Abstract

We present two modified Incomplete Cholesky factorization preconditioners for symmetric, banded linear systems. The first approach normalizes the matrix with respect to its diagonal before the incomplete factorization, removing the operation of division from the inner loop. An additional transformation is employed in the second preconditioner to remove the first co-diagonals of the matrix. In both cases, we take into account the remainder of the factorization by performing an additional iteration within the preconditioned conjugate gradient method. Numerical experiments are performed in a finite-difference approximation of the electric potential problem arising from the 2D electrorresistivity method.

Introduction

The preconditioned conjugate gradient algorithm (PCG) with incomplete Cholesky (IC) factorization is currently the most effective method for the solution of banded, symmetric, positive definite linear systems, arising from numerical modeling with finite difference or finite element techniques.

Finite-difference modeling

Assuming that the electric conductivity \( \sigma \) of the medium varies only along the axis \( x \) and the depth \( z \), the electrical potential generated by a pointwise source at \((x_0; 0; 0)\) is a solution of the Poisson equation,
\[
- \nabla \cdot [\sigma(x, z) \nabla \phi(x, y, z)] = I \delta(x - x_j) \delta(y) \delta(z),
\]
where \( \delta(*) \) is the Dirac delta and \( \nabla \cdot \) is the gradient vector operator. A Fourier transform in the \( y \) direction yields,
\[
- \nabla \cdot [\sigma(x, z) \nabla \hat{\phi}_k(x, z)] + k^2 \sigma(x, z) \hat{\phi}_k(x, z) = \frac{I}{2} \delta(x - x_j) \delta(z),
\]
The equation above is discretized by using an \( N \times M \) non-uniform rectangular grid. We evaluate the finite-difference solution in its interior domain of validity according to Dey and Morrison (1979).

\[
\begin{bmatrix}
C_{11} & C_{12} & 0 & C_{14} \\
C_{21} & C_{22} & C_{23} & 0 \\
0 & C_{32} & C_{33} & C_{34} \\
0 & 0 & C_{43} & C_{44}
\end{bmatrix}
\begin{bmatrix}
\phi_1 \\
\phi_2 \\
\phi_3 \\
\phi_4
\end{bmatrix}
=
\begin{bmatrix}
0 \\
0 \\
0 \\
0
\end{bmatrix}
\]

System to be solved,
\[
Ah = b
\]

Let us assume that the matrix \( A \) has a non-zero main diagonal principal \( \{a\} \) and non-zero co-diagonals \( \{b\} \) and \( \{c\} \) as follows:
\[
A_i = \begin{cases} 
  a_i, & |i - j| = 0, \\
  b_{ij}, & |i - j| = 1, \\
  c_{ij}, & |i - j| = m,
\end{cases}
\]
The preconditioned conjugate gradient algorithm (PCG)

The method is defined by the following iterative equations,
\[ h_{i+1} = h_i + \alpha v_i \]  
\[ v_{i+1} = A^T g_{i+1} + \beta v_i \]  

The matrix \(A^T\) represents the approximate inverse of matrix \(A\), \(g_{i+1} = -b + A h_{i+1}\) is the residual of the approximate solution \(h_{i+1}\), \(v_i\) is an auxiliary vector, \(\alpha\) is chosen such that \(g_{i+1}^T v_i = 0\), \(\beta\) ensures that \(v_i^T A v_i = 0\).

If \(A^T = A^T\) the method converges in a single step.
If \(A^T = I\) the PCG algorithm reduces to the CG algorithm, where the solution of a linear system of size \(n\) can be found in \(n\) steps (Kincaid and Cheney, 1996).

The PCG algorithm with incomplete Cholesky factorization (PCG-IC)

The triangular matrix \(L\) generated in the incomplete Cholesky (IC) decomposition of the matrix \(A\) preserves the same number of bands of \(A\); that is, the matrix \(L\) is constructed only with the corresponding bands \(A_{ij}\), \(b_i\) and \(c_i\), such that \(LL^T\) represent an approximation of \(A\), i.e., \(AA = LL^T\).

The term \(q_{i+1} = A^T g_{i+1}\), needed on eq. (2) can be obtained by solving with back substitutions the linear system
\[ LL^T q_{i+1} = g_{i+1} \]  

The IC factorization \(LL^T\) is performed only once, before the PCG iterations begin.

Steps of the PCG-IC algorithm

- IC decomposition of the matrix \(A\), \(A = LL^T\).
- Iterate with equations (1), (2), and (3) until \(\|g_{i+1}\| < TOL \times 10^{-15}\)

DO \(j = 1, \ldots\)
  - calculate \(\alpha\)
  - update \(h_i\) and \(g_{i+1}\)
  - IF \(\|g_{i+1}\| < TOL \times 10^{-15}\) STOP
  - solve eq. (3) for \(q_{i+1}\)
  - calculate \(\beta\)
  - update \(v_{i+1}\)
ENDDO

Modified incomplete Cholesky factorization (PCG-ICs)

The first modification introduced in PCG-IC algorithm is the normalization of the linear system
\[ A h = b \]

Let \(LL^T\) be the IC factorization of \(A\).

Pre-multiplying the original system by the diagonal matrix \(E\) formed by the inverse of the elements of the main diagonal of the matrix \(L\) yields,
\[ (E A E)(E^T h) = E b \]

\[ A h = b \]  

The second modification consists of the iterative evaluation of the vector \(q_{i+1}\) in eq. (3). The number \(d\) of steps of the PCG algorithm is directly related with the degree of approximation used in the solution of the system,
\[ A q_{i+1} = g_{i+1} \]  

Let \(A = \tilde{A} = L L^T\) and \(R_i = A_i - \tilde{A}\). We can write eq. (5) as
\[ (L L^T + R_i) q_{i+1} = g_{i+1} \]

\(R_i\) represents the residual matrix necessary to complete \(A_i\). By multiplying and reordering the terms of the equation above, we found an iterative formula to approximate the solution of eq. (5),
\[ L L^T q_{i+1, j+1} = g_{i+1} - R_i q_{i+1, j} \]

The equation above is initiated with \(q_{i+1, 0} = 0\). For a preconditioning of order 0 (\(n_i = 0\)), eq. (6) reduces to eq. (3), as in the PCG-IC algorithm.

We can evaluate \(R_i q_{i+1, j}\) as follows:
\[ R_i q_{i+1, j} = A_i q_{i+1, j} - L_i L^T q_{i+1, j} \]

This procedure fits in the framework of the Iterative Refinement (Kincaid and Cheney, 1996).

Steps of the PCG-ICs algorithm

- IC decomposition of the matrix \(A\), \(A = LL^T\).
- Setting up the normalized system \(L h_i = b, (eq. (4))\).
- Computation of the factor \(L = EL\) of the matrix \(A_i\), \(A_i = LL^T\).
- PCG solution of the normalized system by iterating equations (1), (2), and (6) until \(\|g_{i+1}\| < TOL \times 10^{-15}\)
- Evaluation of the solution \(h = E h_i\).
Modified incomplete Cholesky factorization with tridiagonal factorization (PCG-ICT)

Analogously to the former approach, the original system, \( A h = b \), is normalized to \( A h = b \), and then transformed to \( A h = b \) by collapsing the band \( \{ b \} \) of the matrix \( A \), in the main diagonal of the matrix \( A \). To collapse the band \( \{ b \} \) we use a factorization of the tridiagonal portion of the matrix as described further.

The resulting system, \( A h_i = b_i \) is solved with PCG-IC using the IC decomposition corresponding to the bands \( \{ a \} \) and \( \{ c \} \) of the matrix \( A \), as presented below.

Steps of the PCG-ICT algorithm with tridiagonal factorization

- Factorization of the tridiagonal portion of the matrix \( A \) resulting in \( F^T T_{ab} F = E \).

\[
E \quad \text{is diagonal,}
F \quad \text{is triangular, and}
T_{ab} \quad \text{is tridiagonal, formed by bands \( \{ a \} \) and \( \{ b \} \) of matrix \( A \).}
\]

- Normalization of the system \( E^{-1/2} F A h_i = b_i \) to the former algorithm, resulting in \( A h_i = b_i \).

- Factorization of the tridiagonal portion of the matrix \( A \), obtaining matrices \( F \), and \( E = I \) (represents the identity matrix),

\[
F^T T_{ab} F = I
\]

- Collapse of the band \( \{ b \} \) of the matrix \( A \), and transformation into the system \( A h_i = b_i \).

\[
(F^T A F_{1i})(F_{1i} h_i) = F_{1i} b_i
\]

\[
(I + F_{1i} C F_{1i})h_i = b_i
\]

(Note that matrix \( C \) has only the bands \( \{ c \} \) of the matrix \( A \).)

- IC decomposition of the tridiagonal portion associated to the bands \( \{a \} \) and \( \{ c \} \) of the matrix \( A \).

- PCG solution of the modified system, preconditioning with the factors \( L \), and \( L^T \), and performing iterations with equations (1), (2), and (6) until \( ||g|| < TOL \).

- Evaluation of the solution through \( h_i = F_{1i} h_i \) and \( h = E^{-1} h_i \).

Factorization of the tridiagonal matrix, \( T_{ab} \)

By using a Levinson-type algorithm a set of coefficients \( \{ f_i \} \) and \( \{ e_i \}; i = 1,...,n \) may be calculated such that the tridiagonal matrix may be factorized as,

\[
F^T T_{ab} F = E
\]

where \( F \) is a triangular matrix and \( E \) is diagonal.

Exemplifying for a system of order \( n = 4 \):

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & f_3 & 1 & 0 \\
0 & f_3 & f_2 & 1 \\
0 & f_3 & f_2 & f_1 \\
0 & f_3 & f_2 & f_1 & 0
\end{bmatrix}
= \begin{bmatrix}
e_1 & e_2 & e_3 & e_4 \\
e_1 & e_2 & e_3 & e_4 \\
e_1 & e_2 & e_3 & e_4 \\
e_1 & e_2 & e_3 & e_4 \\
e_1 & e_2 & e_3 & e_4 \\
e_1 & e_2 & e_3 & e_4
\end{bmatrix}
\]

The coefficients \( \{ f_i \} \) and \( \{ e_i \}; i = 1,...,n \) may be calculated recursively with 2(n-1) operations, by using the following algorithm,

\[
f_i = -\frac{e_i}{e_{i-1}} \quad \text{for } i = 1, \ldots, n - 1
\]

\[
e_{n+1} = e_n + f_n f_1
\]

Fast vector-matrix multiplication

During the iterations in the PCG-ICT algorithm we need to evaluate a vector to matrix multiplication,

\[
A v = p
\]

\[
(I + F_1 C F_1) v = p
\]

Because the matrix \( F \) has a very special structure one may verify that the product \( F_1(\bullet) \) and \( F_1(\bullet) \) may be evaluated using \( n \) operations only.

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & f_3 & 1 & 0 \\
0 & f_3 & f_2 & 1 \\
0 & f_3 & f_2 & f_1 & 0
\end{bmatrix}
\begin{bmatrix}
y_1 \\
y_2 \\
y_3 \\
y_4 \\
y_4
\end{bmatrix}
= \begin{bmatrix}
y_1 \\
y_2 \\
y_3 \\
y_4 \\
y_4
\end{bmatrix}
\]

\[
y_1 = x_1
y_{i+1} = y_i + f_i x_i + x_{i+1}
\]

\[
\begin{bmatrix}
1 & f_1 & f_2 & f_3 & f_4 \\
0 & 1 & f_3 & f_2 & f_3 \\
0 & 0 & 1 & f_3 & f_2 \\
0 & 0 & 0 & 1 & f_3 \\
0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
y_1 \\
y_2 \\
y_3 \\
y_4 \\
y_4
\end{bmatrix}
= \begin{bmatrix}
y_1 \\
y_2 \\
y_3 \\
y_4 \\
y_4
\end{bmatrix}
\]

\[
y_n = x_n
y_{n-i} = x_{n-i} + f_{n-i} y_{n-i+1}
\]

Considering that the matrix \( C \) has only the co-diagonals \( \{ c \} \), then the cost to evaluate eq. (8) is less than \( 4(n-1) \) operations.

Complexity analysis

Some expressions corresponding to the computational cost of the previous algorithm are presented below. The number of products, divisions, and square root operations were considered.

- \( n \) – dimension of the linear system,
- \( m \) – distance between bands \( \{ a \} \) and \( \{ c \} \),
- \( N_{\text{op}} \) – number of PCG iterations,
- \( k \) – number of iterations of equation (6).

The number of operations of the methods

- PCG-IC and PCG-IC were, respectively,
- \( \text{OP}_{\text{PCG-IC}} = 12n - 3m - 4 + (17n - 4m - 3) N_{\text{op}} \)
- \( \text{OP}_{\text{PCG-IC}} = 24n - 6m - 6 + n(15 + 9 n) - 4m(1 + n) - 2 - 4k \) \( N_{\text{op}} \) -

In particular, if the modified PCG-IC method uses \( n = 0 \) and \( n = 1 \) iterations, we find

- \( \text{OP}_{\text{PCG-IC}} = 24n - 6m - 6 + (24n - 8m - 6) N_{\text{op}} \)
Figure 3 compares the cost of the PCG-IC. The number of iterations is fixed on \( n_{\text{it}} = 50 \) and the system dimension increases from \( 10^4 \) to \( 10^6 \).

Figure 4 compares the cost of the PCG-IC, \( N = 10^5 \) and \( n_{\text{it}} \) increases. In both cases we take \( m' = n \). OP, OP_0, and OP are defined on the equations above. OP_0 represents the number of operations considering that the choice \( n_k = 1 \) reduces \( n_{\text{it}} \) by a factor of 1/2, as seen on Fig. 3.

Figure 5 shows the CPU time spent on a 2D electroresistivity modeling problem (Ferreira et al., 2005); the dimension of the linear systems were in the range of \( 10^4 \) to \( 5 \times 10^5 \). The cost of the PCG-IC method was smaller for \( n = 8 \times 10^5 \), approximately, while PCG-IC presents a lower computational cost for \( n = 8 \times 10^5 \).

Figure 6 shows the reduction of the number of iterations of the algorithm PCG with respect to the degree of approximation \( n_k \) used on equation (6). Number of iterations of PCG-IC decreases as the parameter \( n_k \) increases \( (n = 10^5 \text{ and } m = 10^5) \).

Acknowledgements:
The authors thank FINEP, FAPESP, PETROBRAS, CNPQ and CAPES.

References: