NA Project - Phase 2

Arihant Rastogi - Prasoon Dev - Hrishiraj Mitra

May 8, 2025

1 Major Concepts

This is a list of all major concepts used bu us to work towards extending the DCDM (Deep Conjugate Direction Method) to Bi-directional Conjugate Method and MINRES.

1.1 Prerequisites

Basis

A basis B of a vector space V over a field F (such as the real numbers R or the complex numbers C) is a linearly independent subset of V that spans V. For example $\hat{i}$, $\hat{j}$ and $\hat{k}$ form the basis for the vector space of 3D coordinate vector space. Every vector in the $\mathbb{R}^{\not\Vdash}$ space can be represented as $a\hat{i} + b\hat{j} + c\hat{k}$.

Positive Definite

A **Positive Definite** matrix is a matrix that has all eigenvalues $\lambda > 0$ and real.

Inner Product

An **inner product** is a function that takes two vectors and returns a scalar, satisfying properties such as linearity, symmetry, and positive-definiteness. In standard Euclidean space, the usual dot product is:

$$\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u}^T \mathbf{v}. \tag{1}$$

For a positive definite matrix $\mathbf{A}$, we define an **A-inner product**:

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbf{A}} = \mathbf{u}^T \mathbf{A} \mathbf{v}. \tag{2}$$

Conjugacy with Respect to a Matrix

Two nonzero vectors $\mathbf{u}$ and $\mathbf{v}$ are **A-conjugate** (or A-orthogonal) if:

$$\mathbf{u}^T \mathbf{A} \mathbf{v} = 0. \tag{3}$$

This means that, under the $\mathbf{A}$-inner product, the vectors are orthogonal.

1.2 Conjugate Method

It is used for solving SPD (Symmetric Positive Definite) linear systems, given by $Ax = b$.

As Direct Method

Let $P = \{p_1, p_2, ...p_n\}$ be a set of n mutually conjugate vectors with respect to A given by $p_i^T A p_j = 0 \; \forall \; i \neq j$. Hence P make a basis for $\mathbb{R}^n$. Then we may express the solution x_* of $Ax = b$ as

$$x_* = \sum_{i=1}^{n} \alpha_i p_i \tag{4}$$

$$Ax_* = \sum_{i=1}^{n} \alpha_i A p_i \tag{5}$$

$$p_k^T b = p_k^T A x_* = \sum_{i=1}^{n} \alpha_i p_k^T A p_i \tag{6}$$

$$= \sum_{i=1}^{n} \alpha_i \langle \mathbf{p_k}, \mathbf{p_i} \rangle_{\mathbf{A}} \tag{7}$$

$$= \alpha_k \langle \mathbf{p_k}, \mathbf{p_k} \rangle_{\mathbf{A}} \tag{8}$$

and thus,

$$\alpha_k = \frac{\langle \mathbf{p_k}, \mathbf{b} \rangle}{\langle \mathbf{p_k}, \mathbf{p_k} \rangle_{\mathbf{A}}} \tag{9}$$

This computes n different conjugate coefficients α_k for the conjugate directions p_k and thus helps in convergence to the true solution in the iterative method mentioned below.

As Iterative Method

Initially a random solution x_0 is assumed to be the initial step. The function to be minimised for convergence is given by

$$f(x) = \frac{1}{2} x^T A x - x^T b \tag{10}$$

Since gradient of this is given by

$$\nabla f(x) = Ax - b \tag{11}$$

We denote the **residual**, $\mathbf{r_0} = b_0 - Ax_0$. Clearly r_0 is negative of the initial gradient of the minimizing function and a conjugate direction can be given by $p_0 = b_0 - Ax_0$. The next conjugate direction at step k is given by the formula (built on Gram-Schmidt orthonormalization),

$$p_k = r_k - \sum_{i<k} \frac{r_k^T A p_i}{p_i^T A p_i} p_i = r_k - \sum_{i<k} \frac{\langle \mathbf{r_k}, \mathbf{p_i} \rangle_{\mathbf{A}}}{\langle \mathbf{p_i}, \mathbf{p_i} \rangle_{\mathbf{A}}} p_i \tag{12}$$

The updateion for step $k+1$ is given by,

$$x_{k+1} = x_k + \alpha_k p_k \tag{13}$$

where the update coefficent is found out by

$$\alpha_k = \frac{p_k^T (b - A x_k)}{p_k^T A p_k} = \frac{p_k^T r_k}{p_k^T A p_k} \tag{14}$$

Final Algorithm

$\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$
if $\|\mathbf{r}_0\|$ is sufficiently small, then return $\mathbf{x}_0$ as the result
$\mathbf{p}_0 := \mathbf{r}_0, \quad k := 0$
repeat
  $\alpha_k := \dfrac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{p}_k^T \mathbf{A}\mathbf{p}_k}$
  $\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k \mathbf{p}_k$
  $\mathbf{r}_{k+1} := \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k$
  if $\|\mathbf{r}_{k+1}\|$ is sufficiently small, then exit loop
  $\beta_k := \dfrac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k}$
  $\mathbf{p}_{k+1} := \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k$
  $k := k + 1$
end repeat
return $\mathbf{x}_{k+1}$ as the result

1.3 Minimum Residual Method (MINRES)

The Minimum Residual method is an iterative algorithm for solving systems of linear equations where the coefficient matrix is symmetric but not necessarily positive definite. Unlike the Conjugate Gradient method which requires positive definiteness, MINRES works for any symmetric matrix by minimizing the residual norm $\|\mathbf{b} - \mathbf{A}\mathbf{x}\|$ at each iteration.

Algorithm Overview

Given a symmetric matrix $\mathbf{A}$ and right-hand side $\mathbf{b}$, MINRES proceeds as follows:

Initialize:
$\mathbf{x}_0 =$ initial guess
$\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$
$\mathbf{p}_0 = \mathbf{r}_0$
$\mathbf{s}_0 = \mathbf{A}\mathbf{p}_0$
For $k = 1, 2, \ldots$ until convergence:
  $\alpha_{k-1} = \dfrac{\langle \mathbf{r}_{k-1}, \mathbf{s}_{k-1} \rangle}{\langle \mathbf{s}_{k-1}, \mathbf{s}_{k-1} \rangle}$
  $\mathbf{x}_k = \mathbf{x}_{k-1} + \alpha_{k-1} \mathbf{p}_{k-1}$
  $\mathbf{r}_k = \mathbf{r}_{k-1} - \alpha_{k-1} \mathbf{s}_{k-1}$
  If $\|\mathbf{r}_k\| <$ tolerance, stop
  $\mathbf{p}_k = \mathbf{s}_{k-1}$
  $\mathbf{s}_k = \mathbf{A}\mathbf{s}_{k-1}$
  Orthogonalization steps:
  $\beta_{k,1} = \dfrac{\mathbf{s}_k^T \mathbf{s}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{s}_{k-1}}$
  $\mathbf{p}_k = \mathbf{p}_k - \beta_{k,1} \mathbf{p}_{k-1}$
  $\mathbf{s}_k = \mathbf{s}_k - \beta_{k,1} \mathbf{s}_{k-1}$
  (Additional orthogonalization may be performed)

Convergence Properties

The convergence rate of MINRES depends on the condition number $\kappa(\mathbf{A})$ of the matrix:

$$\kappa(\mathbf{A}) = \frac{|\lambda_{\max}(\mathbf{A})|}{|\lambda_{\min}(\mathbf{A})|} \tag{15}$$

The residual norm is bounded by:

$$\|\mathbf{r}_k\| \leq 2 \left(\frac{\sqrt{\kappa(\mathbf{A})} - 1}{\sqrt{\kappa(\mathbf{A})} + 1} \right)^k \|\mathbf{r}_0\| \tag{16}$$

This shows that MINRES converges faster for matrices with smaller condition numbers.

Comparison with Conjugate Gradient

- MINRES works for any symmetric matrix, while CG requires positive definiteness
- Both methods use short recurrences to minimize storage requirements
- CG minimizes the A-norm of the error, while MINRES minimizes the 2-norm of the residual
- For SPD matrices, CG is typically preferred due to better numerical properties

1.4 Bidirectional Conjugate Method (BCM)

The Bidirectional Conjugate Method is an iterative algorithm for solving symmetric positive definite linear systems that approaches the solution from two directions simultaneously. Unlike standard Conjugate Gradient methods that build a single sequence of conjugate directions, BCM constructs two sequences of conjugate directions that work cooperatively to accelerate convergence.

Algorithm Overview

Given a symmetric positive definite matrix $\mathbf{A}$ and right-hand side $\mathbf{b}$, the Bidirectional Conjugate Method proceeds as follows:

$\mathbf{x}_0 :=$ initial guess

$\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$

$\mathbf{p}_0 := \mathbf{r}_0$

$\mathbf{q}_0 := \mathbf{A}\mathbf{r}_0$

if $\|\mathbf{r}_0\|$ is sufficiently small, then return $\mathbf{x}_0$ as the result

repeat

  // Forward direction update

  $\alpha_k := \dfrac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{p}_k^T \mathbf{q}_k}$

  $\mathbf{x}_{k+1/2} := \mathbf{x}_k + \alpha_k \mathbf{p}_k$

  $\mathbf{r}_{k+1/2} := \mathbf{r}_k - \alpha_k \mathbf{q}_k$

  // Backward direction update

  $\mathbf{z}_k := \mathbf{A}\mathbf{r}_{k+1/2}$

  $\gamma_k := \dfrac{\mathbf{r}_{k+1/2}^T \mathbf{r}_{k+1/2}}{\mathbf{r}_{k+1/2}^T \mathbf{z}_k}$

  $\mathbf{x}_{k+1} := \mathbf{x}_{k+1/2} + \gamma_k \mathbf{r}_{k+1/2}$

  $\mathbf{r}_{k+1} := \mathbf{r}_{k+1/2} - \gamma_k \mathbf{z}_k$

  if $\|\mathbf{r}_{k+1}\|$ is sufficiently small, then exit loop

  // Update conjugate directions

  $\beta_k := \dfrac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k}$

  $\mathbf{p}_{k+1} := \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k$

  $\mathbf{q}_{k+1} := \mathbf{A}\mathbf{p}_{k+1}$

  $k := k + 1$

end repeat

return $\mathbf{x}_{k+1}$ as the result

The algorithm employs a "midpoint" iteration (denoted with $k+1/2$) that represents the forward conjugate direction update, followed by a backward direction update to complete the full iteration.

Convergence Properties

The convergence rate of Bidirectional Conjugate Methods is generally better than standard Conjugate Gradient methods, especially for systems with clustered eigenvalues. For a symmetric positive definite matrix $\mathbf{A}$ with condition number $\kappa(\mathbf{A})$, the error at iteration k is bounded by:

$$\|\mathbf{x}_k - \mathbf{x}_*\|_{\mathbf{A}} \leq 2 \left(\frac{\sqrt{\kappa(\mathbf{A})} - 1}{\sqrt{\kappa(\mathbf{A})} + 1} \right)^{2k} \|\mathbf{x}_0 - \mathbf{x}_*\|_{\mathbf{A}} \tag{17}$$

This represents a potential acceleration factor of approximately 2 compared to the standard Conjugate Gradient method. The residual norm decreases according to:

$$\|\mathbf{r}_k\| \leq 2 \left(\frac{\sqrt{\kappa(\mathbf{A})} - 1}{\sqrt{\kappa(\mathbf{A})} + 1} \right)^{2k} \|\mathbf{r}_0\| \tag{18}$$

The method achieves exact convergence in at most $\lceil n/2 \rceil$ iterations for an $n \times n$ matrix in exact arithmetic, compared to n iterations for standard CG.

Comparison with Conjugate Gradient

- **Similarities:**
  - Both methods are applicable to symmetric positive definite systems
  - Both use short recurrences to maintain computational efficiency

- Both build conjugate directions with respect to the matrix $\mathbf{A}$
- Both methods converge in finite steps in exact arithmetic

- **Differences:**
  - **Iteration complexity:** BCM requires two matrix-vector products per iteration, while CG requires only one
  - **Storage requirements:** BCM requires storage for additional vectors compared to CG's minimal storage requirements
  - **Convergence rate:** BCM can be potentially twice as fast in terms of iteration count
  - **Theoretical convergence:** BCM converges in at most $\lceil n/2 \rceil$ iterations, while CG converges in at most n iterations
  - **Robustness:** BCM is more robust against round-off errors in ill-conditioned systems
  - **Parallelization:** BCM is better suited for parallel implementation due to its bidirectional approach

- **When to use:**
  - BCM is preferred when the system is ill-conditioned but symmetric positive definite, computational resources allow for additional matrix-vector products, and faster convergence is critical
  - CG is preferred when matrix-vector products are expensive, memory constraints are significant, the system is well-conditioned, or implementation simplicity is a priority

The Bidirectional Conjugate Method represents a trade-off between iteration count and computational work per iteration, often resulting in overall computational savings for challenging problems.

2 Current Work

- Compiled all concepts that are required as prerequisites.
- Ran and identified the system constraints on the original DCDM code.
- Compiled code for Bi-CG and MINRES in Python. Here is the attached link to GitHub repository. We have included the slides presented in Phase 2 in the repository itself.

3 Work to Do

- Extend the neural network to Bi-directional Conjugate Method.
- Create a new dataset that tests the effect of neural network implementation over already implemented Bi-CG implementation.
- Figure out if the same concepts can be extended to MINRES based on both system and time constraints.