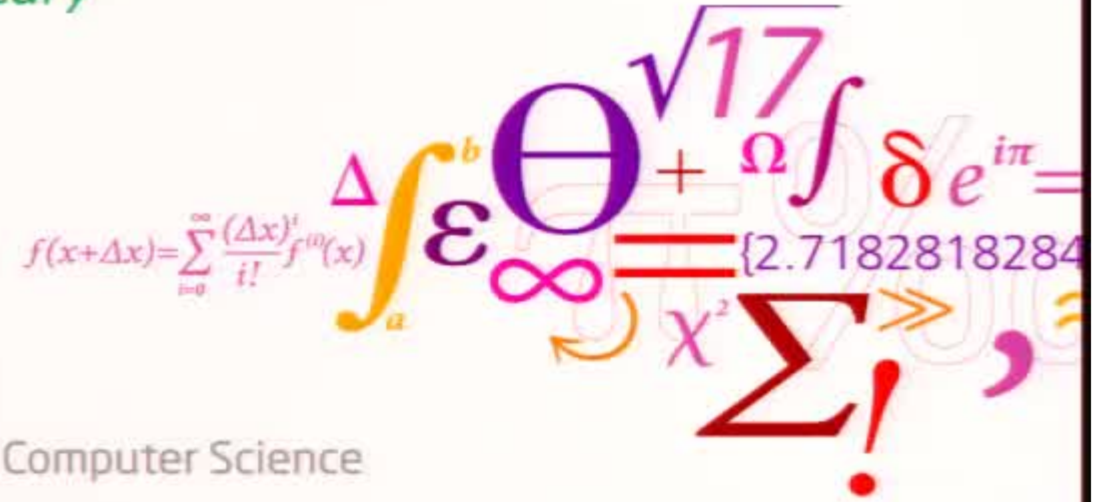
Lanczos Bidiagonalization with Subspace Augmentation for Discrete Inverse Problems

Per Christian Hansen Technical University of Denmark

Ongoing work with Kuniyoshi Abe, Gifu

Dedicated to Dianne P. O'Leary



HIGH - DEFINITION TOMOGRAPHY

DTU Compute
Department of Applied Mathematics and Computer Science

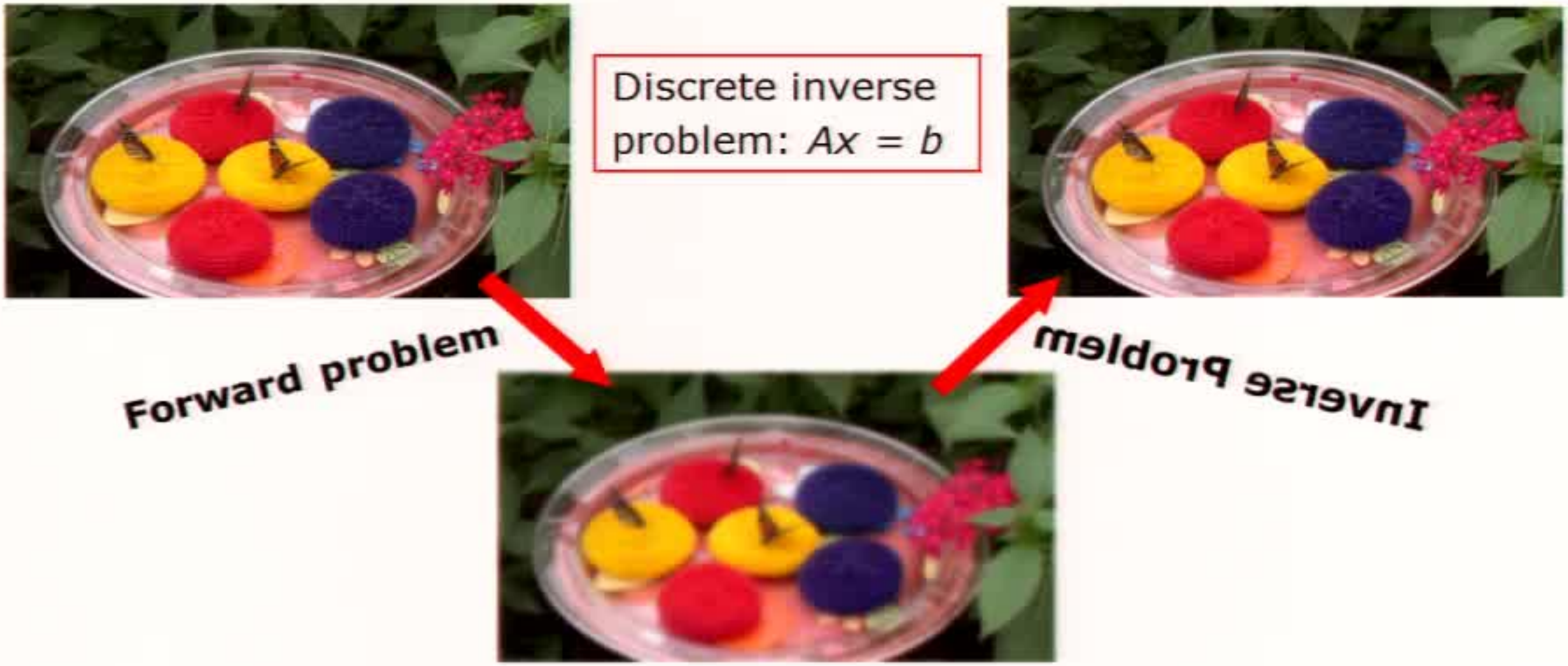
Many Many Thanks to Dianne for Inspiration, Insight, and Very Nice Collaborations ☺

Dianne, Jim and I wrote a book and tried it on students in Bari.

It is water!

Pictures by
Nicola Mastronardi

Overview of Talk



Discrete inverse problem: $Ax = b$

1. Iterative Krylov-subspace methods – regularizing iterations.
2. Augmenting the Krylov subspace for improved solutions.
3. Lanczos bidiagonalization algorithm with augmented subspace.
4. Numerical examples that illustrate the advantage of this idea.

Regularization Algorithms

Variational formulations take the form

$$\min_x \left\{ \|A\,x - b\|_2^2 + \lambda\,\mathcal{R}(x) \right\}$$

where $\mathcal{R}(x)$ is a regularization terms that penalizes unwanted features in the solution, and λ is a user-chosen regularization parameter.

H & O'Leary 1993, O'Leary 2001; Rust & O'Leary 2008 – choosing λ.

Projection formulations take the form

$$\min_x \|A\,x - b\|_2^2 \qquad \text{s.t.} \quad x \in \mathcal{S}_k \ ,$$

where the "signal subspace" $\mathcal{S}_k$ is a linear subspace of dimension k.

If $\mathcal{S}_k$ is chosen such that it captures the main features in the solution, then this approach is well suited for large-scale problems.

Hybrid methods that apply regularization to the projected problem.

Chung, Nagy & O'Leary 2008 – hybrid method with GCV.

The Signal Subspace

In some applications we can use a *pre-determined subspace*, e.g., spanned by the Fourier basis, the discrete cosine bases, a wavelet basis, etc.

An example: truncated SVD

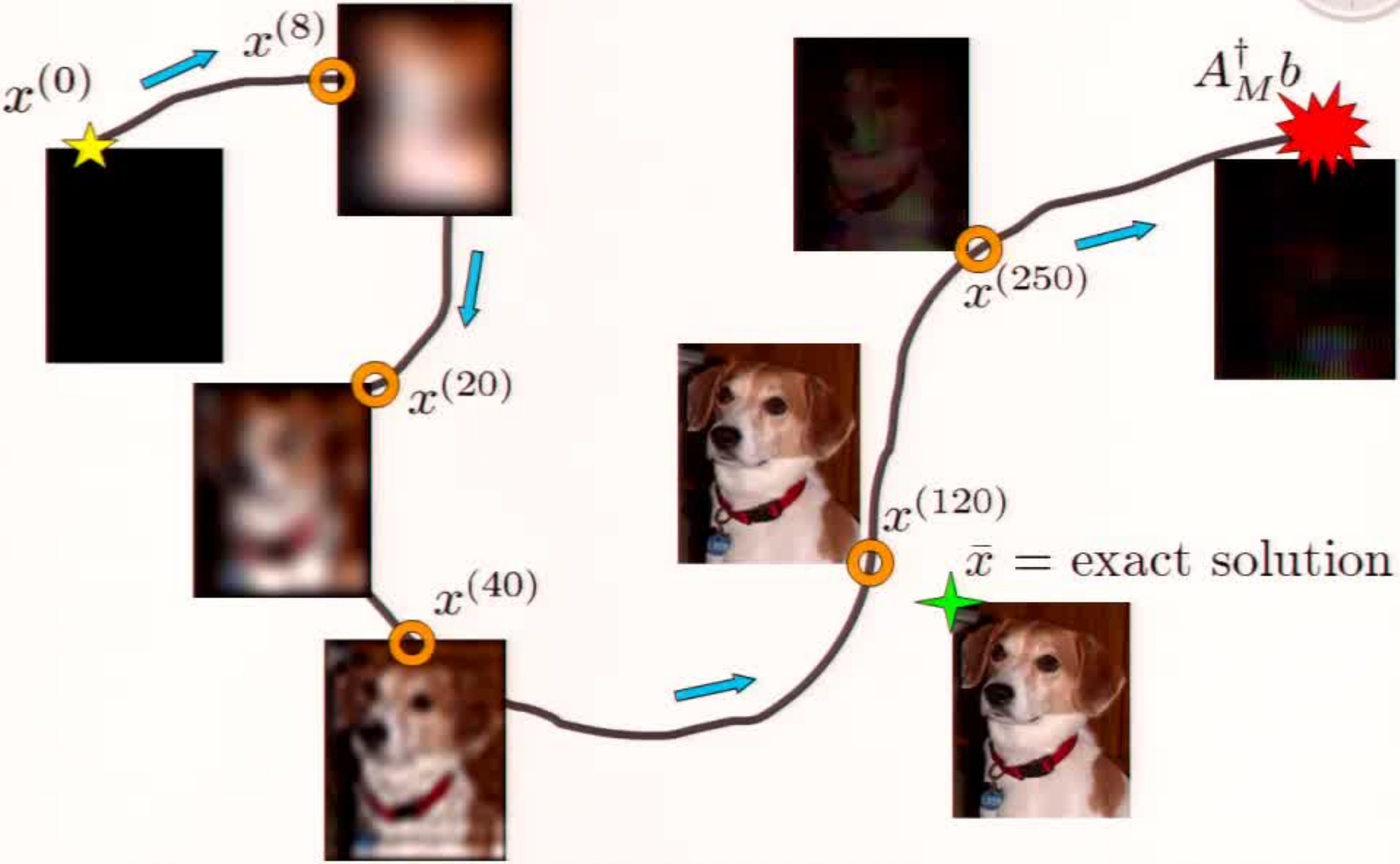
$$\mathcal{S}_k = \mathrm{span}\{v_1, v_2, \ldots, v_k\}.$$

Alternatively we can use a subspace determined by the given problem, e.g., the *Krylov subspace* $\mathcal{K}_k$ associated with a specific iterative method

$$\begin{aligned} \text{CGLS} &: \quad \mathrm{span}\{A^Tb, A^TA\,A^Tb, (A^TA)^2A^Tb, \ldots\}\ , \\ \text{GMRES} &: \quad \mathrm{span}\{b, A\,b, A^2\,b, \ldots\}\ , \\ \text{RRGMRES} &: \quad \mathrm{span}\{A\,b, A^2b, A^3b, \ldots\}\ . \end{aligned}$$

O'Leary & Simmons 1981, Kilmer & O'Leary 2001 – regularizing iterations.

Illustration of Semi-Convergence



Augmented Signal Subspace

Let $\mathcal{W}_p$ denote a linear subspace that captures additional specific components of the desired solution; $\dim(\mathcal{W}_p) = p \ll k =$ no. its.

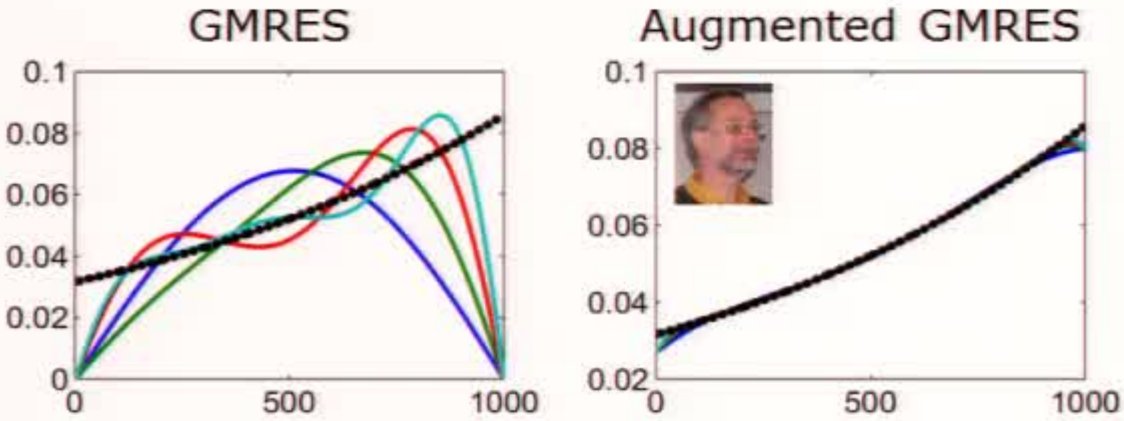
Then it can be advantageous to use an *augmented* linear subspace

$$\mathcal{S}_{p,k} = \mathcal{W}_p + \mathcal{K}_k, \qquad \mathcal{W}_p = \mathcal{R}(W_p) = \operatorname{span}\{w_1, \ldots, w_p\} \ .$$

Ex.: `deriv2` & GMRES.

All vectors in the Krylov subspace → 0 at end points. Now use

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



Here we want an efficient CGLS-type algorithm to solve the problem

$$\boxed{\min_x \|A\,x - b\|_2^2 \quad \text{s.t.} \quad x \in \mathcal{S}_{p,k} = \mathcal{W}_p + \mathcal{K}_k(A^TA, A^Tb) \ .}$$

Overview of Methods

Square matrix $A \in \mathbb{R}^{n \times n}$

- "Augmented (RR)GMRES" (Baglama, Reichel 2007), where the subspace augmentation idea was originally formulated. An elegant and efficient algorithmthat uses an incorrect subspace.
- "R^{3}GMRES" (Dong, Garde, H 2014), uses the correct subspace, less elegant, still efficient.

Rectangular matrix $A \in \mathbb{R}^{m \times n}$ (this work)

- In some problems (e.g., tomography) the matrix A is rectangular.
- In some problems (tomography, inverse heat equation) the Arnoldi subspace is not suited.
- "LBAS" – Lanczos bidiagonalization with augmented subspace.
- Open question: can we use LSQR or LSMR to implement this?→

Towards our Algorithm LBAS

We want to solve

$$\min_x \|A\,x - b\|_2^2 \quad \text{s.t.} \quad x \in \mathcal{W}_p + \mathcal{K}_k(A^T A, A^T b) \ .$$

In principle we could use, say, a Hessenberg decomposition

$$A\,[\,W_p\,,\,A^T b\,,A^T A\,A^T b\,,\cdots,\,(A^T A)^{k-1} A^T b\,] = V_{p+k+1}\,H_{p+k}$$

and compute the solution as

$$\begin{aligned} x^{(k)} &= [\,W_p\,,\,A^T b\,,A^T A\,A^T b\,,\ldots,\,(A^T A)^{k-1} A^T b\,]\,y^{(k)} \ , \\ y^{(k)} &= \operatorname{argmin}_y \|H_{p+1} y - V_{p+k+1}^T b\|_2^2 \ . \end{aligned}$$

But we prefer to use a stable and efficient “standard” algorithm.

Run the *bidiagonalization* algorithm to compute an orthonormal basis of $\mathcal{K}_k(A^T A, A^T b)$, and augment it by $\mathcal{W}_p$ in each step of the algorithm.

This seems cumbersome – but the overhead is favorably small!

Setting the Stage for Our Algorithm

At step k we have the decomposition

$$A\,[\,V_k\,,\,W_p\,] = \left[\,U_{k+1}\,,\,\widetilde{U}_k\,\right] \left[\begin{array}{cc} B_k & G_k \\ 0 & F_k \end{array}\right]$$

where

- $A\,V_k = U_{k+1}B_k$ is obtained after k steps of the bidiag. process.
- $V_k \in \mathbb{R}^{n\times k}$ has orthonormal columns that span $\mathcal{K}_j(A^TA, A^Tb)$.
- $U_{k+1} \in \mathbb{R}^{m\times(k+1)}$ has orthonormal columns, $u_1 = b/\|b\|_2$.
- $\widetilde{U}_k \in \mathbb{R}^{m\times p}$: $\mathrm{range}(AW_p) = \mathrm{range}(U_{k+1}G_k + \widetilde{U}_k F_k)$ and $\widetilde{U}_k^T U_{k+1} = 0$.
- $B_k \in \mathbb{R}^{(k+1)\times k}$ is a lower bidiagonal matrix.
- $F_k \in \mathbb{R}^{p\times p}$ and *changes in every iteration.*
- G_k is $(k+1)\times p$ and is *updated* along with B_k.

The columns of $[\,V_j\,,\,W_p\,]$ form a basis for $\mathcal{S}_{p,j}$.

More Details

Recall that

$$A\,[\,V_k\,,\,W_p\,] = \left[U_{k+1}\,,\,\widetilde{U}_k\right]\left[\begin{array}{cc} B_k & G_k \\ 0 & F_k \end{array}\right] .$$

The matrices $G_k \in \mathbb{R}^{(k+1)\times p}$ and $F_k \in \mathbb{R}^{p\times p}$ are composed of the coefficients of AW_p with respect to basis of range(U_{k+1}) and range$(\widetilde{U}_k)$, respectively:

$$G_k = U_{k+1}^T A W_p, \qquad F_k = \widetilde{U}_k^T A W_p \; .$$

Then the iterate $x^{(k)} \in \mathcal{S}_{p,k}$ is given by $x^{(k)} = [\,V_k\,,\,W_p\,]\,y^{(k)}$, where

$$y^{(k)} = \mathrm{argmin}_y \left\| \left[\begin{array}{cc} B_k & G_k \\ 0 & F_k \end{array}\right] y - \left[\begin{array}{c} U_{k+1}^T \\ \widetilde{U}_j^T \end{array}\right] b \right\|_2^2 .$$

Algorithm: LBAS

1. Set $U_1 = b/\|b\|_2$, $V_0 = [\,]$, $B_0 = [\,]$, $G_0 = U_1^T A W_p$, and $k = 1$.
2. Use the bidiag. process to obtain v_k, u_{k+1} such that $A\,V_k = U_{k+1} B_k$, where
$$V_k = [V_{k-1}, v_k],\; U_{k+1} = [U_k, u_{k+1}],\; B_k = \begin{bmatrix} B_{k-1} & 0 \\ & \times \\ 0 & \times \end{bmatrix}.$$
3. Compute $G_k = \begin{bmatrix} G_{k-1} \\ u_{k+1}^T A W_p \end{bmatrix} \in \mathbb{R}^{(k+1)\times p}$.
4. Orthonormalize AW_p with respect to U_{k+1} to obtain $\widetilde{U}_k \in \mathbb{R}^{m\times p}$.
5. Compute $F_k = \widetilde{U}_k^T A W_p \in \mathbb{R}^{p\times p}$.
6. Solve $\min_y \left\| \begin{bmatrix} B_k & G_k \\ 0 & F_k \end{bmatrix} y - \begin{bmatrix} U_{k+1}^T \\ \widetilde{U}_k^T \end{bmatrix} b \right\|_2^2$ to obtain $y^{(k)}$.
7. Then $x^{(k)} = [\,V_k\,,\,W_p\,]\,y^{(k)}$.
8. Stop, or set $k := k+1$ and return to step 2.

Recomputation of $\widetilde{U}_k$ and F_k in each step; but p is small!

Efficient and Stable Implementation

In each step we update the orthogonal factorization:

$$\begin{bmatrix} B_k & G_k \\ 0 & F_k \end{bmatrix} = Q \begin{bmatrix} T_k^{(11)} & T_k^{(12)} \\ 0 & T_k^{(22)} \\ 0 & 0 \end{bmatrix},$$

$T_k^{(11)} \in \mathbb{R}^{k\times k}$ and $T_k^{(22)} \in \mathbb{R}^{p\times p}$ are upper triangular, Q is orthogonal.

Update $T_k^{(11)}$ via Givens rotations that are also applied to G_k and $U_{k+1}^T b$.

$\widetilde{U}_k$ is already orthogonal to U_k, hence (in principle) we can perform the update

$$\widetilde{U}_{k+1} = (I_m - u_{k+1} u_{k+1}^T)\, \widetilde{U}_k.$$

For numerical stability: must reorthogonalize the columns of V_k, U_{k+1}, and $\widetilde{U}_k$. Consider the use of partial reorthogonalization.

Algorithm HYBR (Chung, Nagy, O'Leary 2008) uses full reorthogonalization.

Numerical Examples

Setting up the test problems:

1. Generate noise-free system: $A\,x_{\text{exact}} = b_{\text{exact}}$.
2. Add noise: $b = b_{\text{exact}} + e$ where e is a random vector of Gaussian white noise scaled such that $\|e\|_2 / \|b_{\text{exact}}\|_2 = \eta$.
3. We show best solution within the iterations plus:
   - relative error $\|x_{\text{exact}} - x^{(k)}\|_2 / \|x_{\text{exact}}\|_2$,
   - relative residual norm $\|b - A\,x^{(k)}\|_2 / \|b\|_2$.

We compare combinations of the following algorithms:

- CGLS is the implementation from Regularization Tools.
- RRGMRES is the implementation from Regularization Tools.
- R^{3}GMRES is our implementation (Dong, Garde, H 2014).
- LBAS is our new algorithm.

Large Component in Augment. Subspace

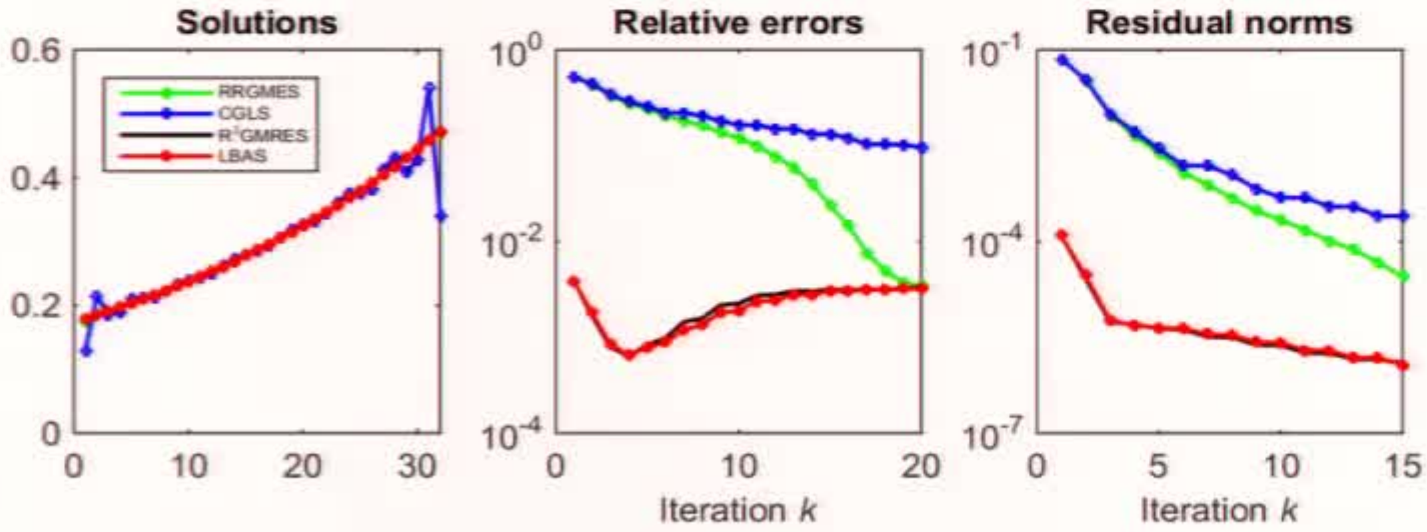
Test problem `deriv2(n,2)`, $n = 32$, relative noise level $\eta = 10^{-5}$.

$$\mathcal{W}_2 = \mathrm{span}\{w_1, w_2\}, \quad w_1 = (1, 1, \ldots, 1)^T, \quad w_2 = (1, 2, \ldots, n)^T.$$

For this problem

$$\|W_2 W_2^T x_{\text{exact}}\|_2 / \|x_{\text{exact}}\|_2 = 0.99 \ ,$$
$$\|(I - W_2 W_2^T) x_{\text{exact}}\|_2 / \|x_{\text{exact}}\|_2 = 0.035 \ ;$$

we only need to spend effort in capturing the small component in $\mathcal{W}_2^{\perp}$.

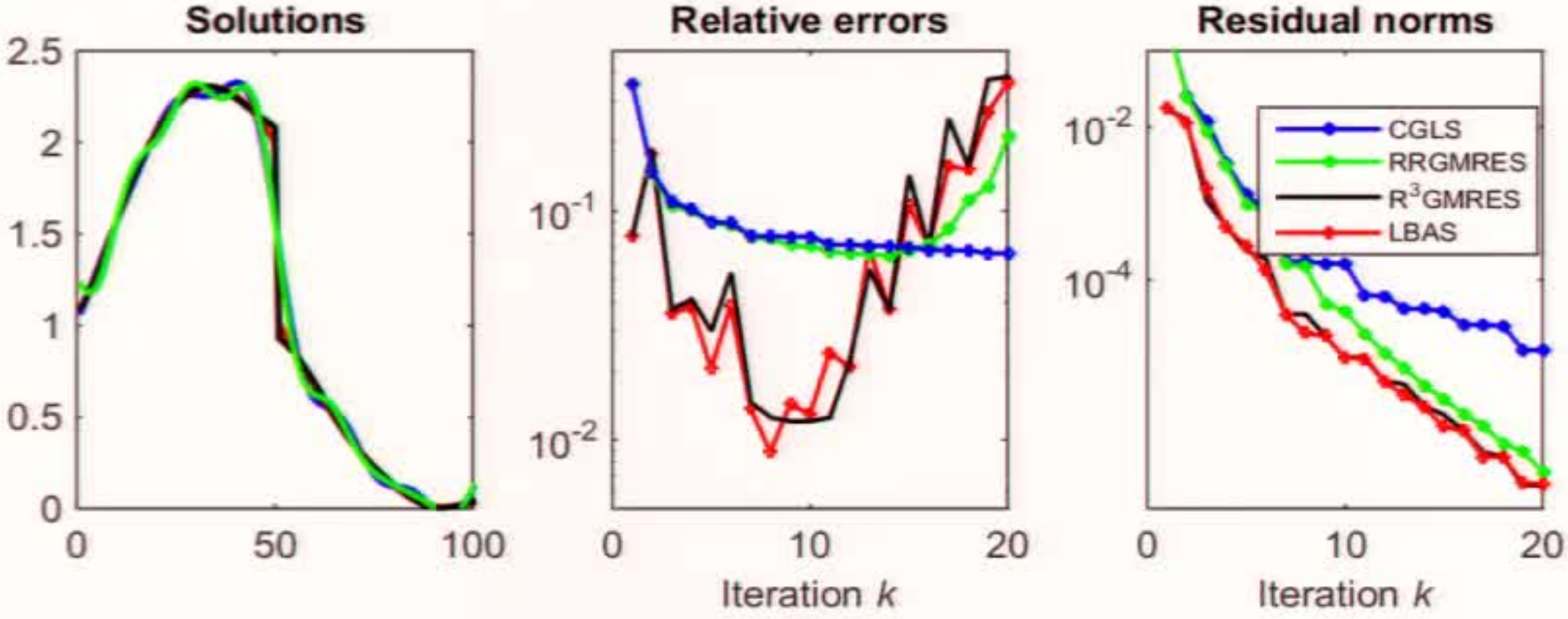


Capture a Discontinuity

Test problem `gravity(n)`, $n = 100$, $\eta = 10^{-3}$, exact sol. changed to include a discontinuity between elements $\ell = 50$ and $\ell + 1 = 51$.

Augmentation matrix W_2 allows us to represent this discontinuity:

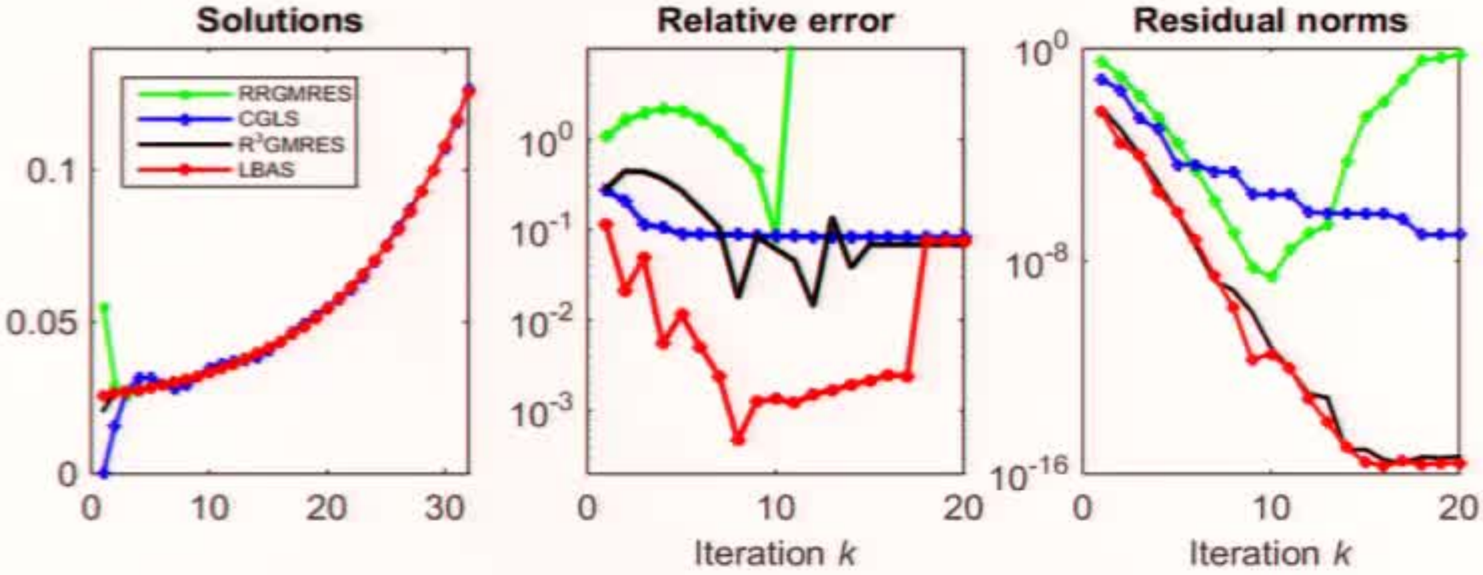
$$w_1 = \begin{bmatrix} \text{ones}(\ell, 1) \\ \text{zeros}(n-\ell, 1) \end{bmatrix}, \quad w_2 = \begin{bmatrix} \text{zeros}(\ell, 1) \\ \text{ones}(n-\ell, 1) \end{bmatrix}.$$



Fix Boundary Conditions

$$\int_0^{\pi} t \exp(-s\,t^2)\, f(t)\, dt = g(s), \quad 0 \le s \le \pi \qquad m = n = 32.$$

$$\mathcal{W}_2 = \mathrm{span}\{w_1, w_2\}, \quad w_1 = (1,1,\ldots,1)^\top, \quad w_2 = (1,2,\ldots,n)^\top.$$



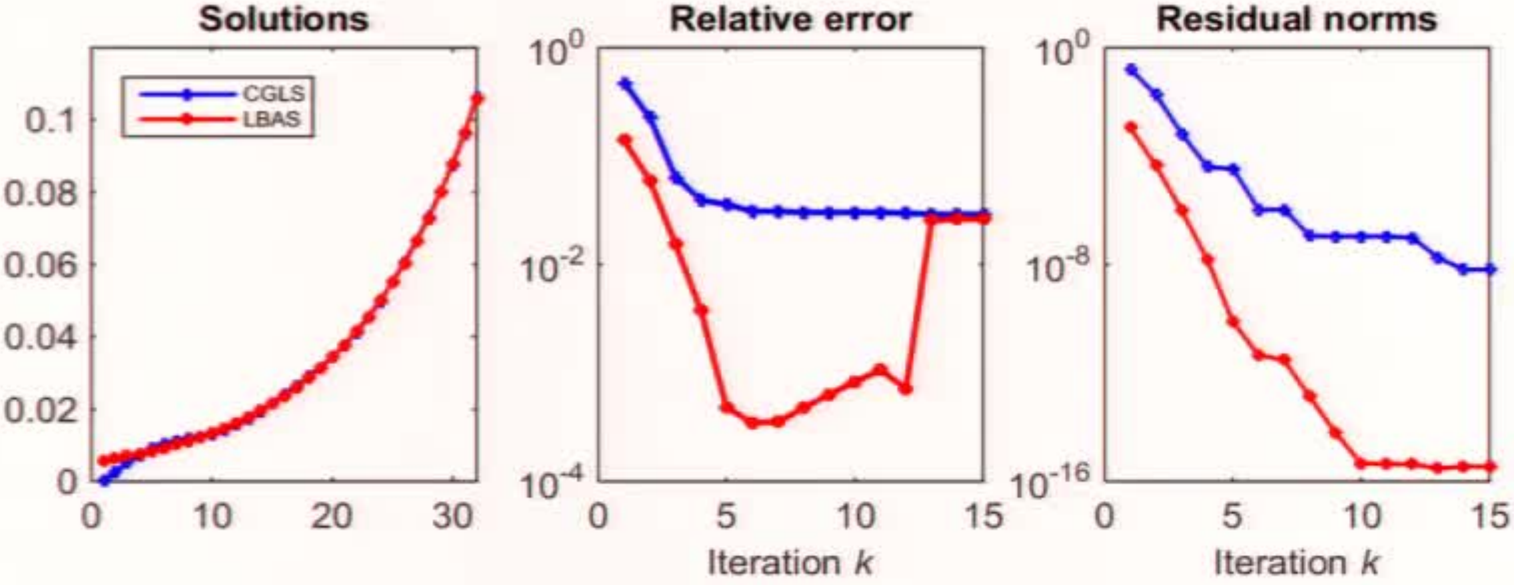
Here $\mathcal{W}_2$ compensates for the "incorrect" or "incompatible" boundary conditions implicit in A, by allowing the regularized solutions to have nonzero values and nonzero derivatives at the endpoints.

Fix Boundary Conditions, Rectangular *A*

$$\int_0^{\pi/2} t \exp(-s\,t^2)\, f(t)\, dt = g(s), \quad 0 \le s \le \pi \qquad m = 64,\ n = 32.$$

$$\mathcal{W}_2 = \mathrm{span}\{w_1, w_2\}, \quad w_1 = (1, 1, \ldots, 1)^\top, \quad w_2 = (1, 2, \ldots, n)^\top.$$

The matrix A is rectangular so RRGMRES and R^{3}GMRES cannot be used.



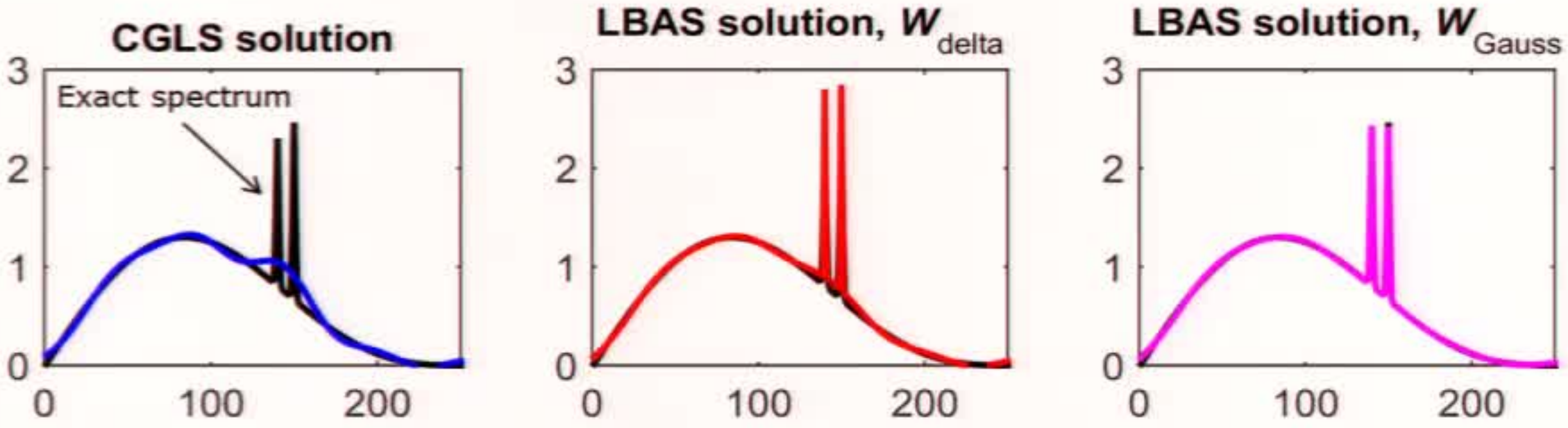
Compute Spectrum of X-Ray Source

The spectrum of an X-ray source (where accelerated electrons hit an anode) consists of a *continuous spectrum* superimposed with *line spectra*.

We know the frequencies of the line spectral, so we can easily incorporate this information through the augmentation subspace.

Experiment with two choices:

- W_{delta} – two delta functions at the right frequencies,
- W_{Gauss} – two narrow Gauss functions at the right frequencies.



Many thanks to Jan Sijbers for inspiration to this example.

Conclusions

- We consider (again) how to augment the Krylov subspace.
- Focus here on rectangular matrices and Lanczos bidiag.
- We develop an efficent algorihtm LBAS.
- Numerical examples demonstrate the advantage of LBAS.
- Future work:
  - Selective reorthogonalization?
  - Is it occasionally necessary to do the MGS twice?
  - A similar algorithm based on MINRES/MR-II?
  - Hybrid algorithm with regularization of projected problem!