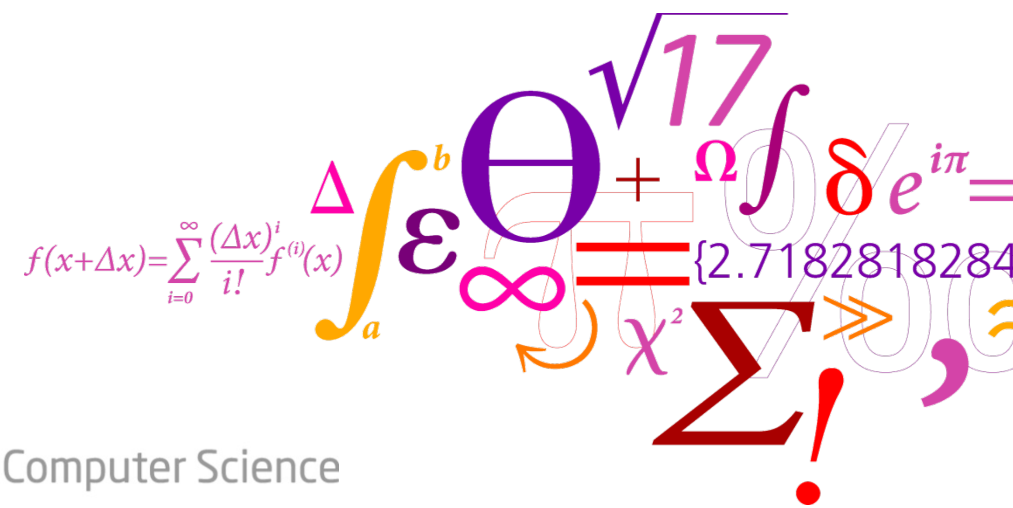
Regularizing Iterations with Krylov Subspace Methods

Per Christian Hansen
Technical University of Denmark

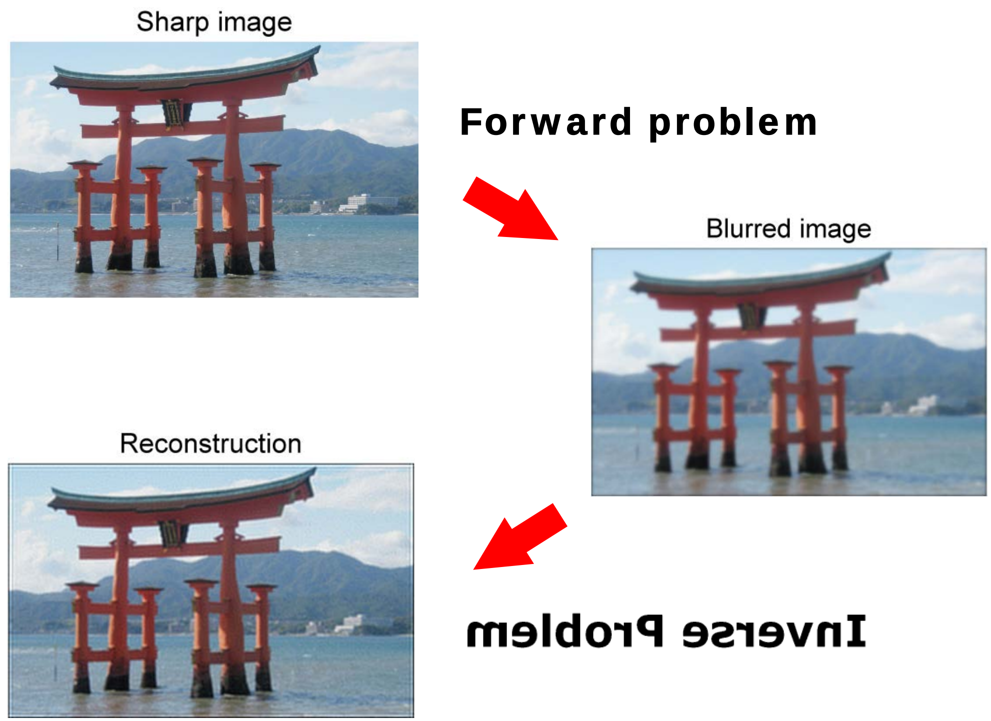
HDtomo
HIGH - DEFINITION TOMOGRAPHY

GPUlab
DTU Informatics



DTU Compute
Department of Applied Mathematics and Computer Science

Image Problems



Discretization yields a LARGE system of linear equations: $A\,x = b$.

Two important aspects related to this system:

- Use the right boundary conditions.
- The matrix A is very ill conditioned $\rightarrow$ $\boxed{\text{Do not solve } A\,x = b\,!}$

A Systematic View of Regularization

We must apply regularization in order to deal with the ill conditioning of the problem and suppress the influence of the noise in the data.

Tikhonov regularization:

$$\min_x \left\{ \|A\,x - b\|_2^2 + \lambda^2\, \|L\,x\|_2^2 \right\}$$

The choice of smoothing norm, together with the choice of λ, forces x to be effectively dominated by components in a low-dimensional subspace, determined by the GSVD of (A, L) – or the SVD of A if $L = I$.

Regularization by projection:

$$\min_x \|A\,x - b\|_2 \quad \text{subject to} \quad x \in \mathcal{W}_k$$

where $\mathcal{W}_k$ is a k-dimensional subspace.

This works well if "most of" x^{exact} lies in a low-dimensional subspace; hence $\mathcal{W}_k$ must be spanned by desirable basis vectors. Think of Truncated SVD: $\mathcal{W}_k = \text{span}\{v_1, v_2, \ldots, v_k\}$, v_i = right singular vectors.

The Projection Method

A more practical formulation of regularization by projection.

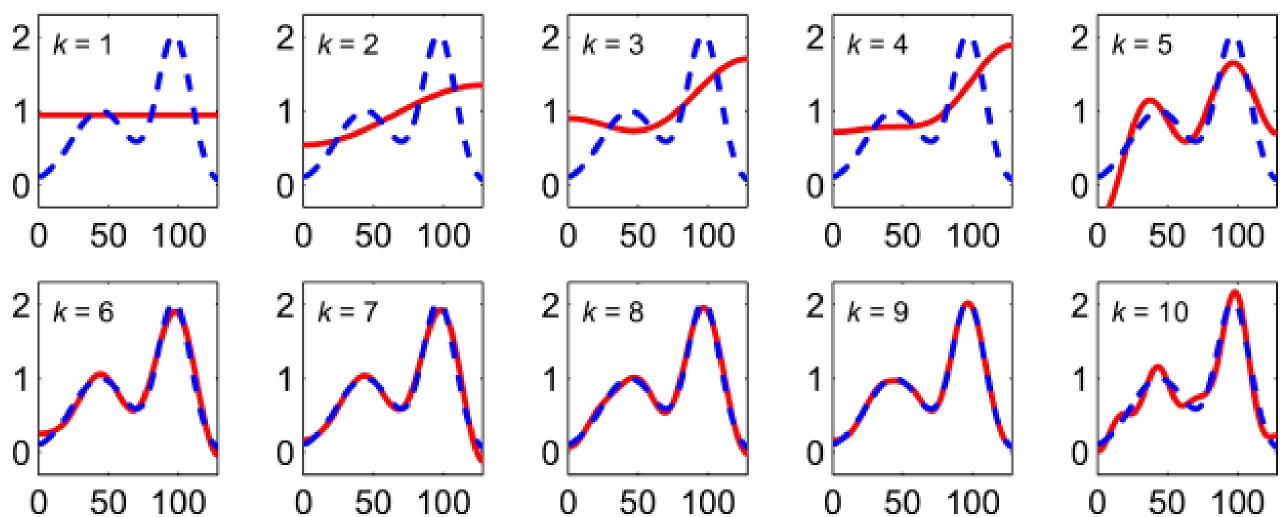
We are given the matrix $W_k = (w_1, \ldots, w_k) \in \mathbb{R}^{n \times k}$ such that $\mathcal{W}_k = \mathcal{R}(W_k)$.

We can write the requirement as $x = W_k\, y$, leading to the formulation

$$x^{(k)} = W_k\, y^{(k)}, \qquad y^{(k)} = \mathrm{argmin}_y \|(A\, W_k)\, y - b\|_2 .$$

Projected problem

Example:
DCT basis



Operations often do not require W_k explicitly.

Some Thought on the Basis Vectors

The DCT basis – and similar bases that define fast transforms:

- computationally convenient (fast) to work with, but
- may not be well suited for the particular problem.

The SVD basis – or GSVD basis if $L \neq I$ – gives an "optimal" basis for representation of the matrix A, but …

- it is computationally expensive (slow), and
- it does not involve information about the right-hand side b.

Is there a basis that is computationally attractive and also involves information about both A and b, and thus the given problem?

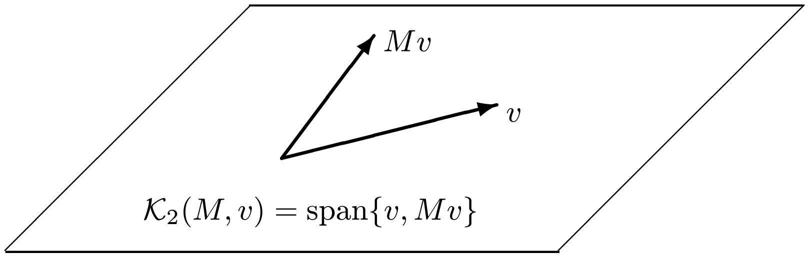
$\rightarrow$ Krylov subspaces!

Krylov Subspaces

Given a square matrix M and a vector v, the associated Krylov subspace is defined by

$$\mathcal{K}_k(M, v) \equiv \text{span}\{v, Mv, M^2 v, \ldots, M^{k-1} v\}, \qquad k = 1, 2, \ldots$$

with $\dim(\mathcal{K}_k(M, v)) \leq k$.



Krylov subspaces have many important applications in scientific computing:

- solving large systems of linear equations,
- computing eigenvalues,
- solving algebraic Riccati equations, and
- determining controllability in a control system.

They are also important tools for regularization of large-scale discretizations of inverse problems, which is the topic of this talk.

More about the Krylov Subspace

The Krylov subspace, defined as

$$\mathcal{K}_k \equiv \mathrm{span}\{A^T b, A^T A\, A^T b, (A^T A)^2 A^T b, \ldots, (A^T A)^{k-1} A^T b\},$$

always *adapts* itself to the problem at hand! But the "naive" basis,

$$p_i = (A^T A)^{i-1} A^T b \,/\, \|(A^T A)^{i-1} A^T b\|_2, \qquad i = 1, 2, \ldots$$

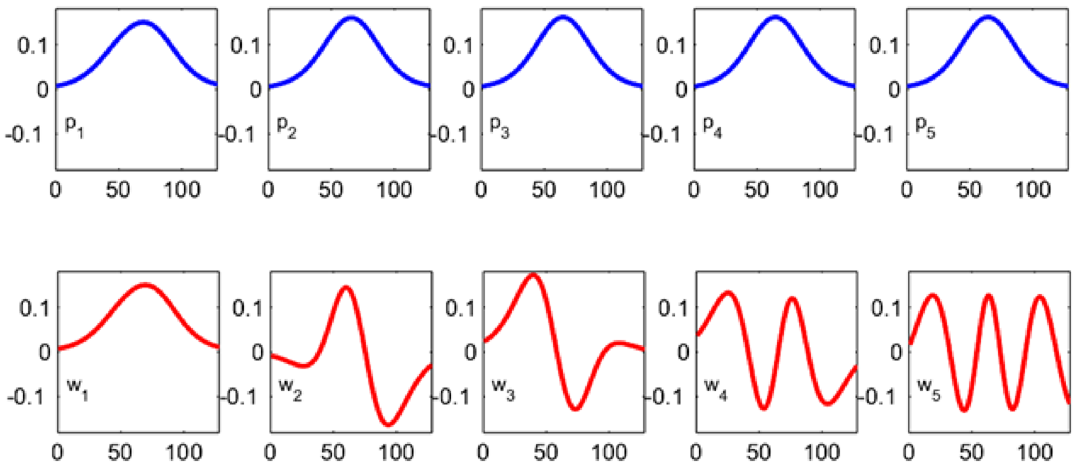
are NOT useful: $p_i \to v_1$ as $i \to \infty$.

Can use modified Gram-Schmidt:

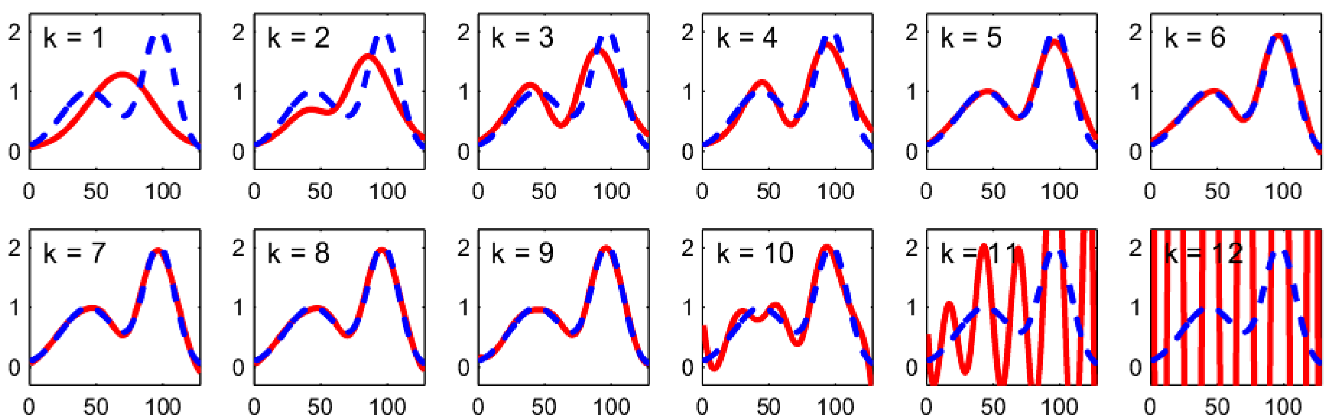
$$\begin{array}{lll}
w_1 \leftarrow A^T b; & w_1 \leftarrow w_1/\|w_1\|_2 & \\
w_2 \leftarrow A^T A\, w_2; & w_2 \leftarrow w_2 - w_1^T w_2\, w_1; & w_2 \leftarrow w_2/\|w_2\|_2 \\
w_3 \leftarrow A^T A\, w_2; & w_3 \leftarrow w_3 - w_1^T w_3\, w_1; & \\
 & w_3 \leftarrow w_3 - w_2^T w_3\, w_2; & w_3 \leftarrow w_3/\|w_3\|_2
\end{array}$$

The Krylov Subspace – Example

Normalized basis vectors p_i (blue) and orthonormal basis w_i (red).



Same example as before: Krylov subspace solutions



Regularizing Iterations

Can we compute $x^{(k)}$ without forming and storing the Krylov basis in W_k?

Apply CG to the normal equations for the least squares problem

$$\min \|A\,x - b\|_2 \qquad \Leftrightarrow \qquad A^T A\,x = A^T b \ .$$

This stable stable and efficient implementation of this algorithm is called CGLS, and it produces a sequence of iterates $x^{(k)}$ which solve

$$\min \|A\,x - b\|_2 \qquad \text{subject to} \qquad x \in \mathcal{K}_k \ .$$

This use of CGLS to compute regularized solutions in the Krylov subspace $\mathcal{K}_k$ is referred to as *regularizing iterations*.

Iterative methods are based on multiplications with A and A^T (blurring).

How come repeated blurings can lead to reconstruction?

$\rightarrow$ CGLS constructs a polynomial approximation to $A^\dagger = (A^T A)^{-1} A^T$.

The CGLS Algorithm

$$
\left.\begin{aligned}
x^{(0)} &= \text{starting vector (e.g., zero)} \\
r^{(0)} &= b - A\,x^{(0)} \\
d^{(0)} &= A^T r^{(0)}
\end{aligned}\right\}\ \text{Initialization}
$$

$$
\begin{aligned}
&\text{for } k = 1, 2, \ldots \\
&\qquad \bar\alpha_k = \|A^T r^{(k-1)}\|_2^2 / \|\underbrace{A\,d^{(k-1)}}_{\text{Mult. with } A}\|_2^2 \\
&\qquad x^{(k)} = x^{(k-1)} + \bar\alpha_k\, d^{(k-1)} \\
&\qquad r^{(k)} = r^{(k-1)} - \bar\alpha_k\, A\,d^{(k-1)} \\
&\qquad \bar\beta_k = \|A^T r^{(k)}\|_2^2 / \|A^T r^{(k-1)}\|_2^2 \\
&\qquad d^{(k)} = \underbrace{A^T r^{(k)}}_{\text{Mult. with } A^{\mathrm{T}}} + \bar\beta_k\, d^{(k-1)} \\
&\text{end}
\end{aligned}
$$

The CGLS Polynomials

CGLS implicitly constructs a polynomial $\mathcal{P}_k$ such that

$$x^{(k)} = \mathcal{P}_k(A^T A)A^T b \ .$$

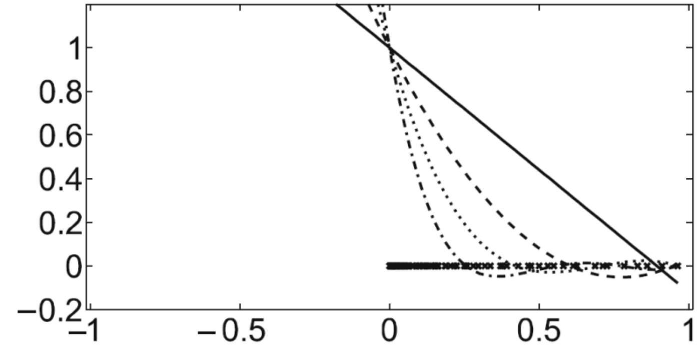
But how is $\mathcal{P}_k$ constructed? Consider the residual

$$\begin{aligned} r^{(k)} &= b - A\,x^{(k)} \;=\; \left(I - A\,\mathcal{P}_k(A^T A)\,A^T\right) b \\ \|r^{(k)}\|_2^2 &= \left\|\left(I - \Sigma\,\mathcal{P}_k(\Sigma^2)\Sigma\right) U^T b\right\|_2^2 \\ &= \sum_{i=1}^{n} \left(1 - \sigma_i^2 \mathcal{P}_k(\sigma_i^2)\right)^2 (u_i^T b)^2 \;=\; \sum_{i=1}^{n} \mathcal{Q}_k(\sigma_i^2)(u_i^T b)^2 \end{aligned}$$

To minimize residual norm $\|r^{(k)}\|_2$:

$\rightarrow$ make $\mathcal{Q}_k(\sigma_i^2)$ small where $(u_i^T b)^2$ is large

$\rightarrow$ force $\mathcal{Q}_k(\sigma_i^2)$ to have roots
near σ_i that corresp. to large $(u_i^T b)^2$.



Semi-Convergence

During the first iterations, the Krylov subspace $\mathcal{K}_k$ captures the "important" information in the noisy right-hand side b.

- In this phase, the CGLS iterate $x^{(k)}$ approaches the exact solution.

At later stages, the Krylov subspace $\mathcal{K}_k$ starts to capture undesired noise components in b.

- Now the CGLS iterate $x^{(k)}$ diverges from the exact solution and approach the undesired solution $A^\dagger b$ to the least squares problem.

The iteration number k (= the dimension of the Krylov subspace $\mathcal{K}_k$) plays the role of the regularization parameter.

This behavior is called *semi-convergence*.

Illustration of Semi-Convergence

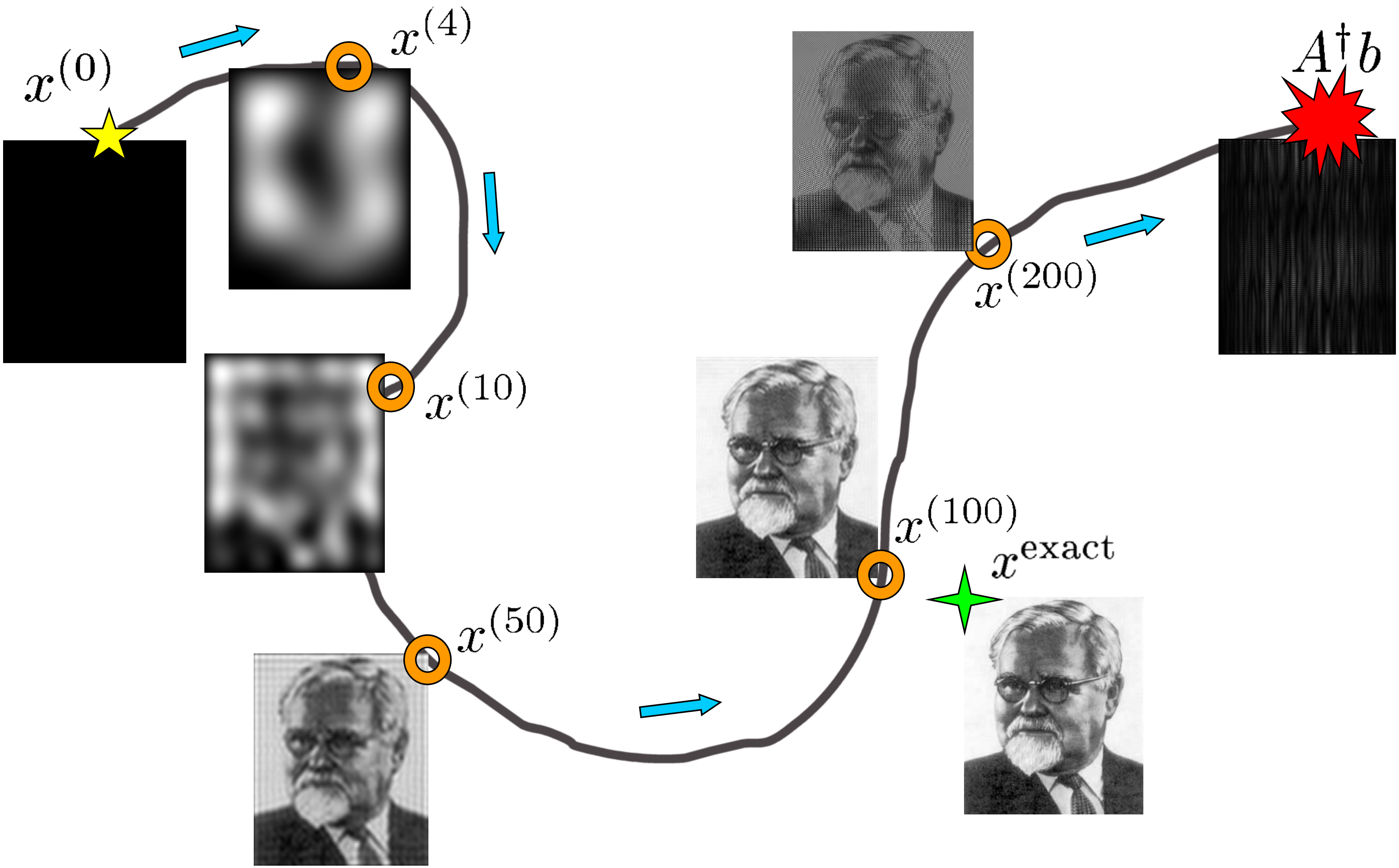
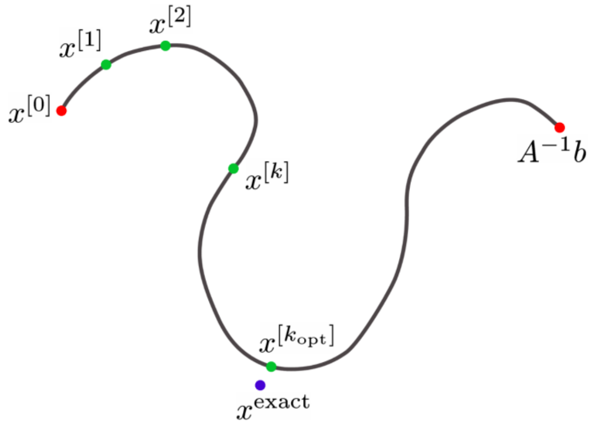
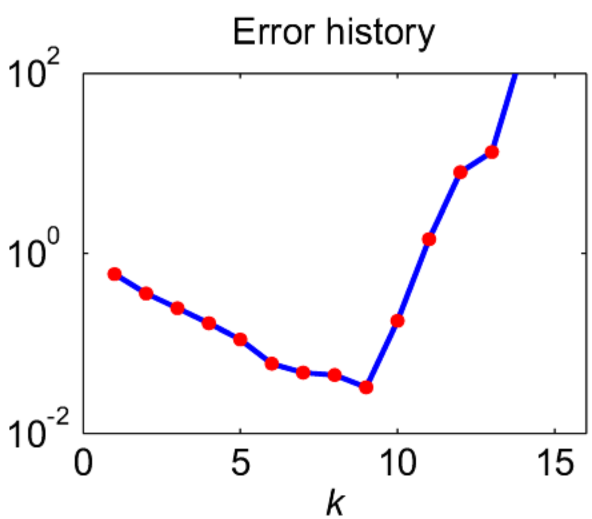


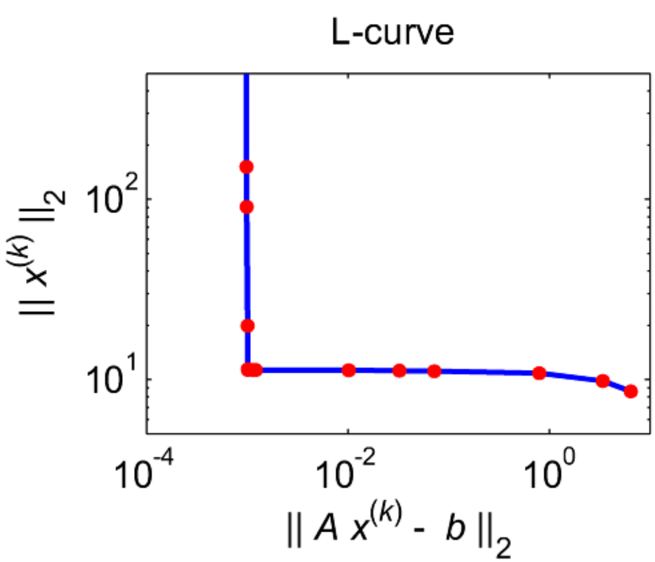
Illustration of Semi-Convergence

Recall this illustration:



The "ideal" behavior of the error $\| x^{(k)} - x^{\mathrm{exact}} \|_2$ and the associated L-curve:





Yet Another Krylov Subspace Method

An ongoing project in HD-Tomo right now (with Yiqiu & Henrik).

If certain components (or features) are missing from the Krylov subspace, then it makes good sense to *augment* the subspace with these components.

Augmented RRGMRES does precisely that:

$$\vec{\mathcal{S}}_k = \mathrm{span}\{w_1, \ldots, w_p\} + \mathrm{span}\{Ab, A^2 b, A^3 b, \ldots, A^k b\}.$$

Example: deriv2.

All vectors in the Krylov subspace $\rightarrow$ 0 at the ends.

$w_1 = (1,1,\ldots,1)^T$
$w_2 = (1,2,\ldots,n)^T$