American Journal of Computational Mathematics, 2024, 14, 248-256
https://www.scirp.org/journal/ajcm
ISSN Online: 2161-1211
ISSN Print: 2161-1203

Another SSOR Iteration Method

Thomas Smotzer, John Buoni

Department of Mathematics and Statistics, Youngstown State University, Youngstown, Ohio, USA
Email: tsmotzer@ysu.edu

How to cite this paper: Smotzer, T. and Buoni, J. (2024) Another SSOR Iteration Method. *American Journal of Computational Mathematics*, **14**, 248-256.
https://doi.org/10.4236/ajcm.2024.142009

Received: April 30, 2024
Accepted: June 15, 2024
Published: June 18, 2024

Abstract

Kellogg gave a version of the Peaceman-Radford method. In this paper, we introduce a SSOR iteration method which uses Kellogg's method. The new algorithm has some advantages over the traditional SSOR algorithm. A Cyclic Reduction algorithm is introduced via a decoupling in Kellogg's method.

Keywords

Matrix Splitting, SSOR Iteration, KSSOR Iteration Method, Kellogg-Type SSOR Iteration, Cyclic Reduction

1. Introduction

Throughout this paper, one studies a solution method for the system of linear equations which occurs throughout the literature [1]-[11].

$$Ax = b \tag{1}$$

where $A = (a_{ij}) \in R^{n \times n}$ is non-singular matrix with unit diagonal b and $x \in R^n$ with x being an unknown vector in R^n.

Splitting the matrix A into

$$A = I - L - U \tag{2}$$

where I is the identity matrix, L and U are the strictly lower and strictly upper triangular parts of the matrix A, respectively; see [8] and [9], the successive overrelaxation iteration formula for the SOR iteration method for an optimal relaxation factor ω is as follows:

$$x^{k+1} = (I - \omega L)^{-1}((1-\omega)I + \omega U)x^k + \omega b. \tag{3}$$

While the following 2-step scheme:

$$\begin{aligned} (I - \omega L)y^k &= ((1-\omega)I + \omega U)x^k + \omega b, \\ (I - \omega U)x^{k+1} &= ((I-\omega)I + \omega L)y^k + \omega b \end{aligned} \tag{4}$$

for any initial guess x^0 defines the iteration formula for the SSOR iteration method with relaxation factor $\omega > 0$, [12] and [10]. The study of allowable ω for convergence of (4) appears throughout the literature [2] and [5].

Define the iteration matrix for the SOR iteration method as

$$\Delta(\omega) := (I - \omega L)^{-1}((1-\omega)I + \omega U), \tag{5}$$

and the iteration matrix for the SSOR iteration method as

$$\Gamma(\omega) := (I - \omega U)^{-1}((1-\omega)I + \omega L)(I - \omega L)^{-1}((1-\omega)I + \omega U). \tag{6}$$

The required two step iteration is a drawback of the SSOR iteration method, when compared to SOR iteration method. However, for a symmetric matrix *A*, the iteration matrix has real spectrum, and the spectral radius of $\Gamma(\omega)$ *i.e.* $\rho(\Gamma(\omega))$, is equal to the square of a norm of $\Delta(\omega)$ ([9]: pg. 237). This paper introduces a variation of the SSOR iteration method based on the work of Kellogg [13] and is similar to work that appears in [7] for the HSS algorithm. This new algorithm requires approximately the amount of work per iteration as the SOR iteration method, and uses an iteration matrix which is similar to the SSOR iteration method matrix with equal spectral radius. This iteration scheme will be entitled the Kellogg-type SSOR iteration method, and its iteration matrix will be denoted by the KSSOR iteration matrix.

2. Different Iterations

Letting the terms in $\Gamma(\omega)$ be somewhat permuted, one defines

$$\begin{aligned} \Psi(\omega) &:= (I - \omega L)^{-1}((1-\omega)I + \omega L)(I - \omega U)^{-1}((1-\omega)I + \omega U), \\ \Theta(\omega) &:= (I - \omega U)^{-1}((1-\omega)I + \omega U)(I - \omega L)^{-1}((1-\omega)I + \omega L). \end{aligned} \tag{7}$$

If *A* is a symmetric matrix then $\Psi(\omega)$ and $\Theta(\omega)$ are symmetric matrices and in this case, the SSOR iteration matrix has real spectrum. A relationship between the spectral radii of the three matrices $\Gamma(\omega)$, $\Psi(\omega)$ and $\Theta(\omega)$ follows.

Lemma 1. $\rho(\Gamma(\omega)) = \rho(\Psi(\omega)) = \rho(\Theta(\omega))$, for all $\omega > 0$.

Proof. Since

$(I - \omega U)\Gamma(\omega)(I - \omega U)^{-1} = ((1-\omega)I + \omega L)(I - \omega L)^{-1}((1-\omega)I + \omega U)(I - \omega U)^{-1}$ $= \Psi(\omega)$, then $\Gamma(\omega)$ is similar to $\Psi(\omega)$. Now $\Theta(\omega)$ and $\Psi(\omega)$ have the same spectral radius from the well known product theorem [14], *i.e.* $\rho(AB) = \rho(BA)$ for any *A* and *B*. Hence, $\rho(\Gamma(\omega))$, $\rho(\Psi(\omega))$ and $\rho(\Theta(\omega))$ are equal.

Lemma 1 suggests using a different ordering for the SSOR iteration scheme.

Assuming that the SSOR iterative method converges for $\omega > 0$, *i.e.* $\rho(\Gamma(\omega)) < 1$ with ω a positive acceleration factor, then the Kellogg-type SSOR iteration method is written as:

$$\begin{aligned} (I - \omega L) y^k &= ((1-\omega)I + \omega L) x^k + \omega b_1, \\ (I - \omega U) x^{k+1} &= ((1-\omega)I + \omega U) y^k + \omega b_2 \end{aligned} \tag{8}$$

where $b = b_1 + b_2$. Now for $z = y^k + x^{k+1}$ in our trial solution, one tests $\|Az - b\|$ where $\|\cdot\|$ is the 2-norm. If this quantity fails to satisfy our convergence criteria, one assigns x^{k+1} to x^k, and reiterates.

Upon eliminating y^k and the x^k, from (8) one has

$$x^{k+1} = \Theta(\omega) x^k + (I - \omega L)^{-1}\left((1-\omega) I + \omega L\right)(I - \omega U)^{-1}\omega b_1 + (I - \omega U)^{-1}\omega b_2$$

and (9)

$$y^{k+1} = \Psi(\omega) y^k + (I - \omega U)^{-1}\left((1-\omega) I + \omega U\right)(I - \omega L)^{-1}\omega b_2 + (I - \omega L)^{-1}\omega b_1.$$

This decouples the original system (4).

The following lemma will be used to show the convergence of the Kellogg-type SSOR iteration method to a unique solution to the system of equations $Ax = b$.

Lemma 2. For any c_1, c_2 then there exists unique y, x satisfying

$$\begin{aligned} x &= (I - \omega L)^{-1}\left((1-\omega) I + \omega L\right) y + c_1, \\ y &= (I - \omega U)^{-1}\left((1-\omega) I + \omega U\right) x + c_2. \end{aligned} \tag{10}$$

Furthemore,

$$\begin{aligned} y &= \Psi(\omega) y + (I - \omega L)^{-1}\left((I-\omega) I + \omega L\right) c_2 + c_1, \\ x &= \Theta(\omega) x + (I - \omega U)^{-1}\left((I-\omega) I + \omega U\right) c_1 + c_2. \end{aligned} \tag{11}$$

Proof. The system of equations (10) may be represented by

$$\tilde{A}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix},$$

where

$$\begin{aligned} \tilde{A} &= \begin{bmatrix} I & -(I - \omega L)^{-1}\left((I-\omega) I + \omega L\right) \\ -(I - \omega U)^{-1}\left((I-\omega) I + \omega U\right) & I \end{bmatrix} \\ &= \begin{bmatrix} I & 0 \\ -(I - \omega U)^{-1}\left((I-\omega) I + \omega U\right) & I \end{bmatrix}\begin{bmatrix} I & 0 \\ 0 & I - \Theta(\omega) \end{bmatrix} \\ &\quad * \begin{bmatrix} I & -(I - \omega L)^{-1}\left((I-\omega) I + \omega L\right) \\ 0 & I \end{bmatrix}. \end{aligned}$$

The det of $\tilde{A}$ is non-zero, completing the proof of the first part. The second part follows immediately.

The following theorem proves that the sequences $\{x^k\}$ and $\{y^k\}$ converge to x and y, respectively, and $x + y$ is the solution to $Ax = b$.

Theorem 1. Let x^k and y^k be defined by (9) Then (i) $\lim_{k\to\infty} x^k = x$, $\lim_{k\to\infty} y^k = y$, and (ii) $A(x + y) = b$.

Proof. Setting $e^n = x - x^n$, $c_1 = (I - \omega L)^{-1}\omega b_1$ and $c_2 = (I - \omega U)^{-1}\omega b_2$. Then

$$\begin{aligned} y - y^n &= (I - \omega L)^{-1}\left((I-\omega) I + \omega L\right) x + c_1 - (I - \omega L)^{-1}\left((I-\omega) I + \omega L\right) x^n - c_1 \\ &= (I - \omega L)^{-1}\left((I-\omega) I + \omega L\right) x - (I - \omega L)^{-1}\left((I-\omega) I + \omega L\right) x^n \end{aligned}.$$

Since $x=(I-\omega U)^{-1}((I-\omega)I+\omega U)y+c_2$ and
$x^n=(I-\omega U)^{-1}((I-\omega)I+\omega U)y^{n-1}+c_2$, then
$(I-\omega L)^{-1}((I-\omega)I+\omega L)x-(I-\omega L)^{-1}((I-\omega)I+\omega L)x^n$ becomes
$\Theta(\omega)y+(I-\omega L)^{-1}((I-\omega)I+\omega L)c_2-\Theta(\omega)x^n-(I-\omega L)^{-1}((I-\omega)I+\omega L)c_2$

In which case, $e^{n+1}=\Theta(\omega)e^n$ and since $\rho(\Theta(\omega))<1$, then $\lim_{k\to\infty}x^k=x$. In a similar fashion, $\lim_{k\to\infty}y^k=y$.

To show that $A(x+y)=b$, set $c_1=(I-\omega L)^{-1}\omega b_1$ and $c_2=(I-\omega U)^{-1}\omega b_2$ by (i) and Lemma 2, the vector $\begin{bmatrix}x\\y\end{bmatrix}$ satisfies

$$\begin{aligned}y&=(I-\omega L)^{-1}((1-\omega)I+\omega L)x+c_1\\x&=(I-\omega U)^{-1}((1-\omega)I+\omega U)y+c_2.\end{aligned}\tag{12}$$

Rewriting this as

$$\begin{aligned}(I-\omega L)y&=((1-\omega)I+\omega L)x+\omega b_1\\(I-\omega U)x&=((1-\omega)I+\omega U)y+\omega b_2,\end{aligned}\tag{13}$$

and then adding them together, canceling, and dividing by ω. This simplifies to $(x+y)-L(x+y)-U(x+y)=(I-L-U)(x+y)=b_1+b_2$ as promised.

A flowchart for the Kellog-type SSOR iteration method follows, see **Figure 1**.

3. A Closer Look

For strictly lower and strictly upper triangular parts of an $n\times n$ matrix, the following holds.

Lemma 3. For L and U the strictly lower and strictly upper triangular parts of the $n\times n$ matrix A then

$$(I-\omega L)^{-1}((1-\omega)I+\omega L)\text{ and }(I-\omega U)^{-1}((1-\omega)I+\omega U)\tag{14}$$

are lower triangular and upper triangular matrices respectively.

Proof. Since L^n is equal to the zero matrix and $\rho(\omega L)<1$ then $(I-\omega L)^{-1}$ exists and is given by $\sum_{i=0}^{n-1}(\omega L)^i$.

Upon multiplying by $((1-\omega)I+\omega L)$ gives the result that $(I-\omega L)^{-1}((1-\omega)I+\omega L)$ is a lower triangular matrix.

Similarly, $(I-\omega U)^{-1}((1-\omega)I+\omega U)$ is upper triangular.

One of the main drawbacks of the SSOR iteration method when compared to SOR iteration method is its two step iteration. Some benefits of SSOR iteration method for a symmetric matrix A, is $\Gamma(\omega)$ will have positive real spectrum and that $\rho(\Gamma(\omega))$ is equal to the square of a norm of $\Delta(\omega)$. See [9] pg. 237. In the KSSOR iteration method, if one takes $b_1=b$ and $b_2=0$, then each step is a triangular matrix (one lower, in the first step of the iteration, while upper in the second step of the iteration) times a vector operation. Since the amount of

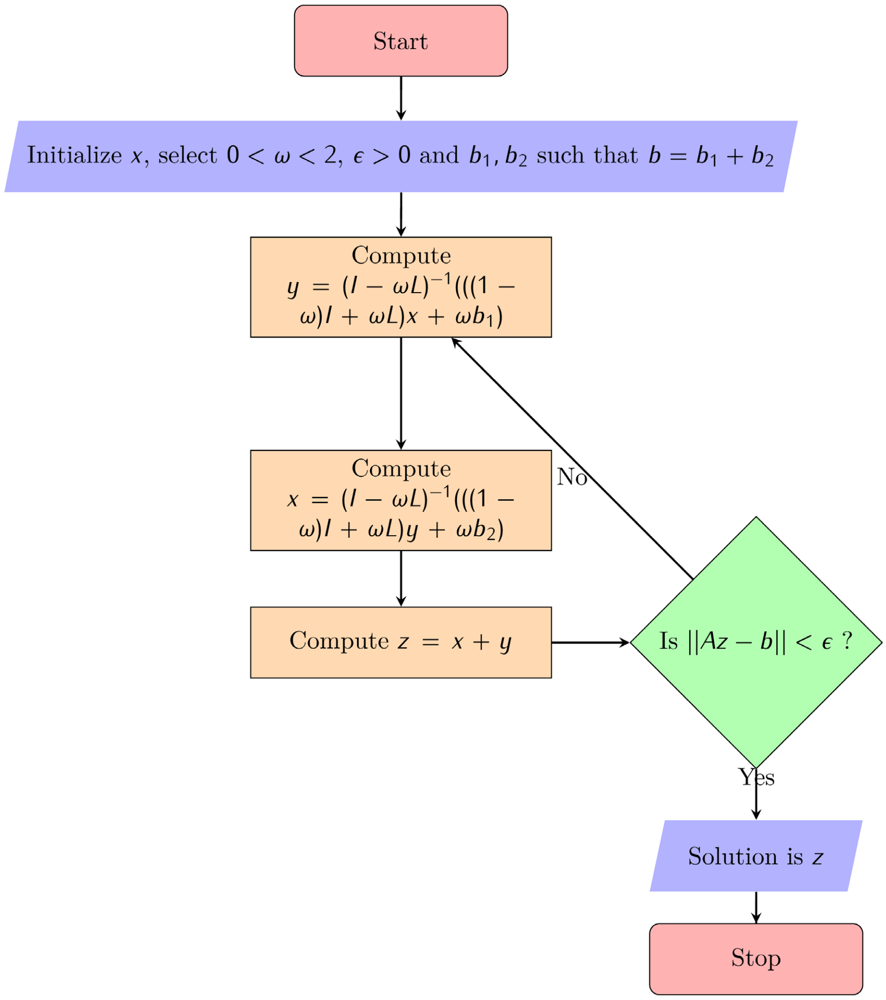

Figure 1. Kellog-type SSOR iteration method flowchart.

work done by KSSOR iteration may be considered less than the SSOR iteration method. Furthermore, since the spectral radius of the SSOR iteration matrix and the KSSOR iteration matrix are equal, their iteration counts should be fairly equal. However, in most instances the inverse would not be taken.

4. Cyclic Reduction

A Cyclic Reduction Algorithm, similar to [3], and [8] pg. 170, is as follows.

Set $\tilde{A} = \begin{bmatrix} I & -(I-\omega L)^{-1}((I-\omega)I+\omega L) \\ -(I-\omega U)^{-1}((I-\omega)I+\omega U) & I \end{bmatrix}$ one sets $K = I - \tilde{A} = \begin{bmatrix} 0 & (I-\omega L)^{-1}((I-\omega)I+\omega L) \\ (I-\omega U)^{-1}((I-\omega)I+\omega U) & 0 \end{bmatrix}$ and then $K^2 = \begin{bmatrix} \Theta(\omega) & 0 \\ 0 & \Psi(\omega) \end{bmatrix}$.

Lemma 4. If $\rho(\Gamma(\omega)) < 1$ then $\rho(K^2) < 1$ and $\rho(K) < 1$.

Proof. Since $\sigma\left(K^2\right)=\sigma\left(\Theta(\omega)\right)\cup\sigma\left(\Psi(\omega)\right)=\sigma\left(\Theta(\omega)\right)$, where the last equality follows from the similarity of $\Theta(\omega)$ and $\Psi(\omega)$ then $\rho\left(K^2\right)<1$. By the spectral mapping theorem $\pm\sqrt{\lambda}\in\sigma(K)$ if and only if $\lambda\in\sigma\left(K^2\right)$; *i.e.* $\text{determinant}\left(A^2-\lambda I\right)=\text{determinant}\left(A+\sqrt{\lambda}I\right)\text{determinant}\left(A-\sqrt{\lambda}I\right)$. hence, $\rho(K)<1$, see [14]. This completes the proof of the lemma.

K^2 defines the following uncoupled iteration scheme.

$$\begin{aligned} x^{m+2} &= \Theta(\omega)x^m + k_1 \\ y^{m+2} &= \Psi(\omega)y^m + k_2 \end{aligned} \tag{15}$$

where $k_1=(I-\omega L)^{-1}\left(\left((I-\omega)I+\omega L\right)(I-\omega U)^{-1}b_2+(I-\omega L)^{-1}b_1\right.$ and $k_2=(I-\omega U)^{-1}\left(\left((I-\omega)I+\omega U\right)(I-\omega L)^{-1}b_1+(I-\omega U)^{-1}b_2\right.$.

We now focus our attention on the solution of this reduced equation $x^{m+2}=\Theta(\omega)x^m+k_1$.

Since $\rho\left(\Theta(\omega)\right)<1$, then the Cyclic Reduction Scheme x^m converges to x and having found the vector x in (10), form the vector $y=(I-\omega U)^{-1}\left(\left((I-\omega)I+\omega U\right)x+(I-\omega U)^{-1}b_2\right.$ and from which $x+y$ is a solution.

A flowchart for the Cyclic Reduction iteration method follows, see **Figure 2**.

5. Examples

In the following examples, a solution vector x is constructed as $x(i)=\frac{i}{N}\sin\left(\frac{i*\pi}{6}\right)$ for $i=1,\cdots,N$ and an initial guess is $x_0=[1,1,\cdots,1]^{\mathrm{T}}$. The error in each of our schemes will be measured by the 2-norm $\|Ay-b\|$. where y is the resulting approximate answer and our desired error tolerance will be 10^{-6}. Our table includes the $\|x-y\|$ *i.e.* the actual 2-norm difference between the true solution, x, and our approximate solution y, the iteration count, and the elapsed time in seconds for the work of the loop doing the iteration. ω is found through an iterative search, all decimals are rounded to two places and in the KSSOR iteration method and the Cyclic Reduction Method, $b_1=b$ and $b_2=0$.

Example 1. Consider the system of linear equations with the $n\times n$ symmetric matrix which appears in [4] Set P to be the $m\times m$ matrix

$$P=\begin{bmatrix} 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & \dots & 0 \\ . & . & . & . & . & . & & \\ 0 & 0 & 0 & .. & & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & .. & 0 & -1 & 2 \end{bmatrix}$$

and $T=(I\otimes P+P\otimes I)-(P\otimes P)/6$, and $A=D^{-1}*T$ where D is the diagonal of T. The resulting matrix A, is then $n\times n$ where $n=m^2$. For $m=32$, the $n\times n$ matrix is 1024×1024 and **Table 1** summarizes our results.

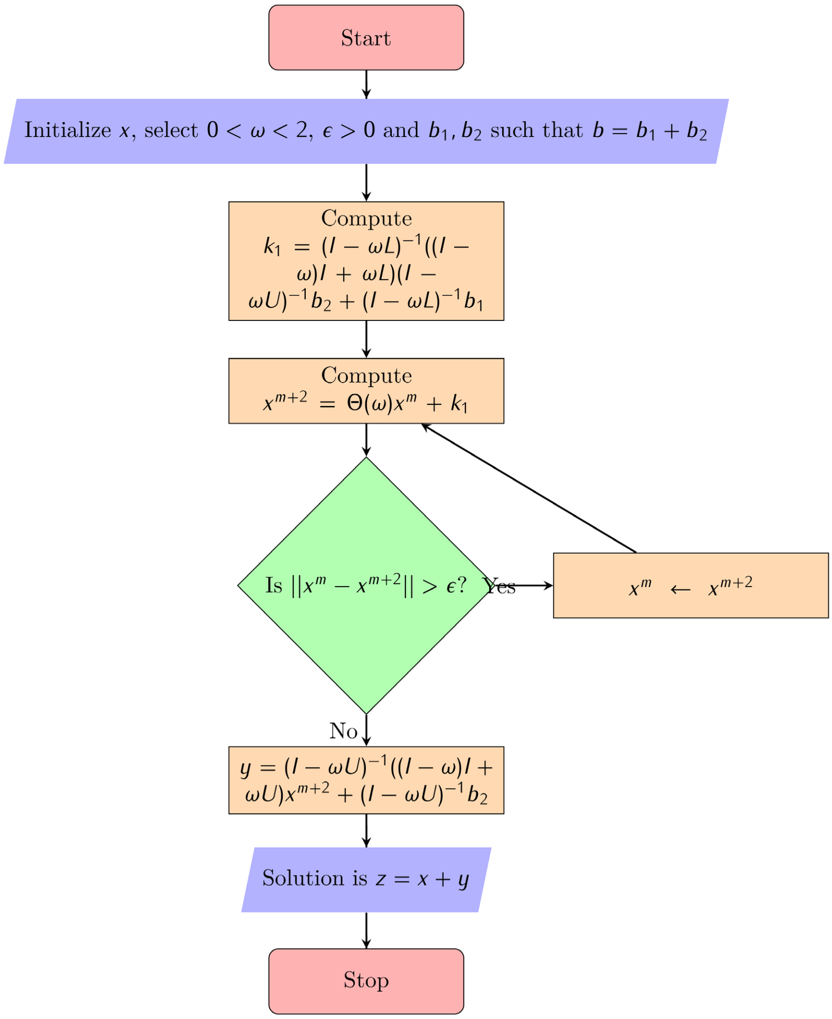

Figure 2. Cyclic Reduction iteration method flowchart.

Table 1. Example 1.

Method	ω	Iterations	$\|Ay-b\|$	$\|x-y\|$	Time
SOR	1.81	96	9.25×10^{-7}	6.24×10^{-5}	0.78 secs
SSOR	1.85	81	9.14×10^{-7}	1.06×10^{-7}	1.12 secs
KSSOR	1.85	83	8.98×10^{-7}	1.04×10^{-4}	0.80 secs
CyclicRed.	1.85	81	1.137×10^{-6}	1.30×10^{-4}	0.53 secs

Example 2. Consider the system of linear equations with 1024×1024 **coefficient matrix which appears in [11].**

$$T = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 3 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 5 & 3 & 0 & 0 & \cdots & 0 \\ . & . & . & . & . & . & & \\ 0 & 0 & 0 & .. & & -1 & 2*N-3 & N-1 \\ 0 & 0 & 0 & 0 & .. & 0 & -1 & 2*N-1 \end{bmatrix}$$

Table 2. Example 2.

Method	ω	Iterations	$\|Ay-b\|$	$\|x-y\|$	Time
SOR	0.88	18	9.70×10^{-7}	1.63×10^{-6}	0.21 secs
SSOR	1.2	8	6.27×10^{-7}	4.45×10^{-7}	0.14 secs
KSSOR	1.2	9	3.19×10^{-7}	3.23×10^{-7}	0.13 secs
CyclicRed.	1.1	9	1.00×10^{-7}	1.17×10^{-74}	0.44 secs

Table 3. Example 3.

Method	ω	Iterations	$\|Ay-b\|$	$\|x-y\|$	Time
SOR	1.51	175	9.9×10^{-7}	7.6×10^{-5}	1.78 secs
SSOR	1.6	82	9.19×10^{-7}	7.64×10^{-5}	1.48 secs
KSSOR	1.6	86	9.54×10^{-7}	7.94×10^{-5}	1.05 secs
CyclicRed.	1.6	82	1.63×10^{-6}	1.36×10^{-4}	0.83 secs

Then A becomes $D^{-1} * T$ where D is the diagonal of T. Table 2 summarizes our results.

Example 3 Consider the system of linear equations with $n \times n$ coefficient matrix

$$T = \begin{bmatrix} B & E \\ -E' & 0.5 * I \end{bmatrix}$$

where $B = \begin{bmatrix} I \otimes C + C \otimes I & 0 \\ 0 & I \otimes C + C \otimes I \end{bmatrix}$, $E = \begin{bmatrix} I \otimes F \\ F \otimes I \end{bmatrix}$.

Where $C = \text{tridiag}(-1, 2, -1)$, $F = h * \text{tridiag}(-1, 1, 0)$ are $m \times m$ matrices with $h = \frac{1}{m+1}$ resulting in $n = 3m^2$. Then A becomes $D^{-1} * T$ where D is the diagonal of T and for $m = 20$ is a 1200×1200 matrix. Table 3 summarizes our results.

Since the spectral radius of all iteration matrices is the same in all the iteration schemes, the convergence to a solution of the system of linear equations has about the same number of iterations for all the iteration convergence schemes considered in this paper.

6. Conclusion

The KSSOR algorithm requires an upper and lower triangular solution to a linear system which may be more accurate than the SSOR algorithm which on the surface requires full matrix solves at each step.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Aljuhani, A., Maturi, D. and Alshehri, H. (2022) Variational Iteration Method for Solving Boussinesq Equations Using Maple. *Applied Mathematics*, **13**, 960-967. https://doi.org/10.4236/am.2022.1312060

[2] Hadjidimos, A. and Neumann, M. (1989) Convergence Domains of SSOR Method Associated with p-Cyclic Matrices. *BIT Numerical Mathematics*, **29**, 311-320. https://doi.org/10.1007/BF01952685

[3] Hageman, L. and Varga, R. (1964) Block Iterative Methods for Cyclically Reduced Matrix Equations. *Numerische Mathematik*, **6**, 106-119. https://doi.org/10.1007/BF01386061

[4] Lynch, R.E., Rice, J.R. and Thomas, D.R. (1965) Tensor Product Analysus of Alternating Direction Implicit Methods*. Journal of the Society for Industrial and Applied Mathematics*, **13**, 995-1006. https://doi.org/10.1137/0113067

[5] Neumaier, G. and Varga, R.S. (1984) Exact Convergence and Divergence Domains for the Symmetric Successive Overrelaxation Iterative (SSOR) Method Applied to H-Matrices. *Linear Algebra and Its Applications*, **58**, 261-274. https://doi.org/10.1016/0024-3795(84)90216-7

[6] Ojarikre, H. and Mamadu, E. (2022) Time Discretized Variational Iteration Method for the Stochastic Volatility Process with Jumps. *Advances in Pure Mathematics*, **12**, 693-700. https://doi.org/10.4236/apm.2022.1211052

[7] Smotzer, T. and Buoni, J. (2023) Another HSS Iteration Method. *Numerical Algorithms*. https://doi.org/10.1007/s11075-023-01728-1

[8] Varga, R.S. (2000) Matrix Iterative Analysis. 2nd Edition, Springer. https://doi.org/10.1007/978-3-642-05156-2

[9] Young, D. (1971) Iterative Solutions of Large Linear Systems. Academic Press.

[10] Young, D. (1971) On the Accelerated SSOR Method for Solving Large Linear Systems. *Advances in Mathematics*, **23**, 215-271. https://doi.org/10.1016/S0001-8708(77)80029-7

[11] Zhang, C.-Y., Xue, Z. and Luo, S. (2016) A Convergence Analysis of SOR Iterative Methods for Linear Systems with Weak H-Matrices. *Open Mathematics*, **14**, 747-760. https://doi.org/10.1515/math-2016-0065

[12] Sheldon, J. (1955) On the Numerical Solution of Elliptic Difference Equations. *Mathematical Tables and Other Aids to Computation*, **9**, 101-112. https://doi.org/10.2307/2002066

[13] Kellogg, R.B. (1963) Another Alternating-Direction Method. *Journal of the Society for Industrial and Applied Mathematics*, **11**, 976-981. https://doi.org/10.1137/0111071

[14] Halmos, P.R. (1974) Finite-Dimensional Vector Spaces. Springer-Verlag. https://doi.org/10.1007/978-1-4612-6387-6