Today

Course Overview

EE 469 B

Assignment

Projects due next Friday
Main Ideas from the Course

Interpolation

One Sample Geometry to Another

Non-Cartesian Samples $\Rightarrow$ Cartesian Grid

Non-Uniform Samples $\Rightarrow$ Uniform Samples

Filling in Missing Data: GRAPPA, SPIRIT
PARTICULARLY IMPORTANT PROBLEM

INTERPOLATION FOLLOWED BY A TRANSFORM

NON-UNIFORM DATA IN SPATIAL FREQUENCY

UNIFORM DATA IN IMAGE SPACE

GRIDDING (DENSITY CORRECTION)

UNIFORM DATA IN IMAGE SPACE

NON-UNIFORM DATA IN SPATIAL FREQUENCY

INVERSE GRIDDING (NO DENSITY CORRECTION)
**System Models**

All of the problems we looked at could be modeled as a matrix equation

\[ y = A x \]

\( A \) is called the "projection operator" in PET essentially a PET system simulator

In MRI it was called the encoding matrix:

\[ E = \begin{pmatrix}
    1 & \cdots & 1
\end{pmatrix}
\]

Each column is complex exponential produced by one voxel

For parallel MRI, coil weighting also included

For MRI, applying \( A \) (or \( E \)) is inverse gridding
Another important operator

\[ \hat{x} = A^* y \]

Adjoint operator, takes data and does some sort of reconstruction.

In PET, CT this is "Backprojection operator."

In MRI this is gridding (without density compensation).

In practice, \( A \) and \( A^* \) can't be computed and stored.

Instead, implemented as functions (m-files), contain

- Interpolations
- Transforms
- Integrations
- Convolutions
- Multiplications

All simple, fast operations.
Given

\[ y = Ax \]

where \( y \) is known, and \( A \) (and \( A^* \)) are functions.

How do we solve for \( x \)?

Apply \( A^* \) to both sides:

\[ (A^* A) x = A^* y \]

Simple Reconstruction

And solve for \( x \).

**Many Special Cases**

**2DFT in MRI**

In this case, \( A \) is Fourier Transform matrix and

\[ (A^* A) x = A^* y \]

\[ \text{Inverse FT} \]

\[ x = A^* y \]
Non-Cartesian M R I

In this case we choose to solve a different problem

\[ W y = W A x \]

\( W \) is a diagonal weighting, or preconditioning matrix, then

\[ A^* W y = (A^* W A) x \]

\[ \underbrace{(A^* W A)}_{I \text{ image}} \uparrow \quad \text{e space} \]

Choose \( W \) so that \( A^* W A = I \)

\[ x = A^* W y \]

\[ \uparrow \quad \text{original density compensation} \]

This is same as weighted least squares reconstruction
**Cartesian Sense MRI**

Here $A^*A$ has a special structure (after reordering).

$X = A^*y$

Im: MAE (data), AC: biased (reconstruction)

For acceleration $R$, each block is $R \times R$

Solve each subequation explicitly

$$x = (A^*A)^{-1} A^* y$$
Non-Cartesian Parallel MRI

Here there is no special structure to exploit iteratively solve

\[(A^*A)x = A^*y\]

Using conjugate gradient algorithm, or something else. MATLAB LSQR does this. You provide functions that implement \(A, A^*\).

You can add preconditioning (density compensation)

\[(A^*WA)x = A^*Wy\]

But you don't need to.

Same approach works for SPIRiT, PRUNO, lots of others.
PET here

\[ y = Ax \]

Event Source Distribution

\( A \) is a PET simulation, \( A^* \) is backprojection

Statistics of \( y \) are Poisson

Iterative solution that maximizes likelihood (ML-EM)

Projection and backprojection of PET

L18
All the same ideas, new applications

Non-Cartesian MRI $\Rightarrow$ Multidimensional RF pulses

Spiral k-space trajectory $+$ RF

Parallel MRI $\Rightarrow$ Parallel transmit

Take any pulse sequence and reverse it
Use density compensation as RF
Use RF as density compensation
This will do something useful
**ADDED BONUS**

Many of the problems are non-linear (rotations) but have explicit solutions!