A Hybrid Kaczmarz - CG Algorithm for Inconsistent Systems arising in Image Reconstruction

Constantin Popa

"Ovidius" University of Constanta, Romania
Faculty of Mathematics and Computer Science

Computational Methods with Applications
August 19 - 25, 2007, Harrachov, Czech Republic

Supported by the Grant CEEEX 05-D11-25/2005
**CT fan beam scanning procedure**

**Fan beam scanning**

*Construction of $A$ and $b$*

$A$: an $M \times N$ matrix; $b$: measurements of $I_{input}$ and $I_{output}$

$M = \text{number of sources} \times \text{number of detectors}$ (left figure)

$N = \text{number of pixels}$ (right figure)

**Mathematical formulation:** $Ax = b, \ x = (x_1, \ldots, x_N)^T$

$x_j =$ ”value of the attenuation function on $P_j$”
Properties of $A$ and $b$

- $A: M \times N$ is rectangular, big, sparse, rank-deficient, $\exists! \; x_{LS}$
- in practice $b \notin R(A) \Rightarrow$, i.e. $Ax = b$ is inconsistent $\Rightarrow$
  \[
  \| Ax - b \| = \min \{ \| Az - b \|, z \in \mathbb{R}^N \} \quad (*)
  \]

- Tikhonov-like regularization
  \[
  \| Ax - b \|^2 + \delta^2 \| Lx \|^2 = \min! \Leftrightarrow \left\| \begin{bmatrix} A \\ \delta L \end{bmatrix} x - \begin{bmatrix} b \\ 0 \end{bmatrix} \right\|^2 = \min!
  \]
  still inconsistent (!), but $x_{LS}(L; \delta) \approx x_{LS}$ for an appropriate choice of $L$ and $\delta$

2D and 3D images provided by Siemens Medical Solutions Dept. Erlangen

during 2006: a master thesis in Constanta (Tiberius Duluman; implementation of algorithms)

Data for a common 3D rec. probl. : \( N = 256^3 = 16,777,216 \), \( M = 150 \text{ sources} \times 1024^2 \text{ detectors} = 157,286,400 \); the matrix \( A \) cannot be any more stored in the computer memory (not even in a compressed form !); an alternative - storing \( A \) on the disk and reading it (and/or \( A^T \)) in each iteration \( \Rightarrow \) too much time consuming!
Three requests have been formulated:

- $R_{regen}$: $A$ has to be regenerated row-by-row in each iteration; the computations in the iterations of the algorithm have to be "compatible" with such a procedure
- $R_{eff}$: the algorithm has to produce an efficient reconstruction
- $R_{inc}$: the algorithm has to be convergent in the inconsistent case

**Candidates**: Kaczmarz Extended (KE) and Generalized CG (CG)
The Generalized CG algorithm

Generalized Conjugate Gradient (CG(A; b; ⋅))

Let \( x^0 \in \mathbb{R}^n \), \( r^0 = b - Ax^0 \); for \( k = 0, 1, \ldots \) do

\[
\begin{align*}
    r^0 &= b - Ax^0, \\
p^0 &= A^T r^0, \\
    \alpha_k &= \| A^T r_k \|^2 / \| Ap^k \|^2, \\
x^{k+1} &= x^k + \alpha_k p^k, \\
r^{k+1} &= b - Ax^{k+1} = r^k - \alpha_k Ap^k, \\
    \beta_k &= \| A^T r_{k+1} \|^2 / \| A^T r^k \|^2, \\
p^{k+1} &= A^T r^{k+1} + \beta_k p^k.
\end{align*}
\]

Convergence: Kammerer and Nashed, SIAM N.A., 9(1972), 165-181

**Theorem**

For any \( x^0 \in \mathbb{R}^N \) the above sequence \( (x^k)_{k \geq 0} \) converges to an element of \( LSS(A; b) \). If \( x^0 = 0 \), then \( \lim_{k \to \infty} x^k = x_{LS} \). 

Constantin Popa

A Hybrid Kaczmarz - CG Algorithm
**Kaczmarz Extended Algorithm**

Let \( x^0 \in \mathbb{R}^N, y^0 = b \); for \( k = 0, 1, \ldots \) do

\[
\begin{align*}
y^{k+1} &= (\varphi_1 \circ \cdots \circ \varphi_N)(y^k) \\
b^{k+1} &= b - y^{k+1} \\
x^{k+1} &= (f_1 \circ \cdots \circ f_M)(b^{k+1}; x^k),
\end{align*}
\]

\[
\varphi_j(y) = y - \frac{\langle y, A^j \rangle}{\|A^j\|^2} A^j, \quad f_i(\beta; x) = x - \frac{\langle x, A_i \rangle - \beta_i}{\|A_i\|^2} A_i
\]


**Theorem**

*Suppose that \( A_i \neq 0, A^j \neq 0, \forall i, j \). Then, for any \( x^0 \in \mathbb{R}^N \) the above sequence \((x^k)_{k \geq 0}\) converges to an element of \( LSS(A; b) \). If \( x^0 = 0 \), then \( \lim_{k \to \infty} x^k = x_{LS} \).*
KE - (very) good results in image reconstruction (belongs to the class of ART methods)

CG - poor results in image reconstruction; comments by prof. Gabor T. Herman (personal communication): *The CG works as advertised: it gets very near to the stated minimizer in an often surprisingly few iterations. However, the results tend to be worse (from the image reconstruction from projections point of view) than those produced by ART. The problem is that the (even regularized) normal equations do not properly capture what seems to be important for image reconstruction; as opposed to this, early iterative steps of ART seem to do that. This is due to some magic that I have never succeeded to understand.*
for (very) big dimensions $M$ and $N$, the matrix $A$ has to be regenerated (row-by-row) in each CG or KE iteration.

CG and KE use in each iteration beside $A$, $A^T$ but differently:

1. for CG: compute $v = A^T y$ by
   
   $v = 0$
   
   for $i = 1 : M$
   
   generate the $i$-th row $A_i$
   
   $u_i = \langle x, A_i \rangle$
   
   $v = v + y_i \cdot A_i$
   
   endfor

2. for KE: for $j = 1 : N$
   
   $aux = \frac{\langle y, A^j \rangle}{\|A^j\|^2}$
   
   $y = y - aux \cdot A^j$
   
   endfor

Unfortunately, formulas (2) request that we regenerate all the $M$ rows of the matrix $A$ for each $j = 1, \ldots, N$ in each KE iteration, which is a huge computational effort!!!
Some experiments

Figure: Image $x^\text{exact}$ (1024 × 1024)

Image provided by J. Hornegger and M. Prümmer, University of Erlangen-Nürnberg, Germany; tests on Intel P IV, 3.00GHz, 1GB RAM
Implementation details

- For CG: the rows of $A$ have been generated 2 times in each iteration
- For KE: $A$ and $A^T$ have been stored in a compressed form on the disk and then read in each iteration
- The total execution time has been measured
- simulation (consistent case): $b = Ax^{\text{exact}}$, then $Ax = b$ was solved for $x_{LS}$
Consistent case - underdetermined

490 sources $\times$ 2048 detectors, 10 iterations

$M = 490 \times 2048 = 1.003.520$, $N = 1024 \times 1024 = 1.048.576$

**CG - Time: 55 min.**

**KE - Time: 625 min.**
Consistent case - overdetermined

1140 sources × 2048 detectors, 10 iterations
M = 1140 × 2048 = 2,334,720, N = 1024 × 1024 = 1,048,576

CG - Time: 130 min.
KE - Time: 1155 min.
## KE and CG versus requests

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>$R_{\text{regen}}$</th>
<th>$R_{\text{eff}}$</th>
<th>$R_{\text{inc}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>KE</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>CG</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>KE+CG=KECG</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>
The KECG algorithm

Algorithm KECG Let $x^0 \in \mathbb{R}^N, y^0 = b$; for $k = 0, 1, \ldots$ do

\begin{align*}
y^{k+1} &= CG(A^T; 0; y^k) \quad (**) \\
b^{k+1} &= b - y^{k+1} \\
x^{k+1} &= (f_1 \circ \cdots \circ f_M)(b^{k+1}; x^k)
\end{align*}

Note. (**) means CG applied to $A^T y = 0$, with $y^0 = b$ which gives $\lim_{k \to \infty} y^k = P_{N(A^T)}(b)$.

Theorem

(C. Popa, 2007) Suppose that $A_i \neq 0, i = 1, \ldots, M$. Then, for $x^0 = 0$ the sequence $(x^k)_{k \geq 0}$ generated with the KECG algorithm converges to $x_{LS}$.
490 sources \times 2048 detectors, 10 iterations

for KECG the rows of $A$ have been regenerated 3 times per iteration

$M = 490 \times 2048 = 1,003,520$, $N = 1024 \times 1024 = 1,048,576$

CG - 55 min.
KE - 625 min.
KECG - 97 min.
Consistent case overdetermined)

1140 sources × 2048 detectors, 10 iterations

\[ M = 1140 \times 2048 = 2.334.720, \quad N = 1024 \times 1024 = 1.048.576 \]

CG - 130 min.
KE - 1155 min.
KECG - 225 min.
The consistent system $b^0 = A x^{\text{exact}}$ was perturbed as

$$b = b^0 + \text{pert}, \quad \text{pert} = 10\% \cdot \| b^0 \| \cdot \text{rand},$$

with $\text{rand}$ a randomly generated vector with unitary norm. Then, the least squares problem $\| A x - b \| = \text{min}$! was solved by regularization

$$\| A x - b \|^2 + \delta^2 \| L x \|^2 = \text{min}! \iff \left\| \begin{bmatrix} A \\ \delta L \end{bmatrix} x - \begin{bmatrix} b \\ 0 \end{bmatrix} \right\|^2 = \text{min}!$$
For each $i \in 1 \ldots n$, let $H_i$ be the set of horizontally neighbour pixels of $i$, $V_i$ the set of vertically neighbour pixels and $D_i$ the set of diagonally neighbour pixels of $i$. For each $j \in 1 \ldots n$,

$$(L)_{ij} = \begin{cases} (L)_{ij} = w_h, & \text{if } j \in H_i \\ (L)_{ij} = w_v, & \text{if } j \in V_i \\ (L)_{ij} = w_d, & \text{if } j \in D_i \\ \sum_{k=1}^{n} |(L)_{ik}|, & \text{if } j = i \text{ and } k \neq i \\ 0, & \text{otherwise} \end{cases}$$

Note. In our experiments we used $w_h = -1, w_v = -1, w_d = -1/\sqrt{2}$ and $\delta^2 = 20$; the matrix $L$ from (2) is positive semidefinite.
Inconsistent + regularization

10 KECG iterations

Underdetermined: 103 min.  Overdetermined: 237 min.
Acknowledgements. The computer programmes for the numerical experiments have been made by my Master student in Constanta, Tiberius Duluman.

Conclusions. We designed a hybrid algorithm - KECG - which combines in an efficient way both good parts of its components:

- the good quality of the reconstructed image, specific to KE
- the possibility to regenerate (only three times) in each KECG iteration the rows of the matrix $A$, specific to CG

Future work. Improving the implementation of the KECG reconstruction algorithm (essentially the scanning procedure and computation of the coefficients in the rows of $A$)