46th Annual Iranian Mathematics Conference
25-28 August 2015
Yazd University

Talk Complete pivoting strategy to compute the IULBF preconditioner pp.: 1–4

Complete pivoting strategy to compute the IULBF preconditioner

A. Rafiei*
Hakim Sabzevari University

Fatemeh Rezaei Fazel†
Hakim Sabzevari University

Abstract

In this paper, a complete pivoting strategy to compute the IULBF preconditioner is presented.

Keywords: pivoting, IULBF preconditioner.
Mathematics Subject Classification [2010]: 65F10, 65F50, 65F08.

1 Introduction

Consider the linear system of equations of the form $Ax = b$, where the coefficient matrix $A \in \mathbb{R}^{n\times n}$ is nonsingular, large, sparse and nonsymmetric and also $x, b \in \mathbb{R}^n$. An IUL preconditioner M for this system is in the form of $M = UDL \approx A$. This preconditioner will change the original system to the left preconditioned system $M^{-1}Ax = M^{-1}b$. For a proper preconditioner, instead of solving the original system, it is better to solve the left preconditioned system by the Krylov subspace methods [4]. In [1, 2], we have proposed an IUL preconditioner for system $Ax = b$. This preconditioner is termed the $IULBF$.

Algorithm 1 (IULBF preconditioner)

Input: $A \in \mathbb{R}^{n\times n}$ and $\tau_z, \tau_w, \tau_l, \tau_u \in (0, 1)$ be drop tolerances parameters.
Output: $A \approx UDL$

1. **for** $i = n$ to 1 **do**
2. $w_i^{(0)} = e_i^T, \quad z_i^{(0)} = e_i.$
3. **for** $j = i + 1$ to n **do**
4. $p_j^{(i-1)} = e_i^T A z_j^{(n-j)} \quad q_j^{(i-1)} = w_j^{(n-j)} A e_i$
5. $U_{ij} = \dfrac{p_j^{(i-1)}}{d_{jj}}, \quad L_{ji} = \dfrac{q_j^{(i-1)}}{d_{jj}}$
6. If $|L_{ji}| < \tau_l$, then set $L_{ji} = 0$. Also if $|U_{ij}| < \tau_u$, then set $U_{ij} = 0$
7. $z_i^{(j-i)} = z_i^{(j-i-1)} - \dfrac{q_j^{(i-1)}}{d_{jj}} z_j^{(n-j)}, \quad w_i^{(j-i)} = w_i^{(j-i-1)} - \dfrac{p_j^{(i-1)}}{d_{jj}} w_j^{(n-j)}$
8. For all $l \geq j$, if $|z_{li}^{(j-i)}| < \tau_z$ and $|w_{il}^{(j-i)}| < \tau_w$, then set $z_{li}^{(j-i)} = 0$ and $w_{il}^{(j-i)} = 0$
9. **end for**
10. $d_{ii} = w_i^{(n-i)} A e_i$
11. **end for**
12. Return $U = (U_{ij})_{1\leq i,j\leq n}$, $D = diag(d_{ii})_{1\leq i\leq n}$ and $L = (L_{ji})_{1\leq j,i\leq n}$.

Algorithm 1, computes the $IULBF$ preconditioner. In this algorithm, matrices L and U are computed column-wise and row wise, respectively.

*rafiei.am@gmail.com, a.rafiei@hsu.ac.ir.
†Speaker, rezaeefazel@gmail.com

2 Pivoting strategy for the IULBF preconditioner

Algorithm 2, computes the IULBF preconditioner which is coupled with complete pivoting strategy. The pivoting strategy of this algorithm is based on the complete pivoting strategy of the Backward IJK version of Gaussian Elimination process. In lines 16 and 35 of this algorithm we use the parameter $\alpha \in (0, 1]$ to control the pivoting process.

Algorithm 2 ($IULBF$ preconditioner coupled with complete pivoting strategy)

Input: Let $A \in \mathbb{R}^{n\times n}$, $U = L = \Pi = \Sigma = I_n$, $\tau_z, \tau_w, \tau_l, \tau_u \in (0, 1)$ be drop tolerances and prescribe a pivoting tolerace $\alpha \in (0, 1]$.
Output: $\Pi A\Sigma \approx UDL$.

1. **for** $i = n$ to 1 **do**
2. $m_i = n_i = iter = 0$
3. $satisfied\_\, p = satisfied\_\, q = false$
4. **while** not $satisfied\_\, p$ **do**
5. $iter = iter + 1$
6. $z_i^{(0)} = e_i$
7. **for** $j = i + 1$ to n **do**
8. $q_j^{(i-1)} = w_j^{(n-j)}(\Pi A\Sigma)e_i$
9. $z_i^{(j-i)} = z_i^{(j-i-1)} - (\frac{q_j^{(i-1)}}{d_{jj}})z_j^{(n-j)}$
10. For all $l \geq j$, if $|z_{li}^{(j-i)}| < \tau_z$, then set $z_{li}^{(j-i)} = 0$.
11. **end for**
12. If $iter = 1$, then set $p_i^{(i-1)} = e_i^T(\Pi A\Sigma)z_i^{(n-i)}$. Otherwise set $p_i^{(i-1)} = q_i^{(i-1)}$
13. **for** $j = i - 1$ to 1 **do**
14. $p_i^{(j-1)} = e_j^T(\Pi A\Sigma)z_i^{(n-i)}$
15. **end for**
16. **if** $|p_i^{(i-1)}| < \alpha\ max_{m\leq i}|p_i^{(m-1)}|$ **then**
17. $m_i = m_i + 1, \quad \pi_{m_i}^{(i)} = I_n$.
18. $satisfied\_\, q = false$
19. Choose k such that $|p_i^{(k-1)}| = max_{m\leq i}|p_i^{(m-1)}|$.
20. interchange the rows i and k of $\pi_{m_i}^{(i)}$ and the elements $p_i^{(i-1)}$ and $p_i^{(k-1)}$
21. $\Pi = \pi_{m_i}^{(i)}\Pi$
22. **end if**
23. $satisfied\_\, p = true$
24. **if** not $satisfied\_\, q$ **then**
25. $w_i^{(0)} = e_i^T$
26. **for** $j = i + 1$ to n **do**
27. $p_j^{(i-1)} = e_i^T(\Pi A\Sigma)z_j^{(n-j)}$
28. $w_i^{(j-i)} = w_i^{(j-i-1)} - (\frac{p_j^{(i-1)}}{d_{jj}})w_j^{(n-j)}$
29. For all $l \geq j$, if $|w_{il}^{(j-i)}| < \tau_w$, then set $w_{il}^{(j-i)} = 0$.
30. **end for**
31. $q_i^{(i-1)} = p_i^{(i-1)}$
32. **for** $j = i - 1$ to 1 **do**
33. $q_i^{(j-1)} = w_i^{(n-i)}(\Pi A\Sigma)e_j$
34. **end for**
35. **if** $|q_i^{(i-1)}| < \alpha\ max_{m\leq i}|q_i^{(m-1)}|$ **then**
36. $n_i = n_i + 1, \quad \sigma_{n_i}^{(i)} = I_n$
37. $satisfied\_\, p = false$
38. Choose l such that $|q_i^{(l-1)}| = \max_{m\leq i}|q_i^{(m-1)}|$.
39. interchange the columns i and l of $\sigma_{n_i}^{(i)}$ and the elements $q_i^{(i-1)}$ and $q_i^{(l-1)}$
40. $\Sigma = \Sigma\sigma_{n_i}^{(i)}$
41. **end if**
42. $satisfied\_\, q = true$
43. **end if**
44. **end while**
45. $d_{ii} = p_i^{(i-1)}$
46. **for** $j = i + 1$ to n **do**
47. $L_{ji} = \frac{q_j^{(i-1)}}{d_{jj}}, \quad U_{ij} = \frac{p_j^{(i-1)}}{d_{jj}}$
48. If $|L_{ji}| < \tau_l$, then set $L_{ji} = 0$. Also if $|U_{ij}| < \tau_u$, then set $U_{ij} = 0$.
49. **end for**
50. **end for**
51. Return $L = (L_{ji})_{1\leq j,i\leq n}$, $D = diag(d_{ii})_{1\leq i\leq n}$, $U = (U_{ij})_{1\leq i,j\leq n}$, Π and Σ.

3 Numerical results

In this section, we have considered 8 artificial linear systems where the coefficient matrices are downloded from [3] and the exact solution of these systems is the vector $[1, \cdots, 1]^T$. We have used two parameters 0.75 and 1.0 as α to compute the $IULBF$ preconditioner with complete pivoting strategy. We have used the command $GMRES$ in Matlab software to solve the original and the left precoditioner systems. We have used 10 as the number of restarts for the $GMRES$ method. The stopping criterion for all linear systems is satisfied when the relative residual is less than 10^{-6}. We have considered the zero vector as the initial solution for all linear systems. The density of all preconditioners is defined as:

$$density = \frac{nnz(L) + nnz(U)}{nnz(A)},$$

where $nnz(L)$, $nnz(U)$ and $nnz(A)$ refer to the number of nonzero entrries of matrices L, U and A, respectively. To compute all of the precoditioners we have considered all of the drop tolerance parameters equal to 0.1 .
Table 1, shows the matrix properties and the information of $GMRES$ method to solve the original linear systems. In this table, n and nnz are the dimension and the number of nonzero entries of the matrix.

Table 1: matrix properties and information of the $GMRES(10)$ method

Matrix	n	nnz	without preconditioner			
			$outer$	$inner$	$flag$	$Itime$
$bfwa62$	62	450	161	2	0	0.5252
$tub100$	100	396	724	10	1	12.5472
$bwm200$	200	796	5000	10	1	14.3536
$saylr1$	238	1128	5000	10	1	13.2154
$cage7$	340	4380	2	8	0	0.0134
$tols340$	340	2196	3881	10	1	13.5088
$bfwb398$	398	1654	3	9	0	0.0778
$olm500$	500	1996	4023	10	1	9.0694

In all the tables, the parameters $outer$, $inner$ and $flag$ indicate the $outer$ iterations, the $inner$ iterations and the status of the convergence for $GMRES(10)$ method.

Table 2: properties of the IULBF preconditioner

Method	IULBF			
Matrix	$density$	$outer$	$inner$	$flag$
$bfwa62$	0.9111	2	8	0
$tub100$	1.0051	1	10	0
$bwm200$	1	4	4	0
$saylr1$	0.9592	4	7	0
$cage7$	0.4841	1	8	0
$tols340$	0.9039	2	10	0
$bfwb398$	0.8368	1	6	0
$olm500$	1.1839	4	7	0

In Tables 1 – 3, when $flag$ is equal to 0, it means that the method has been converged to the desired tolerance within the 2500 outer iterations. $flag = 1$ shows that we can not obtain the convergence in 2500 number of iterations.

Table 3: properties of the IULBFP(0.75) and IULBFP(1.0) preconditioners

Method	IULBFP(0.75)						IULBFP(1.0))					
Matrix	*density*	*Rpiv*	*Cpiv*	*outer*	*inner*	*flag*	*density*	*Rpiv*	*Cpiv*	*outer*	*inner*	*flag*
bfwa62	0.9022	3	2	2	8	0	0.9000	4	4	2	8	0
tub100	1.0657	24	22	1	10	0	1.1086	23	23	1	9	0
bwm200	1.1131	51	45	12	9	0	1.1256	51	51	10	2	0
saylr1	0.9592	0	0	4	7	0	0.9592	0	0	4	7	0
cage7	0.4780	0	0	1	8	0	0.4780	0	0	1	8	0
tols340	0.3679	37	76	1	7	0	0.3657	40	77	1	7	0
bfwb398	0.8368	0	0	1	6	0	0.8368	0	0	1	6	0
olm500	0.9965	499	249	3	6	0	0.9965	499	249	3	6	0

In Tabe 3, notation $IULBFP(\alpha)$ refers to the $IULBF$ preconditioner with complete pivoting strategy which is computed by the parameter α. The columns $Rpiv$ and $Cpiv$ show the total number of row and column pivoting. In Tables 2 and 3, the information in the columns $flag$, $outer$ and $inner$ associated to the three preconditioners indicate that for all of the matrices, one of the preconditioners $IULBFP(1.0)$ or $IULBFP(0.75)$ gives better results of the $GMRES(10)$ method than the $IULBF$ preconditioner. This means that the complete pivoting strategy with one of the values $\alpha = 1.0$ or $\alpha = 0.75$ has a good effect on the quality of the $IULBF$ preconditioner.
If we compare the columns $flag$, $outer$ and $inner$ in Table 2 by the columns $flag$, $outer$ and $inner$ of Table 1, then it is clear that the two preconditioners $IULBFP(1.0)$ and $IULBFP(0.75)$ are useful tools to decrease the number of iterations of the $GMRES(10)$ method.

References

[1] A. Rafiei, *ILU and IUL Factorizations Obtained From Forward and Backward Factored Approximate Inverse Algorithms*. Bulletin of the Iranian Mathematical Soiety. 40 (5), 1327-1346 (2014).

[2] A. Rafiei, F. Shahlaei, *Different Versions of ILU and IUL Factorizations obtained from Forward and Backward Factored Approximate Inverse Processes-Part I* . Advances in Numerical Analysis, Volume 2011, Article ID 703435, 9 pages.

[3] T. Davis, *University of Florida Sparse Matrix Collection.*, *http://www.cise.ufl.edu/research/sparse/matrices.*, Accessed 2015.

[4] Y. Saad, *Iterative Methods for Sparse Linear Systems*. PWS publishing, New York., (1996).