Comparisons of numerical experiments about GRNMM methods

Shaohua Cheng* and Litao Zhang

Department of Mathematics and Physics, Zhengzhou Institute of Aeronautical Industry Management, Zhengzhou, Henan, 450015, P. R. China

Received 2 December 2014, www.cmnt.lv

Abstract

Based on the global relaxed non-stationary multisplitting multi-parameter (GRNMM) methods, we give comparisons of numerical experiments about GRNMM methods and show the efficiency of GRNMM methods associated with TOR multisplitting for solving a large sparse linear system whose coefficient matrix is an H-matrix.

Keywords: Relaxed non-stationary multisplitting multi-parameter method; Parallel multisplitting; Global relaxed method; H-matrix

1 Introduction

For solving the large sparse linear system

\[ Ax = b, \]  

where \( A \in \mathbb{R}^{N \times N} \) is a square nonsingular H-matrix and \( x, b \in \mathbb{R}^N \), an iterative method is usually considered. The concept of multisplitting for the parallel solution of linear system was introduced by O’Leary and White [1] and further studied by many authors [1-23]. The multisplitting method can be thought of as an extension and parallel generalization of the classical block Jacobi method [4]. In this paper, we give comparisons of numerical experiments about GRNMM methods and show the efficiency of GRNMM methods associated with TOR multisplitting for solving a large sparse linear system whose coefficient matrix is an H-matrix.

2 GRNMM method

Global Relaxed Non-stationary Multisplitting Multi-parameter TOR method (GRNMM-TOR) was proposed in [23]. The algorithm is as follows.

Algorithm 1 (GRNMM-TOR)

Given the initial vector

For \( m = 0, 1, \ldots \), repeat (I) and (II), until convergence.

In \( k \) processors, \( k = 1, \ldots, \alpha \), let \( y_k^{(0)} = x^{(m)} \).

(I) For \( i = 1, 2, \ldots, q(m, k) \), (parallel) solving \( y_k^{(i)} \).

\[ [D - (\alpha_k L_k + \beta_k F_k)]y_k^{(i)} = [(1 - \gamma_k) D + (\gamma_k - \alpha_k) L_k + (\gamma_k - \beta_k) F_k + \gamma_k U_k]y_k^{(i-1)} + \gamma_k b. \]

(II) Computing

\[ x^{(m+1)} = \omega \sum_{k=1}^{\alpha} E_k y_k^{(q(m,k))} + (1 - \omega) x^{(m)}. \]

The Algorithm 1 can be written as:

\[ x^{(m+1)} = H^{(m)}(\alpha_1, \beta_1, \gamma_1, \omega) x^{(m)} + G^{(m)}(\alpha_1, \beta_1, \gamma_1, \omega) b, \quad m = 0, 1, \ldots, \]

where

* Corresponding author’s E-mail: shcheng@zzia.edu.cn
3 Numerical comparisons

In this section, we present some numerical experiments which compare the performance of GRNMM-TOR methods with LRNMM-AOR and LRNMM-SOR methods, and numerical experiments achieve effective improvement compared with the methods in [6,10]. By using difference discretization of partial different equation, we can obtain the corresponding coefficient matrix form of the linear system \((n=6)\), which is as follows.

\[
H^{(m)}(\alpha_k, \beta_k, \gamma_k, \omega) = \sum_{k=1}^{a} \sum_{i=1}^{\gamma_{(m,k)}-1} E_k \left[ D - (\alpha_k L_k + \beta_k F_k) \right]^{-1} [(1 - \gamma_k)D + \gamma_k U_k F_k + (1 - \omega)I],
\]

\[
G^{(m)}(\alpha_k, \beta_k, \gamma_k) = \omega \sum_{k=1}^{a} \sum_{i=1}^{\gamma_{(m,k)}-1} \left[ D - (\alpha_k L_k + \beta_k F_k) \right]^{-1} [(1 - \gamma_k)D + \gamma_k U_k F_k + (1 - \omega)I] \times \left[ D - (\alpha_k L_k + \beta_k F_k) \right]^{-1} \gamma_k.
\]

\[
A = \begin{bmatrix}
4 & -1 & 0 & 0 & 0 & 0 \\
2 & 1.5 & 0 & 0 & 0 & 0 \\
0 & 1 & 3 & -1 & 0 & 0 \\
0 & 0 & 1.5 & 2 & 2 & 0 \\
0 & 0 & 0 & 1 & 4 & -1 \\
0 & 0 & 0 & 0 & 2 & 2
\end{bmatrix}, \quad b = \begin{bmatrix}
3 \\
5.5 \\
3 \\
5.5 \\
4 \\
4
\end{bmatrix},
\]

\[
E_1 = diag(1, 1, 0, 0, 0, 0), \quad E_2 = diag(0, 0, 1, 1, 0, 0), \quad E_3 = diag(0, 0, 0, 1, 1).
\]

\[
L_1 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0.6 & 0 & 0 & 0 & 0 & 0 \\
0 & 1.5 & 0 & 0 & 0 & 0 \\
0 & 0 & -1.2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.3 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.8 & 0
\end{bmatrix}, \quad L_2 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
-1.5 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.5 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.7 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.5 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.4 & 0
\end{bmatrix},
\]

\[
L_3 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0.6 & 0 & 0 & 0 & 0 & 0 \\
0 & 1.5 & 0 & 0 & 0 & 0 \\
0 & 0 & -1.2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.3 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.8 & 0
\end{bmatrix}, \quad U_k = D - L_k - A, k = 1, 2, 3.
\]

\[
L_1 = \hat{L}_k + \hat{F}_k = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0.3 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.7 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.9 & 0 & 0 \\
0 & 0 & 0 & 0 & -0.5 & 0
\end{bmatrix}, \quad + \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0.7 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.3 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.1 & 0 & 0 \\
0 & 0 & 0 & 0 & -0.3 & 0
\end{bmatrix},
\]

\[
L_2 = \hat{L}_k + \hat{F}_k = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0.3 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.3 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.4 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.2 & 0
\end{bmatrix}, \quad + \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0.7 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.3 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.2 & 0
\end{bmatrix}.
\]
Let the initial guess and the tolerance be $\chi^{(0)} = (10, 30, -20, -40, -8, 9)^T$ and $\varepsilon = 10^{-10}$, respectively. By numerical experiments, the results of performance improvements with GRNMM-TOR method and LRNMM-SOR method are shown in Table 1. Furthermore, $\rho_{opt}$ and $ite_{opt}$ denote spectral radius of approximate optimization and iterative numbers of approximate optimization, respectively. The improvements percentage $\%$ are obtained from $1 - \frac{ite_{opt}(Re)}{T_{opt(this paper)}}$. Similarly, the performance improvements results with GRNMM-TOR method and LRNMM-AOR method are shown in Table 2. Let the initial guess and the tolerance be $\chi^{(0)} = (0, 10, 20, 30, 30)^T$ and $\varepsilon = 10^{-10}$, respectively. The performance improvements results with GRNMM-TOR method and LRNMM-SOR method are shown in Table 3. Similarly, the performance improvements results with GRNMM-TOR method and LRNMM-AOR method are shown in Table 4.

### Table 1: Comparison of improvements percentage

<table>
<thead>
<tr>
<th>method</th>
<th>$\rho_{opt}$</th>
<th>$ite_{opt}$</th>
<th>improvements $%$</th>
<th>Ref</th>
</tr>
</thead>
<tbody>
<tr>
<td>LRNMM-SOR</td>
<td>0.6433</td>
<td>62</td>
<td>0</td>
<td>[16]</td>
</tr>
<tr>
<td>GRNMM-TOR</td>
<td>0.5705</td>
<td>49</td>
<td>20.97%</td>
<td>this paper</td>
</tr>
</tbody>
</table>

### Table 2: Comparison of improvements percentage

<table>
<thead>
<tr>
<th>method</th>
<th>$\rho_{opt}$</th>
<th>$ite_{opt}$</th>
<th>improvements $%$</th>
<th>Ref</th>
</tr>
</thead>
<tbody>
<tr>
<td>LRNMM-AOR</td>
<td>0.6006</td>
<td>53</td>
<td>0</td>
<td>[10]</td>
</tr>
<tr>
<td>GRNMM-TOR</td>
<td>0.5705</td>
<td>49</td>
<td>7.55%</td>
<td>this paper</td>
</tr>
</tbody>
</table>

### Table 3: Comparison of improvements percentage

<table>
<thead>
<tr>
<th>method</th>
<th>$\rho_{opt}$</th>
<th>$ite_{opt}$</th>
<th>improvements $%$</th>
<th>Ref</th>
</tr>
</thead>
<tbody>
<tr>
<td>LRNMM-SOR</td>
<td>0.6433</td>
<td>62</td>
<td>0</td>
<td>[16]</td>
</tr>
<tr>
<td>GRNMM-TOR</td>
<td>0.5705</td>
<td>50</td>
<td>19.35%</td>
<td>this paper</td>
</tr>
</tbody>
</table>

### Table 4: Comparison of improvements percentage

<table>
<thead>
<tr>
<th>method</th>
<th>$\rho_{opt}$</th>
<th>$ite_{opt}$</th>
<th>improvements $%$</th>
<th>Ref</th>
</tr>
</thead>
<tbody>
<tr>
<td>LRNMM-AOR</td>
<td>0.6006</td>
<td>55</td>
<td>0</td>
<td>[10]</td>
</tr>
<tr>
<td>GRNMM-TOR</td>
<td>0.5705</td>
<td>50</td>
<td>9.09%</td>
<td>this paper</td>
</tr>
</tbody>
</table>

In Figure 1, we show the detailed comparison of residual norm decline about three methods. From Figure 1, we may see clearly that GRNMM-TOR method can achieve much faster convergent speed than LRNMM-AOR method and LRNMM-SOR method.

Remark 3.1 The above numerical experiments indicate: By using our methods, we really achieve effective improvement compared with LRNMM-AOR method and LRNMM-SOR method. When comparing with LRNMM-SOR method and LRNMM-AOR method, the number of iterations for convergence of GRNMM-TOR method improved 20% and 10%, which the tolerance for convergence is residual norm less than $\varepsilon = 10^{-10}$.
4 Conclusions

In this paper, based on the global relaxed non-stationary multisplitting multi-parameter TOR iterative methods for solving linear systems of algebraic equations \( Ax = b \), we show the efficiency of GRNMM methods associated with TOR multisplitting for solving a large sparse linear system whose coefficient matrix is an H-matrix. Furthermore, efficiency of the global relaxed non-stationary multisplitting multi-parameter methods are shown by numerical experiments. Numerical experiments show that when choosing the approximately optimal relaxed parameters, GRNMM-TOR methods have faster convergent rate compared with LRNMM-AOR method and LRNMM-SOR method. Further performance improvement, one can consider how to choose the approximately optimal relaxed parameters to reduce the cost of choosing the relaxed parameters and improve performance strongly.

Acknowledgments

This research of this author is supported by NSFC Tianyuan Mathematics Youth Fund (11226337), NSFC(61203179, 61202098, 61170309, 91130024, 61033009, 61272544,61472462 and 11171039), Aeronautical Science Foundation of China (2013ZDS55006), Project of Youth Backbone Teachers of Colleges and Universities in Henan Province(2013GGJS-142), ZZIA Innovation team fund (2014TD02), Major project of development foundation of science and technology of CAEP (2012A0202008), Defense Industrial Technology Development Program, Basic and Advanced Technological Research Project of of Henan Province (122300410181, 132300410373, 142300410333), Natural Science Foundation of Henan Province (13A110399,14A630019,14B110023), Natural Science Foundation of Zhengzhou City (141PQYJS560).

References


[22]Zhang L-T 2014 A new preconditioner for generalized saddle matrices with highly singular (1,1) blocks International Journal of Computer Mathematics 91(9) 2091-101


Authors

Cheng Shaohua, born in December, 1963, Zhengzhou city, P.R. China

Current position, grades: Professor China of Zhengzhou Institute of Aeronautical Industry Management.
University studies: M.Sc. of College of Science from Nanjing University of Aeronautics and Astronautics in China.
Scientific interest: Parallel Computing, Iterative methods, Linear systems
Experience: teaching experience of 31 years, 5 scientific research projects.

Zhang Litaot, born in March, 1980, Zhengzhou city, P.R. China

Current position, grades: Associate Professor China of Zhengzhou Institute of Aeronautical Industry Management.
University studies: B.Sc. of School of Mathematical Sciences from University of Electronic Science and Technology of China in China. M.Sc. of College of Mathematics and Information Science from Henan Normal University in China.
Scientific interest: Parallel Computing, Preconditioning, Iterative methods, Linear systems
Publications: more than 50 papers published in various journals.
Experience: teaching experience of 5 years, 6 scientific research projects.