Convergence and Non-Convergence of Algebraic Iterative Reconstruction Methods

Per Christian Hansen

DTU Compute, Technical University of Denmark

Joint work with:

Tommy Elfving – Linköping University

Michiel E. Hochstenbach, TU Eindhoven

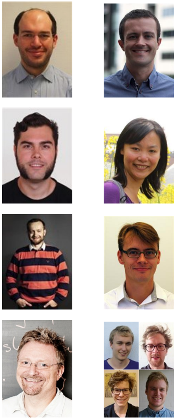
Yiqiu Dong and Nicolai A. B. Riis, DTU

DTU Compute: Section for Scientific Computing

Core areas: structural biology, PDE control, PDE-constrained optimization, computational mathematics and simulation.

Activities related to **inverse problems** and **tomography**:

- Inverse source problems and EM scattering.
- Electrical impedance tomography with hybrid data.
- Variational methods for image deblurring.
- Model error reduction in inverse problems.
- Uncertainty quantification for imaging problems.
- CT imaging with uncertain flat fields.
- Simultaneous CT reconstruction and segmentation.
- Algorithms for sparse CT reconstruction.
- Algebraic iterative methods for CT imaging.



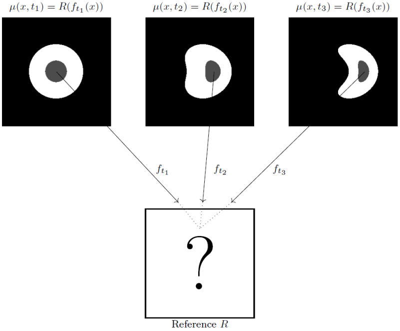
I start with examples from our ERC Advanced Grant project HD-Tomo.

Example: Dynamic CT Reconstruction

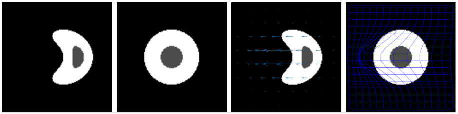
Hari Om Aggrawal, Martin S. Andersen, Jan Modersitzki (U. Lübeck)

Use a regularized “hyper-elascit” deformation model for the dynamics.

Implicit reference model



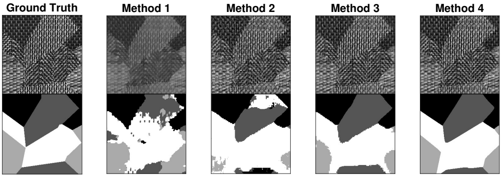
Motion is estimated from the data



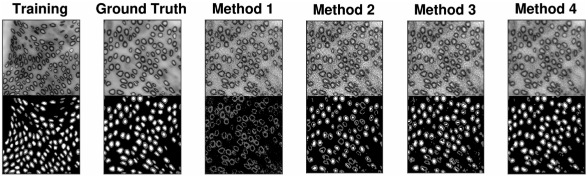
Ex.: Joint Reconstruction/Segmentation with Texture Priors

Yiqiu Dong, Hans Martin Kjer, Per Chr. Hansen **Method 4** = our method

1. Use *discriminative learning* to build a dictionary from training images.
2. Express the solution in terms of the elements of the dictionary.
3. Add total-variation regularization to the segmentation.



Synthetic data.



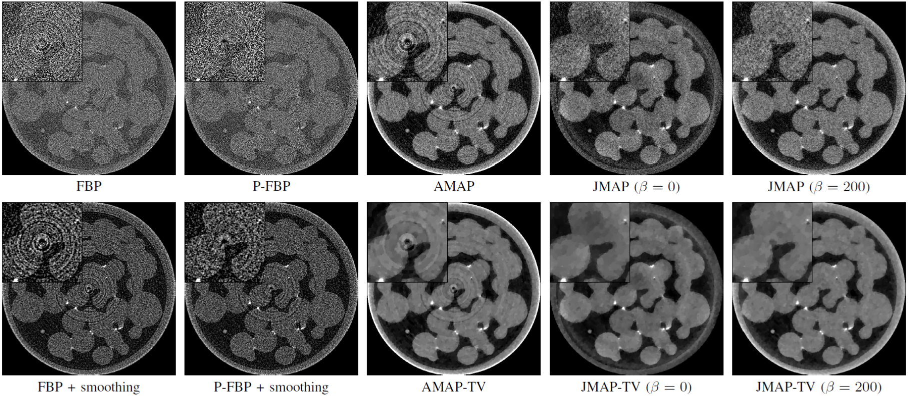
Nerve fibers.

Example: CT Imaging with Uncertain Flat Fields

Hari Om Aggrawal, Martin S. Andersen, Sean Rose, Emil Y. Sidky (U. Chicago)

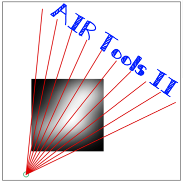
Key idea: estimate the flat field as part of the reconstruction algorithm.

JMAP = Joint MAP-estimation = our method.



Ex.: Matlab-Software – AIR Tools II

Per Chr. Hansen, Jakob S. Jørgensen (U. Manchester)



NB: not a competitor to dedicated CT packages (ASTRA, SNARK, TIGRE) but a tool for algorithm developers.
`github.com/jakobsj/AIRToolsII`

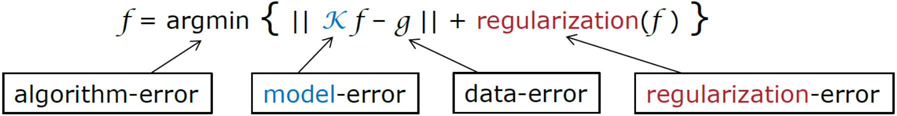
Flexible implementations of algebraic iterative reconstruction (AIR) methods for computing regularized solutions, e.g., in CT.

- Ideal for experiments with AIR methods and their convergence.
- **ART:** methods that involve one row at a time (Kaczmarz etc.).
- **CART:** column-action (or column-relaxation) methods.
- **SIRT:** methods based on matrix multiplications.
- Automatic stopping rules.
- Heuristics for the relaxation parameter.
- Test problems from tomography.

VILLUM Investigator research grant, starting fall 2019

Uncertainty Quantification (UQ)

All kinds of errors have influence on the solution:



UQ is the end-to-end study of the impact of all forms of error and uncertainty in the data and models arising in the application.

We develop the mathematical, statistical and computational framework for applying UQ to inverse problems such as deconvolution, image deblurring, tomographic imaging, source reconstruction, and fault inspection.

The goal is to create a computational platform, for suited for non-experts.

And Now: Algebraic Iterative Methods

Algebraic representation of the reconstruction problem:

$$A\,x = b, \qquad x = \text{object}, \qquad b = \text{sinogram}.$$

Multiplication with A $\leftrightsquigarrow$ projector.

Multiplication with A^T $\leftrightsquigarrow$ backprojector.

Scenarios:

1. Slow algorithm that solves $A\,x = b$.
2. Fast algorithm that doesn't solve $A\,x = b$, but solves related problem.
3. Fast algorithm that does not solve anything.

Quiz: which is the most popular?

Unmatched Projector/Backprojector Pairs

Linear algebra and optimization: the backprojector A^T is the transposed of the projector A. *Otherwise the theory and the algorithms do not work.*

Many (most?) software packages implement the backprojector in such a way that it is **not** the exact transposed of the projector.

- Why: depends on the application and the traditions.
- Philosophy: the computations approximate the physics $\rightarrow$ different approaches for A and A^T.
- Practicality/HPC: the code must make optimal use of multi-core processors, GPUs and other hardware accelerators.

This work:

Study the influence of unmatched projector/backprojetor pairs on the computed solutions and the convergence of the iterations.

Convergence Analysis for Unmatched Pairs

To study many different cases we consider the generic **BA Iteration**

$$x^{k+1} = x^k + \omega\, B\,(b - A\,x^k)\ , \qquad \omega > 0.$$

- Any fixed point x^* satisfies $BA\,x^* = B\,b$.
- If $B\,A$ is invertible then $x^* = (BA)^{-1}B\,b$.
- If $B = A^T$ then $x^* =$ least squares solution (steepest descent alg.).

Shi, Wei, Zhang (2011); Elfving, Hansen (2018)

The **BA Iteration** converges to a solution of $BA\,x = B\,b$ if and only if

$$0 < \omega < \frac{2\,\Re(\lambda_j)}{|\lambda_j|^2} \quad \text{and} \quad \boxed{\Re(\lambda_j) > 0} \qquad \{\lambda_j\} = \texttt{eig}(BA)\ .$$

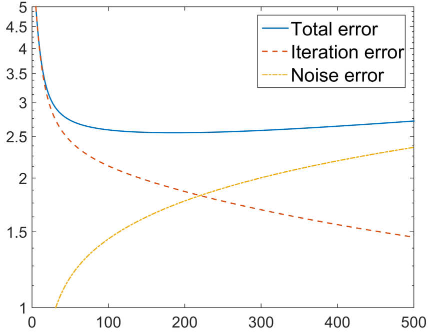
Zeng & Gullberg (2000): similar analysis but ignoring complex λ_j.

Convergence Analysis: Split the Error

Let $\bar{x}^k$ denote the iterates for a noise-free right-hand side. We consider:

$$\underbrace{x^k - \bar{x}}_{\text{total error}} = \underbrace{x^k - \bar{x}^k}_{\text{noise error}} + \underbrace{\bar{x}^k - \bar{x}}_{\text{iteration error}}$$

Semi-convergence: the iteration error decreases & the noise error increases. The total error has a minimum.

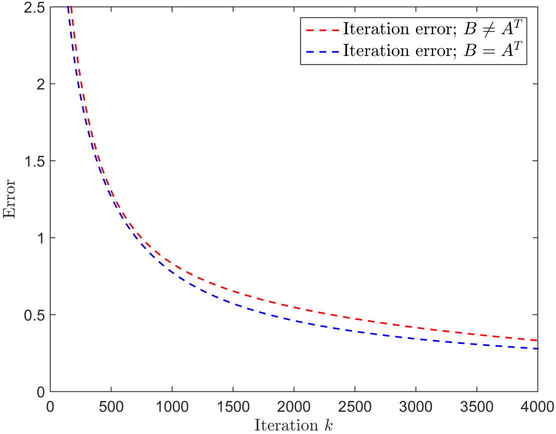


Cimmino's method.

Test problem

- ▷ 64 × 64 phantom
- ▷ 180 projections at
- ▷ $1^\circ, 2^\circ, 3^\circ, \ldots, 180^\circ$
- ▷ $m = 16\,380$
- ▷ $n = 4\,096$

$$\Re(\lambda_j) > 0$$



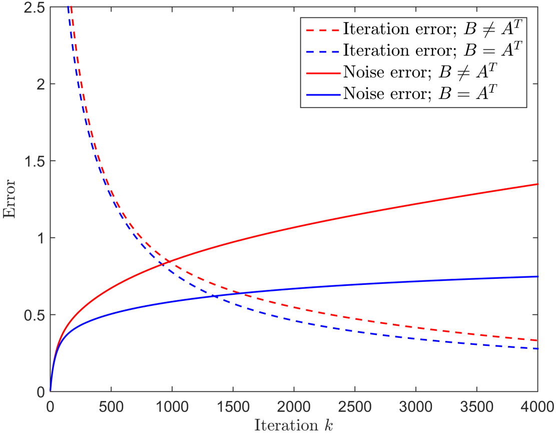
Iteration error: both versions converge to $\bar{x}$; the one with $B \neq A^T$ is slower.

Cimmino's method.

Test problem

- ▷ 64×64 phantom
- ▷ 180 projections at
- ▷ $1^\circ, 2^\circ, 3^\circ, \ldots, 180^\circ$
- ▷ $m = 16\,380$
- ▷ $n = 4\,096$

$$\Re(\lambda_j) > 0$$



Iteration error: both versions converge to $\bar{x}$; the one with $B \neq A^T$ is slower.

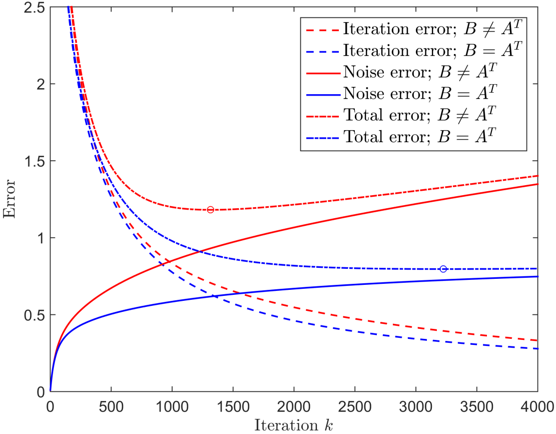
Noise error: the one for $B \neq A^T$ increases faster.

Cimmino's method.

Test problem

- ▷ 64×64 phantom
- ▷ 180 projections at
- ▷ $1^\circ, 2^\circ, 3^\circ, \ldots, 180^\circ$
- ▷ $m = 16\,380$
- ▷ $n = 4\,096$

$$\Re(\lambda_j) > 0$$



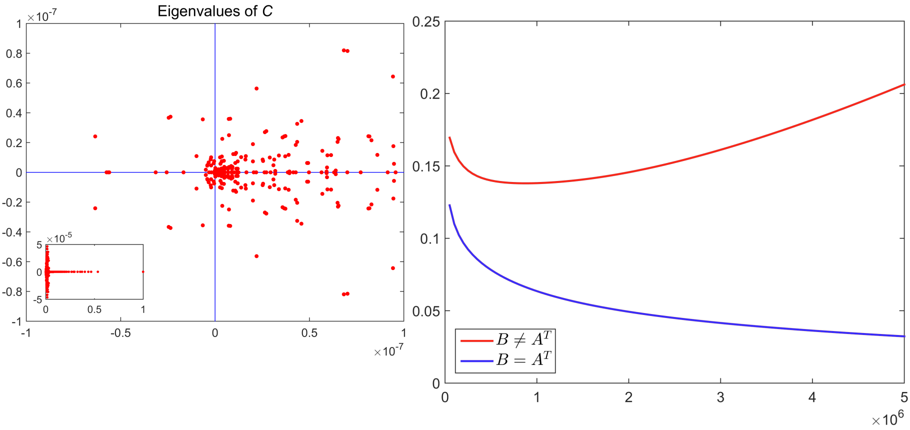
Iteration error: both versions converge to $\bar{x}$; the one with $B \neq A^T$ is slower.

Noise error: the one for $B \neq A^T$ increases faster.

Total error: semi-convergence, the iteration with $B \neq A^T$ reaches the min. error ○ 1.181 after 1314 iterations. This error is 48% larger than the min. error ○ 0.796 for the iterations with A^T, achieved after 3225 iterations.

And Now: Negative Real Parts → No Convergence

Parallel-beam CT, **unmatched** pair from *package X*, 64 × 64 Shepp-Logan phantom, 90 projections, 60 detector pixels, $\min \Re(\lambda_j) = -6.4 \cdot 10^{-8} < 0$.



The Shifted BA Iteration

We introduce a modified algorithm that has *guaranteed convergence*, and whose fixed point *approximates* the exact solution $\bar{x}$.

The **shifted BA iteration** takes the form:

$$x^{k+1} = (1 - \alpha\,\omega)\,x^k + \omega\,B\,(b - A\,x^k)\;, \qquad k = 0, 1, 2, \ldots$$

Small overhead associated with the shift $(1 - \alpha\,\omega)\,x^k$ (`xSCAL` operation).

Dong, Hansen, Hochstenbach, Riis (2018)

Let λ_j denote eigenvalues of BA different from $-\alpha$. Then the shifted BA iteration converges to a fixed point if and only if α and ω satisfy

$$0 < \omega < 2\,\frac{\Re(\lambda_j) + \alpha}{|\lambda_j|^2 + \alpha\,(\alpha + 2\,\Re(\lambda_j))} \qquad \text{and} \qquad \boxed{\Re(\lambda_j) + \alpha > 0}$$

The Approximation Error

The fixed point x_α^* to the shifted BA iteration is an approximation to the exact solution. How good is it?

Dong, Hansen, Hochstenbach, Riis (2018)

Assume that $B\,A + \alpha\,I$ is nonsingular. The fixed point x_α^* satisfies

$$x_\alpha^* = (B\,A + \alpha\,I)^{-1} B\,b = B\,(A\,B + \alpha\,I)^{-1} b\ , \qquad x_\alpha^* \in \mathcal{R}(B)\ .$$

For noise-free data $\bar{b} = A\,\bar{x}$ the fixed point satisfies

$$x_\alpha^* = (B\,A + \alpha\,I)^{-1} B\,A\,\bar{x}\ , \qquad x_\alpha^* \in \mathcal{R}(BA)\ ,$$

and the *approximation error* is

$$x - \bar{x}_\alpha^* = \alpha\,(B\,A + \alpha\,I)^{-1} \bar{x}\ .$$

In practice α is small, so we are good.

How to Compute/Estimate the Shift

Key idea: introduce a small α, just large enough to ensure that all the shifted eigenvalues have a positive real part, i.e., $\Re(\lambda_j) + \alpha > 0$.

We need to estimate the leftmost eigenvalue λ_{lm} of BA.

Methods based on Krylov subspaces are well suited:

- The **Krylov–Schur** method is similar to Matlab's `eigs`, but is a factor 1.2–1.3 faster.
- The **Jacobi–Davidson** method is worthwhile when λ_{lm} is not well separated from neighboring eigenvalues. This is not the case for CT, and Jacobi–Davidson performs worse than Krylov–Schur.
- The **field-of-values** method (a new idea by us) is faster than Krylov–Schur if we can accept a low-precision λ_{lm}.

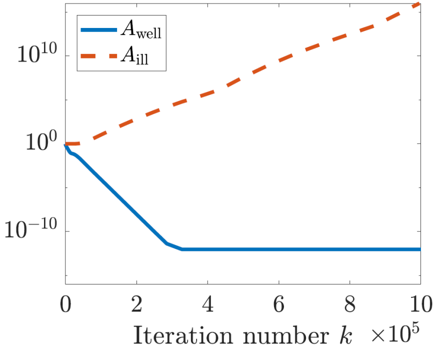
The overhead is estimating λ_{lm} is typically equivalent to a few hundred iterations. Acceptable if the same system matrix A is used several times.

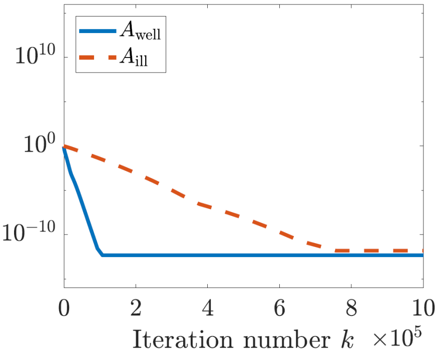
Performance

Tiny 64×64 test problem (`regutm` from Reg. Tools).

Reconstruction errors for A_{well} ($\lambda_{\text{lm}} > 0$) and A_{well} ($\lambda_{\text{lm}} < 0$).

- Left: BA iteration.
- Right: shifted BA iteration with $\alpha = 2\,\lambda_{\text{lm}}$.



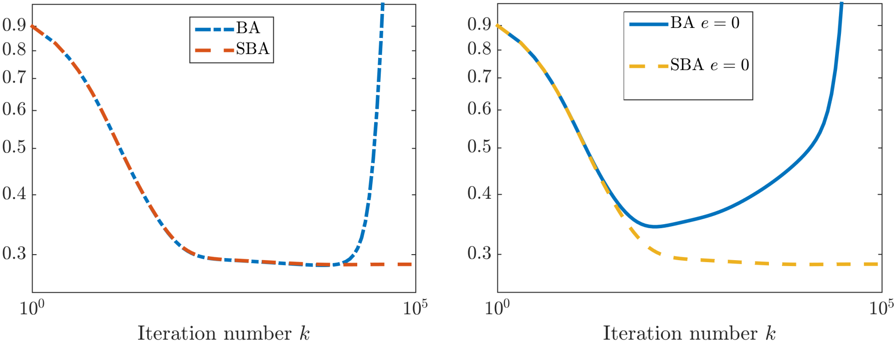


Performance

Realistic test problem from GPU-ASTRA $\rightarrow$ same behavior.

BA Iteration and Shifted BA Iteration with $\alpha = 2\,\lambda_{\mathrm{lm}}$.

- Left: convergence to slightly perturbed solution $\bar{x}^*_\alpha$.
- Right: convergece to ground truth (exact) solution $\bar{x}$.



Conclusion

- We studied the influence of an unmatched backprojector $B \neq A^T$.
  - Convergence if all eigenvalues of BA have positive real parts.
  - Non-convergence if leftmost eigenvalue of BA has a negative real part.
- We introduced the **shifted BA iteration** and proved that it converges to a slightly perturbed solution.
- We use efficient algorithms to estimate the necessary shift.
- We showed a few numerical results that illustrate the performance.
- Matlab code is available from us.

The overhead of the complete algorithm, with guaranteed convergence, may be acceptable if we use the same system matrix A several times.