Krylov Subspace Methods: GMRES, BICG

Matthew King-Roskamp[1]

[1] Department of Mathematics
McGill University

October 27th, 2020

An Overview

1. Part 1: The Krylov subspace
2. Part 2: GMRES - A First Approach.
   - The Problem
   - The GMRES Algorithm
   - Analysis of computational cost and convergence
3. Part 3: BCG - Extending CG
   - From CG to BCG, Relaxing Symettry
   - Algorithm statement
   - Advantages and Disadvantages: extending BCG

Follow the treatment of Trefethen Bau [TBl97, Lec. 33-39], and Saad [Saa03, Ch 6,7]

The Setting

- Suppose we want to do numerical linear algebra with a matrix $A \in \mathbb{R}^{m\times m}$. However the matrix A is infeasible or impossible to work with directly
    - m is too large.
    - We cannot directly access A at all.
- So, what can we do?
  To do efficient computations we usually factor A?

The Setting

- Suppose we want to do numerical linear algebra with a matrix $A \in \mathbb{R}^{m \times m}$. However the matrix A is infeasible or impossible to work with directly
  - m is too large.
  - We cannot directly access A at all.
- So, what can we do?
  To do efficient computations we usually factor A?

The Setting

- Suppose we want to do numerical linear algebra with a matrix $A \in \mathbb{R}^{m \times m}$. However the matrix A is infeasible or impossible to work with directly
  - m is too large.
  - We cannot directly access A at all.
- So, what can we do?
  To do efficient computations we usually factor A?

The Setting

- Suppose we want to do numerical linear algebra with a matrix $A \in \mathbb{R}^{m \times m}$. However the matrix A is infeasible or impossible to work with directly
  - m is too large.
  - We cannot directly access A at all.
- So, what can we do?
  To do efficient computations we usually factor A?

The Krylov Subspace

- The Krylov subspace $\mathcal{K}_n$ is defined as

$$\mathcal{K}_n(a, b) = \mathcal{K}_n = \text{span}(b, Ab, \ldots, A^{n-1}b)$$

  We've seen that $\mathcal{K}_n$ contains information about solutions of $Ax = b$, eigenvalues of A, and so on. (e.g. CG!)

- Orthogonalizing these vectors $A^i b$ gives basis $\{q_1, \ldots, q_n\}$. equivalently

$$K_n = \left[\ b \ \middle|\ Ab \ \middle|\ \ldots \ \middle|\ A^{n-1}b\ \right] = Q_n R_n$$

  for the QR factors of K_n

The Krylov Subspace

- The Krylov subspace $\mathcal{K}_n$ is defined as

$$\mathcal{K}_n(a, b) = \mathcal{K}_n = \mathrm{span}(b, Ab, \ldots, A^{n-1}b)$$

We've seen that $\mathcal{K}_n$ contains information about solutions of $Ax = b$, eigenvalues of A, and so on. (e.g. CG!)

- Orthogonalizing these vectors $A^i b$ gives basis $\{q_1, \ldots, q_n\}$. equivalently

$$K_n = \left[\ b \mid Ab \mid \ldots \mid A^{n-1}b \ \right] = Q_n R_n$$

for the QR factors of K_n

Computing with this space

- A natural question is how does Q_n relate to A? Well using the definition of $\mathcal{K}_n$

$$Aq_n = \left(\sum_{i=0}^{n-1} c_i A^{i+1} b\right) = \sum_{i=1}^{n} \tilde{h}_{in} q_i$$

- But writing this in matrix form

$$AQ_n = Q_{n+1}\tilde{H}_n = Q_n \begin{bmatrix} h_{11} & \cdots & h_{1n} \\ h_{21} & & \vdots \\ & \ddots & \\ & & h_{n+1,n} \end{bmatrix}$$

Where $\tilde{H}_n$ is upper Hessenberg.

Computing with this space

- A natural question is how does Q_n relate to A? Well using the definition of $\mathcal{K}_n$

$$Aq_n = \left(\sum_{i=0}^{n-1} c_i A^{i+1} b \right) = \sum_{i=1}^{n} \tilde{h}_{in} q_i$$

- But writing this in matrix form

$$AQ_n = Q_{n+1} \tilde{H}_n = Q_n \begin{bmatrix} h_{11} & \cdots & h_{1n} \\ h_{21} & & \vdots \\ & \ddots & \\ & & h_{n+1,n} \end{bmatrix}$$

Where $\tilde{H}_n$ is upper Hessenberg.

The Arnoldi Iteration

Removing the last row of $\tilde{H}_n$ gives a square matrix $H_n \in \mathbb{R}^{n \times n}$ with $H_n = Q_n^T A Q_n$

Arnoldi Iteration:

Given b, and the ability to compute $x \rightarrow Ax$:

```
for $j = 1, 2, \ldots$ do
    $v = Aq_n$; for $j = 1, \ldots, n$ do
        $h_{jn} = q_j^T v$
        $v = v - h_{jn} q_j$
    end
    $h_{n+1,n} = \|v\|$
    $q_{n+1} = v / h_{n+1,n}$
end
```

Output: $Q \in \mathbb{R}^{n \times n}$, $\tilde{H}_n \in \mathbb{R}^{n+1 \times n}$.

Lets Pause:

- With a simple algorithm (modified Gram-Schmidt) applied to the Krylov vectors, we construct an orthonormal basis of the Krylov subspace, and simultaneously reduce the first $n \times n$ submatrix of A to upper Hessenberg form.
- At no point during this process do we ever explicity work with the matrix A, beyond matrix vector multiplication.
- One can make variants of Arnoldi which use different projections (e.g. Householder!) to reduce computational cost.

Lets Pause:

- With a simple algorithm (modified Gram-Schmidt) applied to the Krylov vectors, we construct an orthonormal basis of the Krylov subspace, and simultaneously reduce the first $n \times n$ submatrix of A to upper Hessenberg form.
- At no point during this process do we ever explicity work with the matrix A, beyond matrix vector multiplication.
- One can make variants of Arnoldi which use different projections (e.g. Householder!) to reduce computational cost.

Lets Pause:

- With a simple algorithm (modified Gram-Schmidt) applied to the Krylov vectors, we construct an orthonormal basis of the Krylov subspace, and simultaneously reduce the first $n \times n$ submatrix of A to upper Hessenberg form.
- At no point during this process do we ever explicity work with the matrix A, beyond matrix vector multiplication.
- One can make variants of Arnoldi which use different projections (e.g. Householder!) to reduce computational cost.

An aside on eigenvalues

- This process of factorizing

  $$H_n = Q_n^T A Q_n$$

  need not (and does not!) preserve eigenvalues. However, the eigenvalues of H_n, called the Ritz values are remarkably good estimates of the eigenvalues of A.

- Some intuition: the diagonals of H are Rayleigh Quotients w.r.t q_i. This can be made formal, but is beyond the scope of this talk.

An aside on eigenvalues

- This process of factorizing

  $$H_n = Q_n^T A Q_n$$

  need not (and does not!) preserve eigenvalues. However, the eigenvalues of H_n, called the Ritz values are remarkably good estimates of the eigenvalues of A.
- Some intuition: the diagonals of H are Rayleigh Quotients w.r.t q_i. This can be made formal, but is beyond the scope of this talk.

- Now that we have developed the Krylov subspace, let's use this! First solving approximately $Ax = b$. We want to find at step n,

$$x^* \in \mathcal{K}_n(A, b) \text{ minimizing } \|b - Ax^*\|$$

- Question: What norm do we want this be be minimized in? We've seen CG is optimal for PSD A under the norm

$$\|x\|_A = \sqrt{x^T Ax}.$$

  Lets do something more basic: simple 2-norm (least squares).

- One can more generally see this is as a projection method

$$x^* \in \mathcal{K} \text{ with } b - Ax^* \perp \mathcal{L}$$

  Where $\mathcal{K} = \mathcal{K}_n, \mathcal{L} = A\mathcal{K}_n$.
  Sanity check: What are $\mathcal{K}, \mathcal{L}$ for CG with $A = A^T$?

- Just $\mathcal{K}_n$.

- Now that we have developed the Krylov subspace, let's use this! First solving approximately $Ax = b$. We want to find at step n,

$$x^* \in \mathcal{K}_n(A, b) \text{ minimizing } \|b - Ax^*\|$$

- Question: What norm do we want this be be minimized in? We've seen CG is optimal for PSD A under the norm

$$\|x\|_A = \sqrt{x^T Ax}.$$

  Lets do something more basic: simple 2-norm (least squares).

- One can more generally see this is as a projection method

$$x^* \in \mathcal{K} \text{ with } b - Ax^* \perp \mathcal{L}$$

  Where $\mathcal{K} = \mathcal{K}_n, \mathcal{L} = A\mathcal{K}_n$.
  Sanity check: What are $\mathcal{K}, \mathcal{L}$ for CG with $A = A^T$?

- Just $\mathcal{K}_n$.

- Now that we have developed the Krylov subspace, let's use this! First solving approximately $Ax = b$. We want to find at step n,

$$x^* \in \mathcal{K}_n(A, b) \text{ minimizing } \|b - Ax^*\|$$

- Question: What norm do we want this be be minimized in? We've seen CG is optimal for PSD A under the norm

$$\|x\|_A = \sqrt{x^T Ax}.$$

  Lets do something more basic: simple 2-norm (least squares).

- One can more generally see this is as a projection method

$$x^* \in \mathcal{K} \text{ with } b - Ax^* \perp \mathcal{L}$$

  Where $\mathcal{K} = \mathcal{K}_n, \mathcal{L} = A\mathcal{K}_n$.
  Sanity check: What are $\mathcal{K}, \mathcal{L}$ for CG with $A = A^T$?

- Just $\mathcal{K}_n$.

- Now that we have developed the Krylov subspace, let's use this! First solving approximately $Ax = b$. We want to find at step n,

$$x^* \in \mathcal{K}_n(A, b) \text{ minimizing } \|b - Ax^*\|$$

- Question: What norm do we want this be be minimized in? We've seen CG is optimal for PSD A under the norm

$$\|x\|_A = \sqrt{x^T Ax}.$$

  Lets do something more basic: simple 2-norm (least squares).

- One can more generally see this is as a projection method

$$x^* \in \mathcal{K} \text{ with } b - Ax^* \perp \mathcal{L}$$

  Where $\mathcal{K} = \mathcal{K}_n, \mathcal{L} = A\mathcal{K}_n$.
  Sanity check: What are $\mathcal{K}, \mathcal{L}$ for CG with $A = A^T$?

- Just $\mathcal{K}_n$.

GMRES Problem

- For each n, find $x_{n+1} \in A\mathcal{K}_n$ minimizing the least squares distance, namely

$$x_{n+1} = \operatorname*{argmin}_c \|AK_n c - b\|_2$$

So; how does one do this? We've seen QR solutions, so factor $AK_n = QR$?. This is numerically unstable

- Instead use orthonomal basis of Arnoldi! For $x = Q_n y$

$$AK_n c - b = AQ_n y - b = Q_{n+1}\tilde{H}_n y - b$$

- final useful trick is to note Q_{n+1} spans $\mathbb{R}^n$, so that

$$\|Q_{n+1}\tilde{H}_n y - b\|_2 = \|\tilde{H}_n y - Q_{n+1}^T b\|_2 = \|\tilde{H}_n y - \|b\|_2 e_1\|_2$$

GMRES Problem

- For each n, find $x_{n+1} \in A\mathcal{K}_n$ minimizing the least squares distance, namely

  $$x_{n+1} = \operatorname*{argmin}_c \|AK_n c - b\|_2$$

  So; how does one do this? We've seen QR solutions, so factor $AK_n = QR$?. This is numerically unstable

- Instead use orthonomal basis of Arnoldi! For $x = Q_n y$

  $$AK_n c - b = AQ_n y - b = Q_{n+1}\tilde{H}_n y - b$$

- final useful trick is to note Q_{n+1} spans $\mathbb{R}^n$, so that

  $$\|Q_{n+1}\tilde{H}_n y - b\|_2 = \|\tilde{H}_n y - Q_{n+1}^T b\|_2 = \|\tilde{H}_n y - \|b\|_2 e_1\|_2$$

GMRES Problem

- For each n, find $x_{n+1} \in A\mathcal{K}_n$ minimizing the least squares distance, namely

$$x_{n+1} = \underset{c}{\operatorname{argmin}} \|AK_n c - b\|_2$$

So; how does one do this? We've seen QR solutions, so factor $AK_n = QR$?. This is numerically unstable

- Instead use orthonomal basis of Arnoldi! For $x = Q_n y$

$$AK_n c - b = AQ_n y - b = Q_{n+1}\tilde{H}_n y - b$$

- final useful trick is to note Q_{n+1} spans $\mathbb{R}^n$, so that

$$\|Q_{n+1}\tilde{H}_n y - b\|_2 = \|\tilde{H}_n y - Q_{n+1}^T b\|_2 = \|\tilde{H}_n y - \|b\|_2 e_1\|_2$$

Algorithm Statement

```
GMRES:
Given $b$, and the ability to compute $x \to Ax$: for
 $j = 1, 2, \ldots$ do
    $v = Aq_n$
    for $j = 1, \ldots, n$ do
        $h_{jn} = q_j^T v$
        $v = v - h_{jn} q_j$
    end
    $h_{n+1,n} = \|v\|$
    $q_{n+1} = v / h_{n+1,n}$ Find $y$ minimizing
     $\|r_n\|_2 = \|\tilde{H}_n y - \|b\|_2 e_1\|_2$
    Set $x_n = Q_n y$
end
```

```
GMRES:
Given $b$, and the ability to compute $x \to Ax$: for
$j = 1, 2, \ldots$ do
    ⟨ Do step $n$ of Arnoldi ⟩
    Find $y$ minimizing $\|r_n\|_2 = \|\tilde{H}_n y - \|b\|_2 e_1\|_2$
    Set $x_n = Q_n y$
end
```

How to solve this system?

- $\tilde{H}_n$ is upper hessenberg, so can compute its QR factorization in $O(n^2)$ operations. But, we can make an easier observation:
- Given a QR factorization of $\tilde{H}_n$, how can we find a QR factorization of $\tilde{H}_{n+1}$?
- Well, we need to zero out one new subdiagonal entry and update the last column. This is 1 Givens Rotation and $O(n)$ work!

How to solve this system?

- $\tilde{H}_n$ is upper hessenberg, so can compute its QR factorization in $O(n^2)$ operations. But, we can make an easier observation:
- Given a QR factorization of $\tilde{H}_n$, how can we find a QR factorization of $\tilde{H}_{n+1}$?
- Well, we need to zero out one new subdiagonal entry and update the last column. This is 1 Givens Rotation and $O(n)$ work!

How to solve this system?

- $\tilde{H}_n$ is upper hessenberg, so can compute its QR factorization in $O(n^2)$ operations. But, we can make an easier observation:
- Given a QR factorization of $\tilde{H}_n$, how can we find a QR factorization of $\tilde{H}_{n+1}$?
- Well, we need to zero out one new subdiagonal entry and update the last column. This is 1 Givens Rotation and $O(n)$ work!

Convergence of GMRES

Theorem (Convergence of GMRES)

At step n of the GMRES iteration, if $A = X\Lambda X^{-1}$, the residual satisfies

$$\frac{\|r_n\|_2}{\|b\|_2} \leq \kappa(x) \inf_{p \in P_n} \sup_{\lambda \in \Lambda} |p(\lambda)|$$

This means controlling the eigenvalues of A is ideal for convergence. If the eigenvalues of A “poor” (i.e. $\sup_{\lambda \in \Lambda} |p(\lambda)| \approx 1$, convergence can be extremely slow. Preconditioning is key in this regime.

Proof.

As x_n is in the Krylov subspace, $x_n = q_{n-1}(A)$ for polynomial $q_{n-1} \in P_{n-1}$, we have
$r_n = b - Ax_n = (I - Aq_{n-1}(A))b = p_n(A)b$. Then,

$$\|r_n\|_2 = \|p_n(A)b\|_2 \leq \|p_n(A)\|_2\|b\|_2 = \inf_{p \in P_n} \|p(A)\|_2\|b\|_2$$

As A is diagonalizable, $A = X\Lambda X^{-1}$. Then

$$\begin{aligned}\|p(A)\|_2 = \|Xp(\Lambda)X^{-1}\|_2 &\leq \|X\|_2\|p(\Lambda)\|_2\|X^{-1}\|_2 \\ &\leq k(X) \sup_{\lambda \in \Lambda(A)} |p(\lambda)|.\end{aligned}$$

Rearranging gives the result. □

Biorthogonal Conjugate Gradient

- A new perspective on conjugate gradient, as a projection method: For $A = A^T$ find

$$x^* \in \mathcal{K}_n(A, b) \text{ with } b - Ax^* \perp \mathcal{K}_n(A^T, w)$$

Where of course here $A = A^T$; but what about in the nonsymettric case?

- Equivalently, we've seen that if A is symettric, it satisfies a three term recurrence. Or in matrix form,

$$A = V^T TV$$

For tridiagonal T.

Biorthogonal Conjugate Gradient

- A new perspective on conjugate gradient, as a projection method: For $A = A^T$ find

$$x^* \in \mathcal{K}_n(A, b) \text{ with } b - Ax^* \perp \mathcal{K}_n(A^T, w)$$

  Where of course here $A = A^T$; but what about in the nonsymettric case?

- Equivalently, we've seen that if A is symettric, it satisfies a three term recurrence. Or in matrix form,

$$A = V^T T V$$

  For tridiagonal T.

- Generalizing Arnoldi: Hessenberg Orthogonalization

  $$A = QHQ^T.$$

  What if we want the superdiagonal entries to be zero as well (upper and lower Hessenberg, tridiagonal)?
  We have to give up our orthogonal transformations: for nonsingular V

  $$A = VTV^{-1}.$$

- Transposing this equation and rearranging we see

  $$((V^T)^{-1})^T V = V^{-1} V = I.$$

  So while the columns of V are not orthogonal, they are orthogonal to the columns of $(V^T)^{-1} := W$. This is Biorthogonality.

- Generalizing Arnoldi: Hessenberg Orthogonalization

  $$A = QHQ^T.$$

  What if we want the superdiagonal entries to be zero as well (upper and lower Hessenberg, tridiagonal)?
  We have to give up our orthogonal transformations: for nonsingular V

  $$A = VTV^{-1}.$$

- Transposing this equation and rearranging we see

  $$((V^T)^{-1})^T V = V^{-1} V = I.$$

  So while the columns of V are not orthogonal, they are orthogonal to the columns of $(V^T)^{-1} := W$. This is Biorthogonality.

A New Iteration

- Arnoldi iteration in matrix form gave us

$$AQ_n = Q_{n+1}\tilde{H}_n, \qquad H_n = Q_n A Q_n^T$$

- Correspondingly for a Biorthogonal method, we have the iteration steps

$$\begin{aligned} AV_n &= V_{n+1}\tilde{T}_n \\ AW_n &= W_{n+1}\tilde{S}_n \\ T_n = S_n &= W_n^* A V_n \end{aligned}$$

- Here, T_n, S_n are tridiagional, and W_n, V_n are biorthogonal!

A New Iteration

- Arnoldi iteration in matrix form gave us

$$AQ_n = Q_{n+1}\tilde{H}_n, \qquad H_n = Q_n A Q_n^T$$

- Correspondingly for a Biorthogonal method, we have the iteration steps

$$\begin{aligned} AV_n &= V_{n+1}\tilde{T}_n \\ AW_n &= W_{n+1}\tilde{S}_n \\ T_n &= S_n = W_n^* A V_n \end{aligned}$$

- Here, T_n, S_n are tridiagional, and W_n, V_n are biorthogonal!

A New Iteration

- Arnoldi iteration in matrix form gave us

$$AQ_n = Q_{n+1}\tilde{H}_n, \qquad H_n = Q_n A Q_n^T$$

- Correspondingly for a Biorthogonal method, we have the iteration steps

$$\begin{aligned} AV_n &= V_{n+1}\tilde{T}_n \\ AW_n &= W_{n+1}\tilde{S}_n \\ T_n &= S_n = W_n^* A V_n \end{aligned}$$

- Here, T_n, S_n are tridiagional, and W_n, V_n are biorthogonal!

A New Iteration

- If we expand these equations out, calling the diagonal entries α_i and the sub/super diagonals β_i, γ_i respectively, we get a three term recurrence relation for v_n:

$$Av_n = \gamma_{n-1} v_n + \alpha_n v_n + \beta_n v_{n+1}$$

- and for w_n

$$A^T w_n = \beta_{n-1} w_n + \alpha_n w_n + \gamma_n w_{n+1}$$

This recurrence relation is the basis of the Biorthogonal Conjugate Gradient method

A New Iteration

- If we expand these equations out, calling the diagonal entries α_i and the sub/super diagonals β_i, γ_i respectively, we get a three term recurrence relation for v_n:

$$Av_n = \gamma_{n-1} v_n + \alpha_n v_n + \beta_n v_{n+1}$$

- and for w_n

$$A^T w_n = \beta_{n-1} w_n + \alpha_n w_n + \gamma_n w_{n+1}$$

This recurrence relation is the basis of the Biorthogonal Conjugate Gradient method

```
BCG (Biorthogonal Conjugate Gradient)
Given $x_0$, $b = p_0 = r_0$, $q_0 = s_0$ arbitrary, and the
 ability to compute $x \to Ax$, $y \to A^T y$: for $j = 1, 2, \ldots$
 do
    Set: $\alpha_n = (s_{n-1}^T r_{n-1})/(q_{n-1}^T A p_{n-1})$
    Update:
    $x_n = x_{n-1} + \alpha p_{n-1}$
    $r_n = r_{n-1} - \alpha A p_{n-1}$
    $s_n = s_{n-1} - \alpha A^T q_{n-1}$
    Set: $\beta_n = (s_n^T r_n)/(s_{n-1}^T r_{n-1})$
    Update:
    $p_n = r_n + \beta_n p_{n-1}$
    $q_n = s_n + \beta_n q_{n-1}$
end
```

Pros and Cons

- Sanity Check: $s_n^T r_j = 0$, and $q_n^T A p_j = 0$. for $j < n$.
- Advantages of BCG: minimal computation per iteration. Disadvantage: Convergence can be slow (or even erratic and non monotonic) compared to GMRES. Furthermore, it requires access to A^T.

  To solve these issues, many variants of BCG have been proposed:

  - Look ahead Lanczos, CG squared, Quasi-minimal residuals, BIC-Stab

Pros and Cons

- Sanity Check: $s_n^T r_j = 0$, and $q_n^T A p_j = 0$. for $j < n$.
- Advantages of BCG: minimal computation per iteration. Disadvantage: Convergence can be slow (or even erratic and non monotonic) compared to GMRES. Furthermore, it requires access to A^T.

  To solve these issues, many variants of BCG have been proposed:

  - Look ahead Lanczos, CG squared, Quasi-minimal residuals, BIC-Stab

Pros and Cons

- Sanity Check: $s_n^T r_j = 0$, and $q_n^T A p_j = 0$. for $j < n$.
- Advantages of BCG: minimal computation per iteration. Disadvantage: Convergence can be slow (or even erratic and non monotonic) compared to GMRES. Furthermore, it requires access to A^T.

  To solve these issues, many variants of BCG have been proposed:

  - Look ahead Lanczos, CG squared, Quasi-minimal residuals, BIC-Stab

Summary:

With this, we conclude. What should the listener take way?

- At no point during this presentation did we ever explicity work with the matrix A of A^T, beyond matrix vector multiplication
- However, using the Krylov subspace, we discovered factorizations of A into a Hessenberg matrix and tridiagnonal matrix.
- GMRES is a natural first approach working in the Krylov subspace, with convergence guarantees.
- BCG naturally extends CG. This is computationally cheap, with many high-performance extensions.

Summary:

With this, we conclude. What should the listener take way?

- At no point during this presentation did we ever explicity work with the matrix A of A^T, beyond matrix vector multiplication
- However, using the Krylov subspace, we discovered factorizations of A into a Hessenberg matrix and tridiagnonal matrix.
- GMRES is a natural first approach working in the Krylov subspace, with convergence guarantees.
- BCG naturally extends CG. This is computationally cheap, with many high-performance extensions.

Summary:

With this, we conclude. What should the listener take way?

- At no point during this presentation did we ever explicity work with the matrix A of A^T, beyond matrix vector multiplication
- However, using the Krylov subspace, we discovered factorizations of A into a Hessenberg matrix and tridiagnonal matrix.
- GMRES is a natural first approach working in the Krylov subspace, with convergence guarantees.
- BCG naturally extends CG. This is computationally cheap, with many high-performance extensions.

Summary:

With this, we conclude. What should the listener take way?

- At no point during this presentation did we ever explicity work with the matrix A of A^T, beyond matrix vector multiplication
- However, using the Krylov subspace, we discovered factorizations of A into a Hessenberg matrix and tridiagnonal matrix.
- GMRES is a natural first approach working in the Krylov subspace, with convergence guarantees.
- BCG naturally extends CG. This is computationally cheap, with many high-performance extensions.

Summary:

Thank you for your time! Any questions?

[Saa03] Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.

[TBI97] Lloyd N Trefethen and David Bau III. *Numerical linear algebra*, volume 50. Siam, 1997.