as an essential tool.
(Standard field maps based on echo-time differences work poorly for spiral-in / spiral-out sequences due to phase discrepancies.)

# Tracking Respiration-Induced Field Changes



# Regularization

- Conventional regularization for MRI uses a roughness penalty for the *complex* voxel values:

$$R(\boldsymbol{f}) \approx \sum_{j=1}^{M} \left| f_j - f_{j-1} \right|^2 \qquad \text{(in 1D).}$$

- Regularizes the real and imaginary image components equally.
- In some MR studies, including BOLD fMRI:
  - magnitude of $f_j$ carries the information of interest,
  - phase of $f_j$ should be spatially smooth.
  - This *a priori* information is ignored by $R(\boldsymbol{f})$.
- Alternatives to $R(\boldsymbol{f})$:
  - Constrain $\boldsymbol{f}$ to be real?
    (Unrealistic: RF phase inhomogeneity, eddy currents, ...)
  - Determine phase of $\boldsymbol{f}$ "somehow," then estimate its magnitude.
    - Non-iteratively (Noll, Nishimura, Macovski, IEEE T-MI, 1991)
    - Iteratively (Lee, Pauly, Nishimura, ISMRM, 2003)

# Separate Magnitude/Phase Regularization

Decompose $\boldsymbol{f}$ into its "magnitude" $\boldsymbol{m}$ and phase $\boldsymbol{x}$:

$$f_j(\boldsymbol{m}, \boldsymbol{x}) = m_j e^{\imath x_j}, \qquad m_j \in \mathbb{R}, \qquad x_j \in \mathbb{R}, \qquad j = 1, \ldots, M.$$

(Allow "magnitude" $m_j$ to be negative.)

Proposed cost function with separate regularization of $\boldsymbol{m}$ and $\boldsymbol{x}$:

$$\Psi(\boldsymbol{m}, \boldsymbol{x}) = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{f}(\boldsymbol{m}, \boldsymbol{x})\|^2 + \gamma R_1(\boldsymbol{m}) + \beta R_2(\boldsymbol{x}).$$

Choose $\beta \gg \gamma$ to strongly smooth phase estimate.

Joint estimation of magnitude and phase via regularized LS:

$$(\hat{\boldsymbol{m}}, \hat{\boldsymbol{x}}) = \underset{\boldsymbol{m} \in \mathbb{R}^M,\ \boldsymbol{x} \in \mathbb{R}^M}{\arg\min} \Psi(\boldsymbol{m}, \boldsymbol{x})$$

$\Psi$ is not convex $\Longrightarrow$ need good initial estimates $(\boldsymbol{m}^{(0)}, \boldsymbol{x}^{(0)})$.

# Alternating Minimization

Magnitude Update:

$$\boldsymbol{m}^{\text{new}} = \underset{\boldsymbol{m} \in \mathbb{R}^M}{\arg\min}\, \Psi\left(\boldsymbol{m}, \boldsymbol{x}^{\text{old}}\right)$$

Phase Update:

$$\boldsymbol{x}^{\text{new}} = \underset{\boldsymbol{x} \in \mathbb{R}^M}{\arg\min}\, \Psi(\boldsymbol{m}^{\text{new}}, \boldsymbol{x}),$$

Since $f_j = m_j e^{\imath x_j}$ is linear in $m_j$, the magnitude update is easy. Apply a few iterations of slightly modified CG algorithm (constrain $\boldsymbol{m}$ to be real)

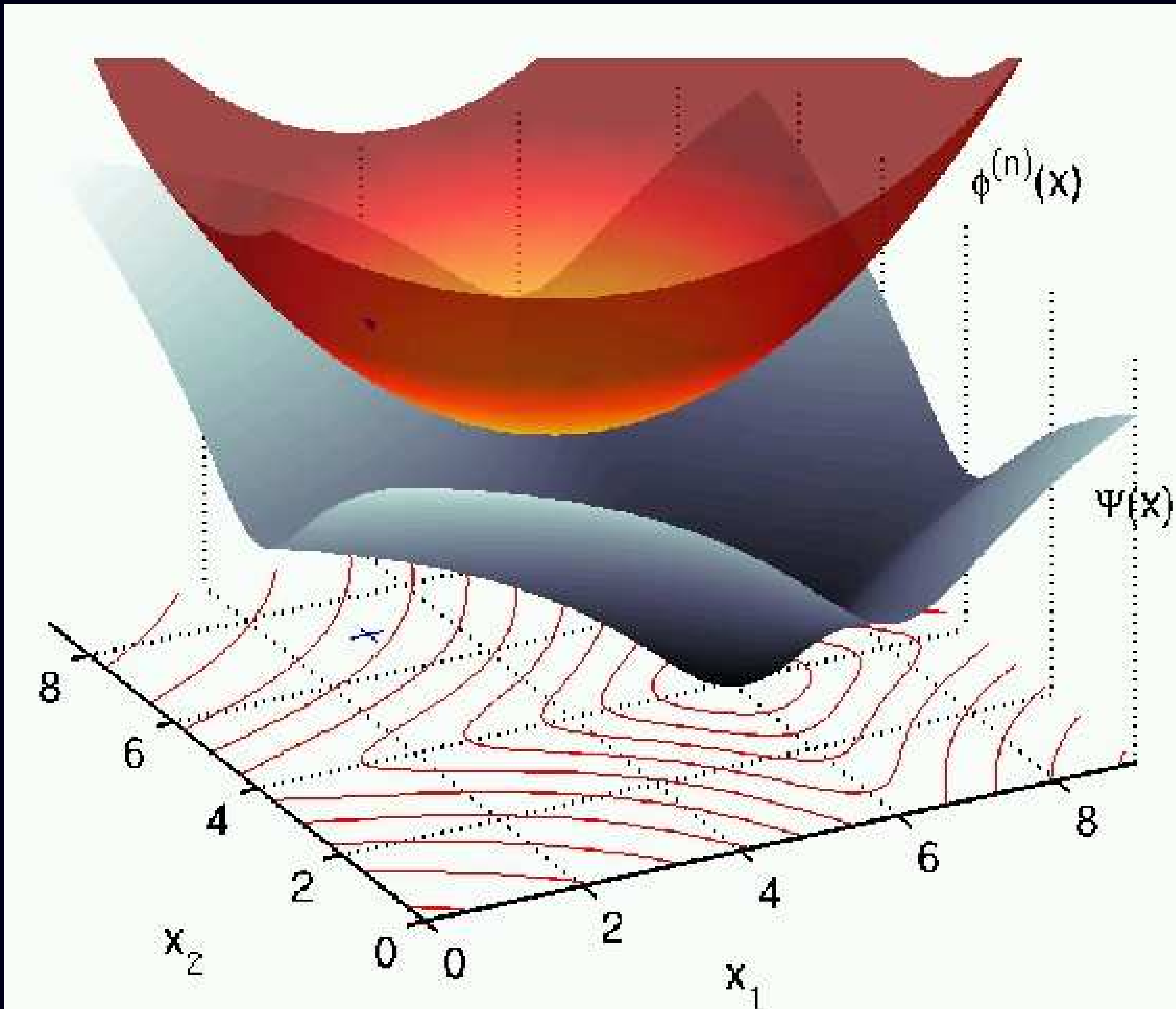
But $f_j = m_j e^{\imath x_j}$ is highly nonlinear in $\boldsymbol{x}$. Complicates "argmin."

Steepest descent?

$$\boldsymbol{x}^{(n+1)} = \boldsymbol{x}^{(n)} - \lambda \nabla_{\boldsymbol{x}} \Psi\left(\boldsymbol{m}^{\text{old}}, \boldsymbol{x}^{(n)}\right).$$

Choosing the stepsize $\lambda$ is difficult.

# Optimization Transfer



# Surrogate Functions

To minimize a cost function $\Phi(\boldsymbol{x})$, choose surrogate functions $\phi^{(n)}(\boldsymbol{x})$ that satisfy the following *majorization* conditions:

$$\begin{aligned} \phi^{(n)}\left(\boldsymbol{x}^{(n)}\right) &= \Phi(\boldsymbol{x}^{(n)}) \\ \phi^{(n)}(\boldsymbol{x}) &\geq \Phi(\boldsymbol{x}), \qquad \forall \boldsymbol{x} \in \mathbb{R}^M. \end{aligned}$$

Iteratively minimize the surrogates as follows:

$$\boldsymbol{x}^{(n+1)} = \arg\min_{\boldsymbol{x}^{(n)} \in \mathbb{R}^M} \phi^{(n)}(\boldsymbol{x}) \,.$$

This will decrease $\Phi$ monotonically; $\Phi(\boldsymbol{x}^{(n+1)}) \leq \Phi(\boldsymbol{x}^{(n)})$.

The art is in the design of surrogates.
Tradeoffs:
- complexity
- computation per iteration
- convergence rate / number of iterations.

# Surrogate Functions for MR Phase

$$\mathsf{L}(\boldsymbol{x}) \triangleq \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{f}(\boldsymbol{m},\boldsymbol{x})\|^2 = \sum_{i=1}^{N} \mathsf{h}_i([\boldsymbol{A}\boldsymbol{f}(\boldsymbol{m},\boldsymbol{x})]_i),$$

where $\mathsf{h}_i(t) \triangleq |y_i - t|^2$ is convex.

Extending De Pierro (IEEE T-MI, 1995), for $\pi_{ij} \geq 0$ and $\sum_{j=1}^{M} \pi_{ij} = 1$:

$$[\boldsymbol{A}\boldsymbol{f}(\boldsymbol{m},\boldsymbol{x})]_i = \sum_{j=1}^{M} b_{ij}\, \mathrm{e}^{\imath x_j} = \sum_{j=1}^{M} \pi_{ij} \left[\frac{b_{ij}}{\pi_{ij}} \left(\mathrm{e}^{\imath x_j} - \mathrm{e}^{\imath x_j^{(n)}}\right) + \bar{y}_i^{(n)}\right],$$

where $b_{ij} \triangleq a_{ij} m_j$, $\bar{y}_i^{(n)} \triangleq [\boldsymbol{A}\boldsymbol{f}(\boldsymbol{m},\boldsymbol{x}^{(n)})]_i$. Choose $\pi_{ij} \geq 0$ and $\sum_{j=1}^{M} \pi_{ij} = 1$.

Since $\mathsf{h}_i$ is convex:

$$\begin{aligned} \mathsf{h}_i([\boldsymbol{A}\boldsymbol{f}(\boldsymbol{m},\boldsymbol{x})]_i) &= \mathsf{h}_i\left(\sum_{j=1}^{M} \pi_{ij} \left[\frac{b_{ij}}{\pi_{ij}} \left(\mathrm{e}^{\imath x_j} - \mathrm{e}^{\imath x_j^{(n)}}\right) + \bar{y}_i^{(n)}\right]\right) \\ &\leq \sum_{j=1}^{M} \pi_{ij}\, \mathsf{h}_i\left(\frac{b_{ij}}{\pi_{ij}} \left(\mathrm{e}^{\imath x_j} - \mathrm{e}^{\imath x_j^{(n)}}\right) + \bar{y}_i^{(n)}\right), \end{aligned}$$

with equality when $\boldsymbol{x} = \boldsymbol{x}^{(n)}$.

# Separable Surrogate Function

$$\mathsf{L}(\boldsymbol{x}) = \sum_{i=1}^{N} \mathsf{h}_i([\boldsymbol{A}\boldsymbol{f}(\boldsymbol{m},\boldsymbol{x})]_i) \le \sum_{i=1}^{N}\sum_{j=1}^{M} \pi_{ij}\,\mathsf{h}_i\left(\frac{b_{ij}}{\pi_{ij}}\left(\mathrm{e}^{\imath x_j} - \mathrm{e}^{\imath x_j^{(n)}}\right) + \bar{y}_i^{(n)}\right)$$
$$= \sum_{j=1}^{M} \underbrace{\sum_{i=1}^{N} \pi_{ij}\,\mathsf{h}_i\left(\frac{b_{ij}}{\pi_{ij}}\left(\mathrm{e}^{\imath x_j} - \mathrm{e}^{\imath x_j^{(n)}}\right) + \bar{y}_i^{(n)}\right)}_{Q_j(x_j;\boldsymbol{x}^{(n)})}.$$

Construct similar surrogates $\{S_j\}$ for (convex) roughness penalty...

$$\text{Surrogate: } \phi^{(n)}(\boldsymbol{x}) = \sum_{j=1}^{M} Q_j(x_j;\boldsymbol{x}^{(n)}) + \beta S_j(x_j;\boldsymbol{x}^{(n)}).$$

Parallelizable (simultaneous) update, with 1D minimizations:

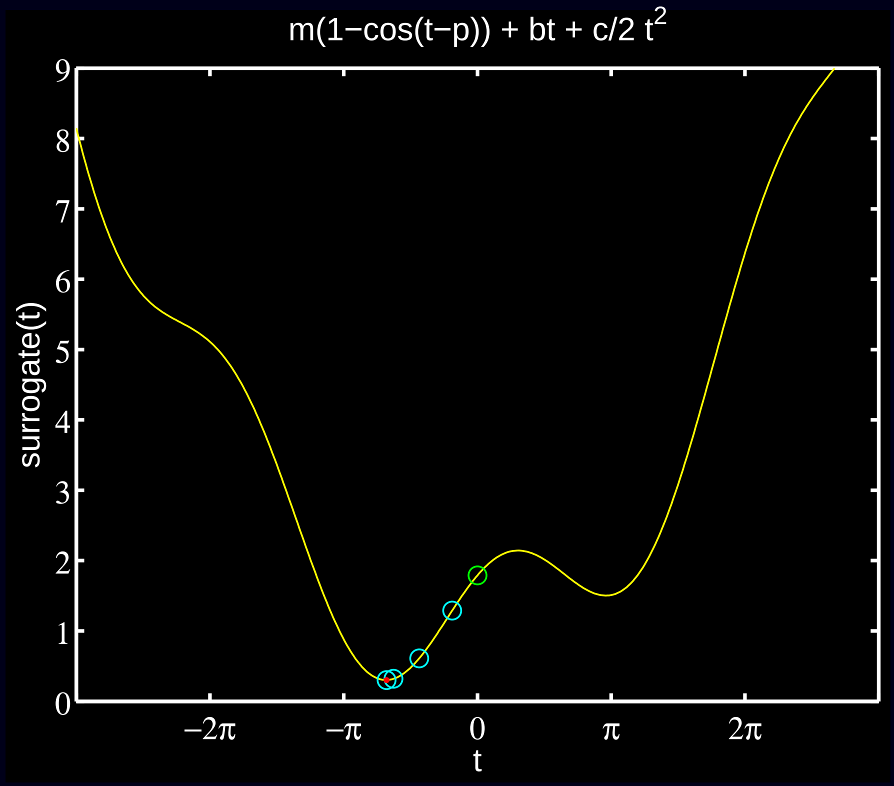
$$\boldsymbol{x}^{(n+1)} = \arg\min_{\boldsymbol{x}^{(n)}\in\mathbb{R}^M} \phi^{(n)}(\boldsymbol{x}) \implies x_j^{(n+1)} = \arg\min_{x_j\in\mathbb{R}} Q_j(x_j;\boldsymbol{x}^{(n)}) + \beta S_j(x_j;\boldsymbol{x}^{(n)}).$$

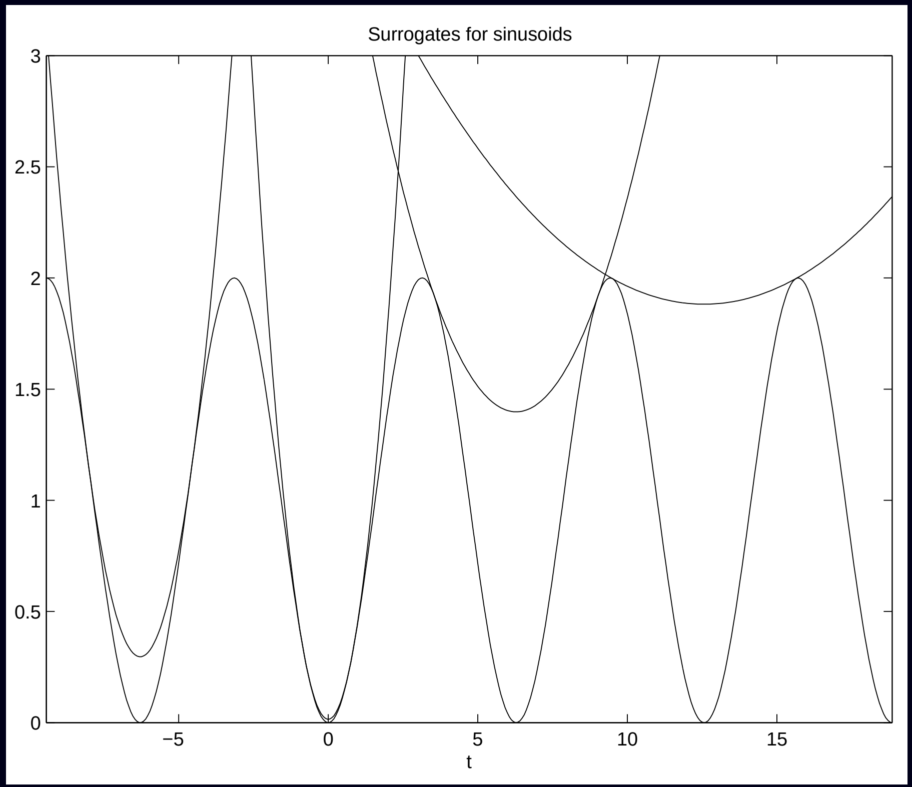
Intrinsically guaranteed to monotonically decrease the cost function.

# 1D Minimization: cos + quadratic

$$\ldots\; Q_j(x_j;\boldsymbol{x}^{(n)}) \equiv -\left|r_j^{(n)}\right| \cos\left(x_j - x_j^{(n)} - \angle r_j^{(n)}\right),$$

$$r_j^{(n)} = \left(f_j^{(n)}\right)^* [\boldsymbol{A}'(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}^{(n)})]_j + |m_j|^2 M \sum_{i=1}^{N} \left|B(\vec{k}(t_i))\right|^2$$




Simple 1D optimization transfer iterations...

# Final Algorithm for Phase Update

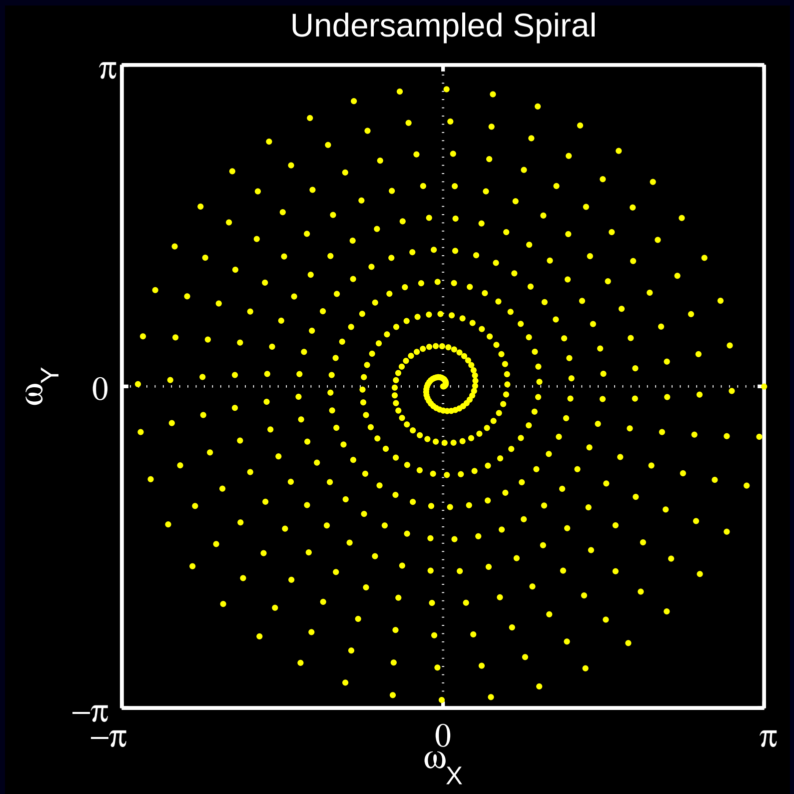
Diagonally preconditioned gradient descent:

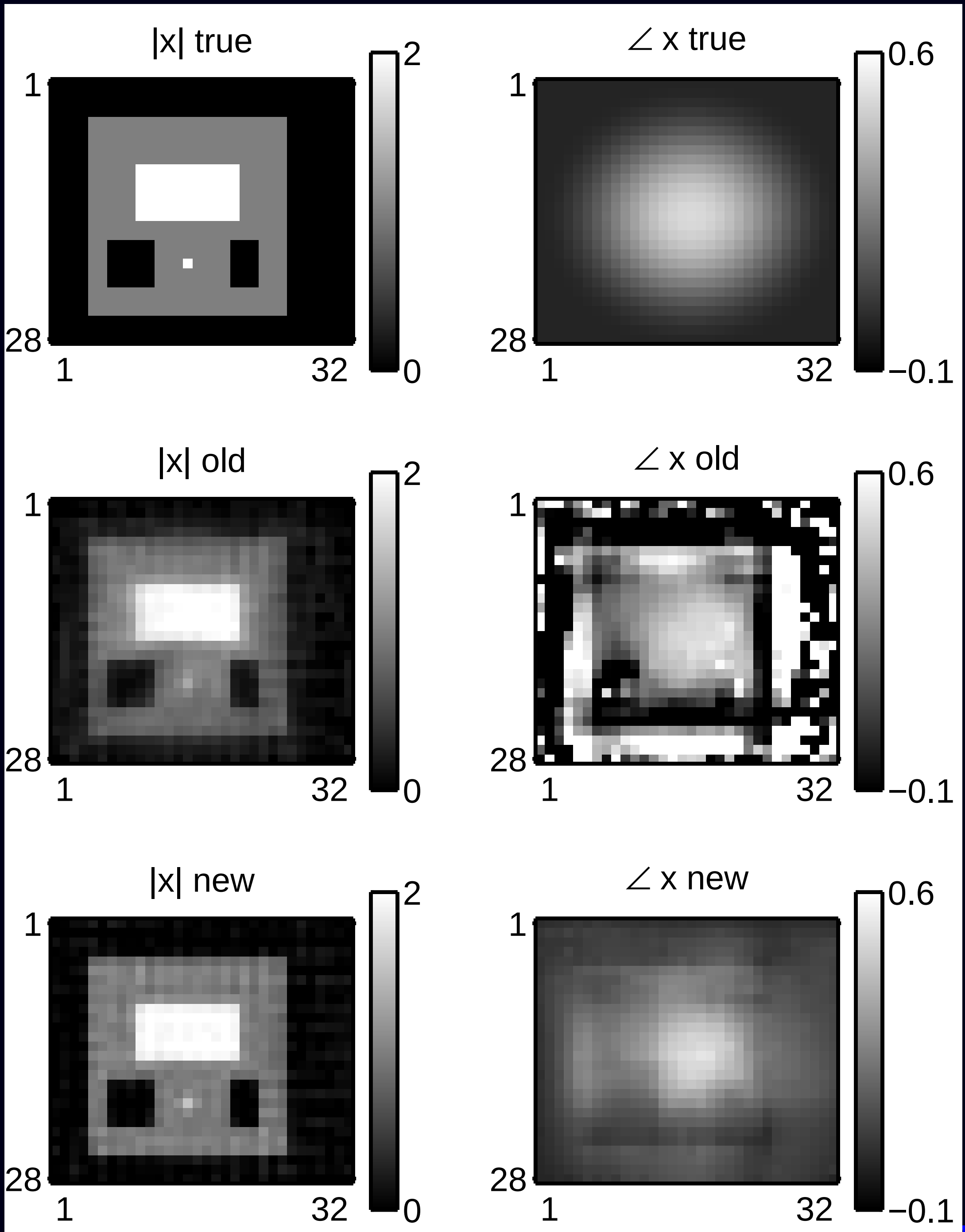
$$\boldsymbol{x}^{(n+1)} = \boldsymbol{x}^{(n)} - \boldsymbol{D}(\boldsymbol{x}^{(n)})\nabla\Phi(\boldsymbol{x}^{(n)})$$

where the diagonal matrix $\boldsymbol{D}$ has elements that ensure $\Phi$ decreases monotonically.

Alternate between magnitude and phase updates...

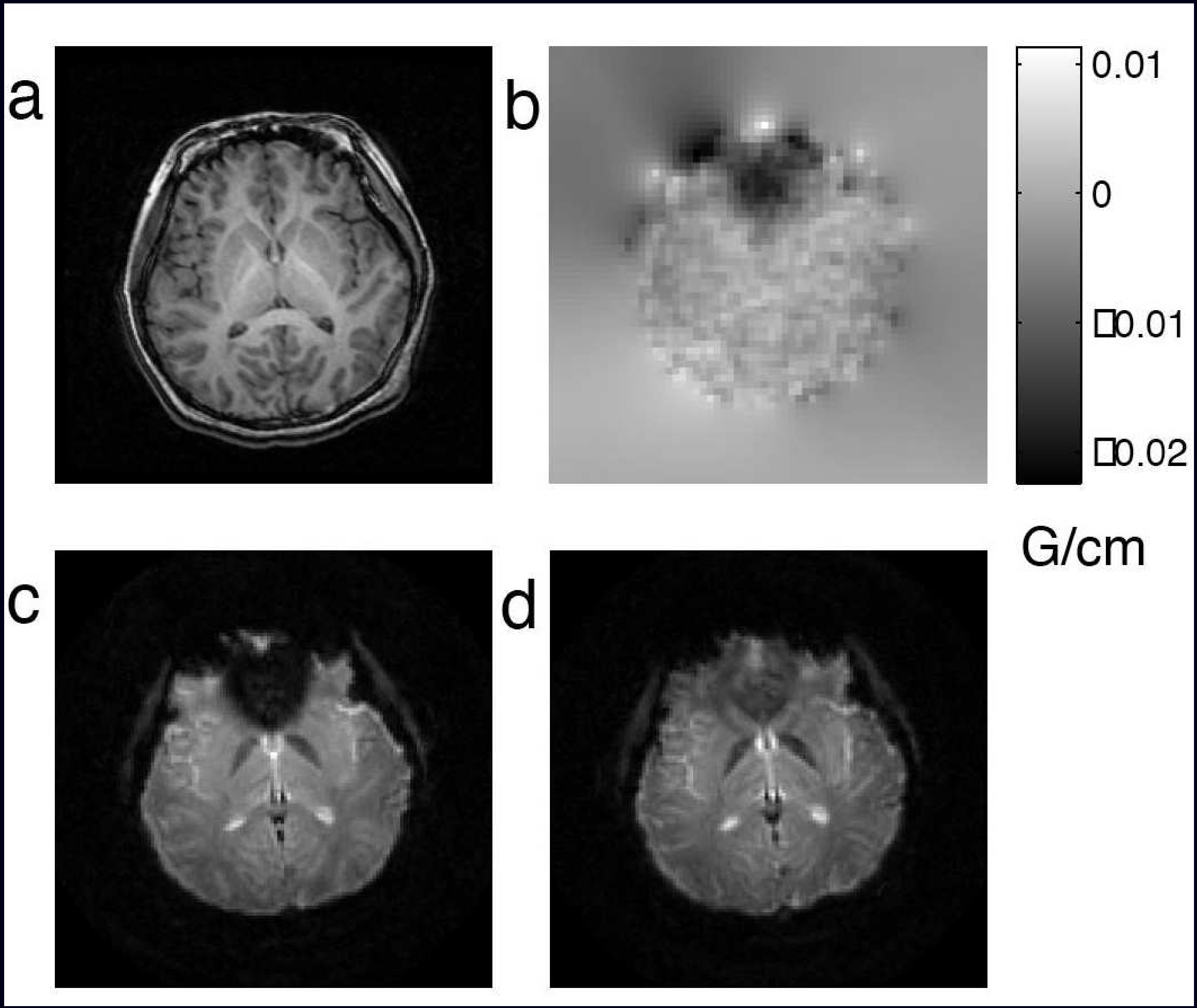
# Preliminary Simulation Example



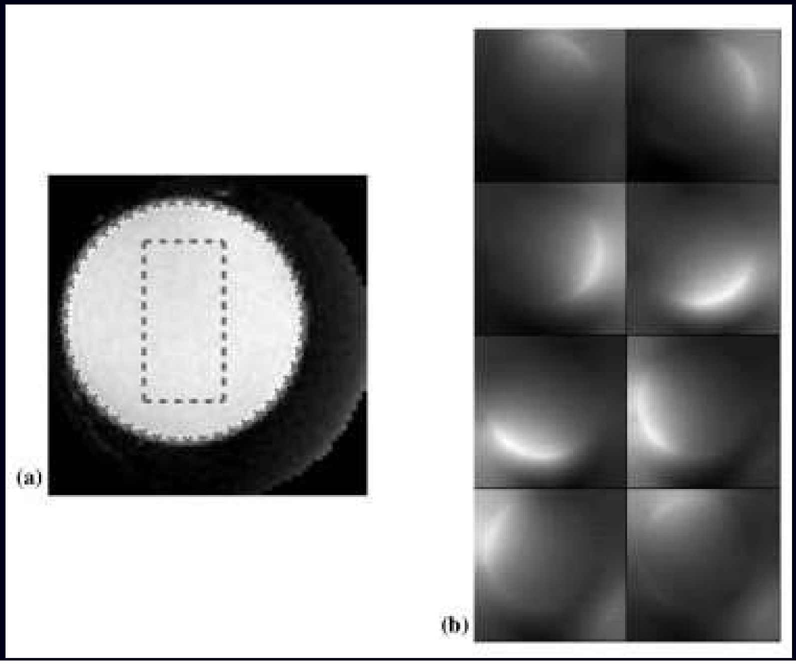



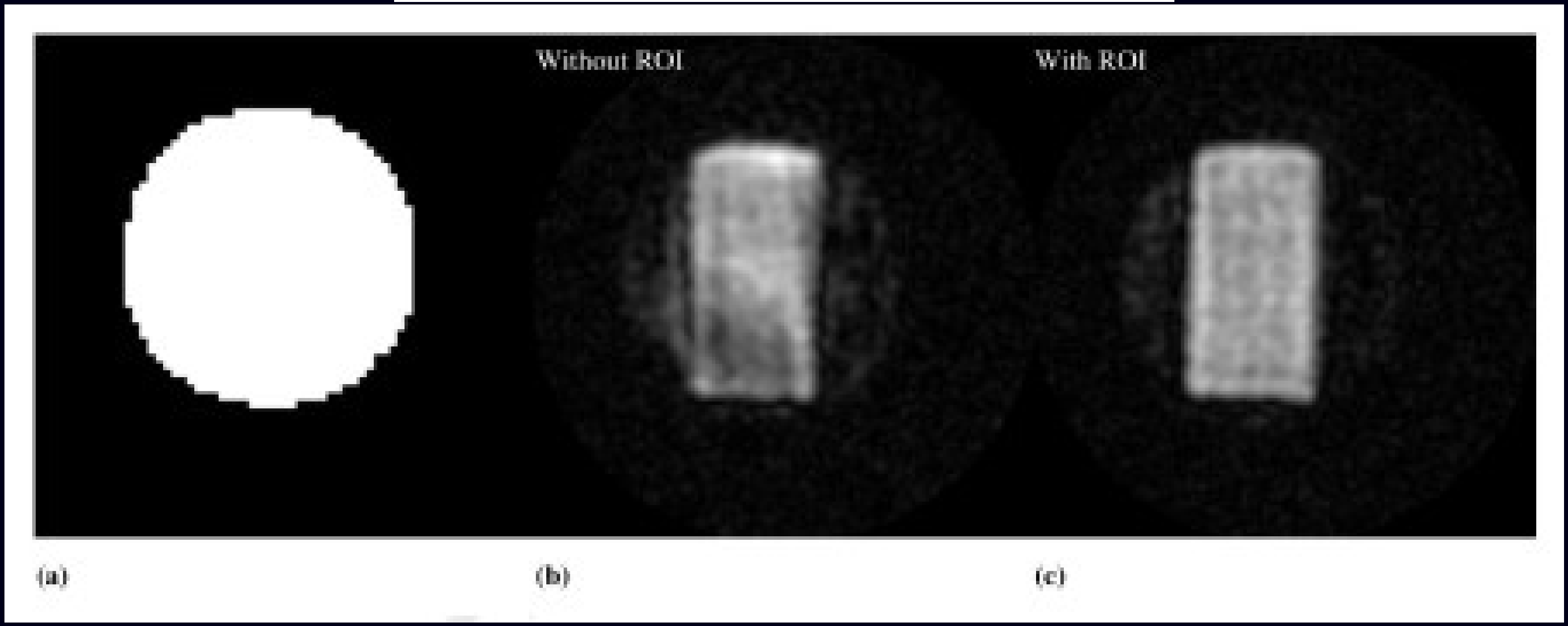
# Example: Iterative Pulse Sequence Design

(3D tailored RF pulses for through-plane dephasing compensation)



# Multiple-coil Transmit Imaging Pulses (Mc-TIP)



# Summary

- Iterative reconstruction: much potential in MRI
- Computation: reduced by tools like NUFFT / temporal interpolation;
  combined with careful optimization algorithm design
  *cf.* Shepp and Vardi, 1982, PET
- Problems involving phase terms $e^{\imath x}$ suitable for optimization transfer.

# Future work

- Multiple receive coils (SENSE)
- Through-voxel field inhomogeneity gradients
- Motion (dynamic field maps...)
- Real data...