CALCULATING AN INVARIANT SUBSPACE OF DIAGONALLY DOMINANT MATRICES – PART II

Noah H. Rhee

University of Missouri – Kansas City

Abstract. In part I of this paper, a modification of the algorithm of Blevins and Stewart for calculating an invariant subspace of diagonally dominant matrices was given. Here, we will discuss the numerical aspects of this algorithm. In particular, numerical results show that the modified algorithm seems to converge faster than the original algorithm.

The notation and terminology we will use will be the same as in Part I [4] of the article.

4. Numerical Results. In practice $\epsilon = ||\mathbf{E}||_2$ is difficult to calculate. But we note that $\tilde{\epsilon} = ||E_{11}|| + ||E_{22}||$ is easy to compute and $\epsilon \leq \tilde{\epsilon}$. This can be seen by using [1] as follows. Using the definition of spectral norm and relation (9) we obtain

$$
\begin{aligned}
\epsilon \leq ||\mathbf{E}||_2 &= \sup_{||P||=1} ||\mathbf{E}P|| \\
&= \sup_{||P||=1} ||PE_{11} - E_{22}P|| \\
&\leq \sup_{||P||=1} (||P||||E_{11}||_2 + ||E_{22}||_2||P||) \\
&= ||E_{11}||_2 + ||E_{22}||_2 \\
&\leq ||E_{11}|| + ||E_{22}|| = \tilde{\epsilon}.
\end{aligned}
$$

Let

$$
\tilde{\Gamma} = \left\{ P \in \mathbb{R}^{(n-l)\times l} : ||P|| \leq \frac{2\gamma}{\delta - \tilde{\epsilon}} \right\}. \tag{27}
$$

Note that the following variant of Theorem 9 is also valid.

Theorem 13. Let $\eta = ||E_{12}||$, $\gamma = ||E_{21}||$, $\tilde{\epsilon} = ||E_{11}|| + ||E_{22}||$, and $\delta = ||\mathbf{D}^{-1}||_2^{-1}$. Then if

$$\delta - \tilde{\epsilon} > 2\sqrt{\eta\gamma}, \tag{28}$$

then, for any $P_0 \in \Gamma$, the sequence P_k, defined by

$$P_{k+1} = \Phi(P_k), \quad k = 0, 1, \dots,$$

converges to the unique solution P^* of equation (7) in $\tilde{\Gamma}$. Moreover,

$$||P^* - P_k|| \leq \frac{\tilde{\rho}}{1 - \tilde{\rho}} ||P_k - P_{k-1}||, \quad k = 1, 2, \dots, \tag{29}$$

and

$$||P_{k+1} - P_k|| \leq \tilde{\rho} ||P_k - P_{k-1}||, \quad k = 1, 2, \dots, \tag{30}$$

where

$$\tilde{\rho} = \frac{\tilde{\epsilon}}{\delta} + \frac{4\eta\gamma}{\delta(\delta - \tilde{\epsilon})} < 1. \tag{31}$$

Remark 14. Condition (28) is stronger than condition (18), since $\tilde{\epsilon} \geq \epsilon$. However, the advantage of Theorem 13 over Theorem 9 is that $\tilde{\epsilon}$ is easy to compute. Note that η and γ are easy to compute. Hence, if we know how to compute δ, we can check convergence criterion (28), and compute $\tilde{\rho}$ in relation (31). But, from relation (11), it is easy to see [1] that

$$\delta = ||\mathbf{D}^{-1}||^{-1} = \min \left\{ |d_j - d_i| : l + 1 \leq i \leq n, \ 1 \leq j \leq l \right\}. \tag{32}$$

Under condition (28), Algorithm 11 becomes, with a slight change in step (1)

Algorithm 15.
(1) $\overline{P}_0 = 0$ and $\overline{P}_1 = \Phi(\overline{P}_0)$
(2) For $k = 1, 2, \ldots,$
$\overline{P}_{k+1} = \Psi(\overline{P}_k)$.
If $||\overline{P}_{k+1}|| > 2\gamma/(\delta - \tilde{\epsilon})$ or $||\overline{P}_{k+1} - \overline{P}_k|| > \tilde{\rho}||\overline{P}_k - \overline{P}_{k-1}||$ $(k \geq 2)$,
go to (3);
otherwise
go to (2).
(3) For $j = k, k+1, \ldots,$
$\overline{P}_{j+1} = \Phi(\overline{P}_j)$.

Remark 16. The change in step (1) is due to the fact that $\Phi(0)$ is cheaper to compute than $\Psi(0)$.

Now for the given diagonally dominant matrix A with which the first l diagonal elements form a cluster that is well separated from the other diagonal elements, it is easy to check whether the condition (28) is satisfied or not. Once condition (28) is satisfied, we can locate an invariant subspace of A either by using Algorithm 5 or Algorithm 15. The iteration can be terminated with P_{k+1} $(\overline{P}_{k+1})$ if

$$||P_{k+1} - P_k||(||\overline{P}_{k+1} - \overline{P}_k||) \leq eps$$

in both algorithms, where eps is a prescribed error tolerance. Note that from relation (29) this implies

$$||P^* - P_{k+1}|| \leq \frac{\tilde{\rho}}{1 - \tilde{\rho}} eps.$$

How to use Algorithm 5 is straightforward. Now we briefly discuss the practical details in using Algorithm 15. There are two things that need to be explained; (i) how to form $\overline{P}_{k+1}$ from $\overline{P}_k$ by using the function Ψ in step (2) and (ii) how to check two conditions in step (2).

(i) To compute $\overline{P}_{k+1}$ from $\overline{P}_k$ we have to solve (see relation (23))

$$(\mathbf{D} + \mathbf{L})\overline{P}_{k+1} = E_{21} - \overline{P}_k E_{12} \overline{P}_k - \mathbf{U}\overline{P}_k.$$

Let

$$\overline{Q}_k = E_{21} - \overline{P}_k E_{12} \overline{P}_k - \mathbf{U}\overline{P}_k.$$

Then the above linear system becomes

$$(\mathbf{D}+\mathbf{L})\overline{P}_{k+1}=\overline{Q}_k.$$

Let $\overline{P}_{k+1}=(p_{ij})$, $\overline{Q}_k=(q_{ij})$, $l+1\le i\le n$, $1\le j\le l$. Then from equation (25) we can express p_{ij} by the following equations

$$p_{ij}=\frac{q_{ij}+\sum_{k=l+1}^{i-1}l_{ik}p_{kj}-\sum_{k=j+1}^{l}p_{ik}l_{kj}}{d_j-d_i},\quad l+1\le i\le n,\quad 1\le j\le l.$$

Here the empty sums are regarded as zero. By using the ordering defined in relation (26), the above equations can be solved by forward substitution, since the matrix representation of $\mathbf{L}+\mathbf{D}$ is a lower triangular matrix.

(ii) To check the terminating condition we need to form

$$\sigma_k=||\overline{P}_{k+1}-\overline{P}_k||,\quad k=0,1,\ldots.$$

To check the second condition in step (2), all we need to check is whether σ_k/σ_{k-1} is bigger than $\tilde{\rho}$ or not for $k=2,3,\ldots$. The first condition in step (2) can be checked by directly computing $||\overline{P}_{k+1}||$ and compare it with $2\gamma/(\delta-\tilde{\epsilon})$. However, a more economic way of checking this condition is to form $\sigma_0+\sigma_1+\cdots+\sigma_k$, and compare this with $2\gamma/(\delta-\tilde{\epsilon})$. Note that $||\overline{P}_{k+1}||\le\sigma_0+\sigma_1+\cdots+\sigma_k$. Hence, if

$$\sigma_0+\sigma_1+\cdots+\sigma_k>\frac{2\gamma}{\delta-\tilde{\epsilon}} \tag{33}$$

and

$$\sigma_0+\sigma_1+\cdots+\sigma_{k-1}\le\frac{2\gamma}{\delta-\tilde{\epsilon}}, \tag{34}$$

then $||\overline{P}_k||\le 2\gamma/(\delta-\tilde{\epsilon})$ and $||\overline{P}_{k+1}||$ may be greater than $2\gamma/(\delta-\tilde{\epsilon})$. So if conditions (33) and (34) are satisfied for $k=1,2,\ldots$, we go to step (3). We also mention that in using Algorithm 15 we stored A^T instead of A to make the program column oriented. Now we conclude this paper with several numerical examples, which demonstrate faster convergence of Algorithm 15.

5. Numerical Examples. We will denote Algorithm 5 by algorithm (**BS**) and Algorithm 15 by algorithm (**MBS**). In both algorithms *eps* was set to 10^{-15}. All computations were done on a VAX 6000 Model 500 using double precision arithmetic, that is, the machine precision of 1.11×10^{-16}.

Example 16. We generated a matrix A of order $n = 40$, whose off diagonal entries were determined by $(2 \text{ random } - 1) \times 0.025$, where random denotes a random number between 0 and 1. The diagonal entries were set $d_1 = d_2 = d_3 = 3$ and $d_i = i$, $4 \leq i \leq n$. l was chosen to be 3. Both algorithms were used to locate the invariant subspace spanned by eigenvectors whose eigenvalues are close to 3. The convergence condition (28) was satisfied. In fact, we have

$$\delta - \tilde{\epsilon} = 0.4272, \quad 2\sqrt{\eta\gamma} = 0.3101$$

$$\frac{2\gamma}{\delta - \tilde{\epsilon}} = 0.7579, \quad \tilde{\rho} = 0.7979$$

Then we obtained the following results:

	(**BS**)	(**MBS**)
k	$\|\|P_{k+1} - P_k\|\|$	$\|\|\overline{P}_{k+1} - \overline{P}_k\|\|$
0	$3.1075 \cdot 10^{-2}$	$3.1075 \cdot 10^{-2}$
1	$7.8481 \cdot 10^{-4}$	$7.8347 \cdot 10^{-4}$
2	$1.8771 \cdot 10^{-5}$	$9.5336 \cdot 10^{-6}$
3	$5.3214 \cdot 10^{-7}$	$2.2330 \cdot 10^{-7}$
4	$1.7232 \cdot 10^{-8}$	$4.5688 \cdot 10^{-9}$
5	$3.8518 \cdot 10^{-10}$	$1.0147 \cdot 10^{-10}$
6	$8.5870 \cdot 10^{-12}$	$8.5379 \cdot 10^{-13}$
7	$1.7331 \cdot 10^{-13}$	$4.7367 \cdot 10^{-15}$
8	$3.2774 \cdot 10^{-15}$	$1.4187 \cdot 10^{-17}$
9	$9.2689 \cdot 10^{-17}$	

We note that 1 iteration was saved by using algorithm (**MBS**).

Example 17. Let A be of order $n = 40$, whose upper (lower) off diagonal entries are uniformly 0.01 (0.02). The diagonal entries were set $d_1 = d_2 = d_3 = 3$ and $d_i = i$, $4 \le i \le n$. l was chosen to be 3. Both algorithms were used to locate the invariant subspace spanned by eigenvectors whose eigenvalues are close to 3. The convergence condition (28) was satisfied. In fact, we have

$$\delta - \tilde{\epsilon} = 0.3842, \quad 2\sqrt{\eta\gamma} = 0.2980$$

$$\frac{2\gamma}{\delta - \tilde{\epsilon}} = 1.0969, \quad \tilde{\rho} = 0.8469$$

Then we obtained the following results:

	(**BS**)	(**MBS**)
k	$\|\|P_{k+1} - P_k\|\|$	$\|\|\overline{P}_{k+1} - \overline{P}_k\|\|$
0	$4.4067 \cdot 10^{-2}$	$4.4067 \cdot 10^{-2}$
1	$1.3768 \cdot 10^{-3}$	$1.3574 \cdot 10^{-3}$
2	$8.4031 \cdot 10^{-5}$	$8.0349 \cdot 10^{-5}$
3	$3.4503 \cdot 10^{-6}$	$9.4730 \cdot 10^{-7}$
4	$1.8811 \cdot 10^{-7}$	$1.1611 \cdot 10^{-8}$
5	$9.0326 \cdot 10^{-9}$	$9.6739 \cdot 10^{-11}$
6	$4.9280 \cdot 10^{-10}$	$4.9305 \cdot 10^{-13}$
7	$2.6768 \cdot 10^{-11}$	$5.1917 \cdot 10^{-15}$
8	$1.5416 \cdot 10^{-12}$	$3.1615 \cdot 10^{-17}$
9	$9.0982 \cdot 10^{-14}$	
10	$5.4875 \cdot 10^{-15}$	
11	$3.3363 \cdot 10^{-16}$	

We note that 3 iterations were saved by using algorithm (**MBS**). So in this example algorithm (**MBS**) was substantially faster than algorithm (**BS**).

Example 18. Let A be of order $n = 400$, whose upper (lower) diagonal entries are uniformly 0.001 (0.002). The diagonal entries were set $d_1 = d_2 = d_3 = 3$ and $d_i = 4$, $4 \le i \le n$. l was chosen to be 3. Both algorithms were used to locate the invariant subspace spanned by eigenvectors whose eigenvalues are close to 3. The convergence condition (28) was satisfied. In fact, we have

$$\delta - \tilde{\epsilon} = 0.3692, \quad 2\sqrt{\eta\gamma} = 0.0971$$

$$\frac{2\gamma}{\delta - \tilde{\epsilon}} = 0.3739, \quad \tilde{\rho} = 0.6566$$

It was observed that $||P_{58} - P_{57}|| \le eps$, while $||\overline{P}_{15} - \overline{P}_{14}|| \le eps$. Hence, 43 iterations were saved by using algorithm (**MBS**). Accordingly the execution time of algorithm (**BS**) was 33.86 seconds, while the execution time of algorithm (**MBS**) was 9.24 seconds. This is an extreme example. However, this example demonstrates the point that to compute $\Psi(P)$ and $\Phi(P)$ takes essentially the same time and also the point that in some cases, algorithm (**MBS**) is much faster than algorithm (**BS**).

Remark 19. In the above three examples, step (3) of algorithm (**MBS**) was never used.

References

1. M. M. Blevins and G. W. Stewart, "Calculating the Eigenvectors of Diagonally Dominant Matrices," *Journal of the Association for Computing Machinery,* 21 (1974), 261–271.
2. W. W. Hager, *Applied Numerical Linear Algebra,* Prentice Hall, Englewood Cliffs, N.J., 1988.
3. J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables,* Academic Press, New York, 1970.
4. N. H. Rhee, "Calculating an Invariant Subspace of Diagonally Dominant Matrices – Part I," *Missouri Journal of Mathematical Sciences,* 5 (1993), 71–82.
5. G. W. Stewart, "Error Bounds for Invariant Subspaces of Closed Operators," *SIAM J. Numer. Anal.,* 8 (1971), 196–208.
6. J. H. Wilkinson, *The Algebraic Eigenvalue Problem,* Oxford Univ. Press, London/New York, 1965.