Serdica J. Computing **7** (2013), No 3, 245–256

Serdica
Journal of Computing

Bulgarian Academy of Sciences
Institute of Mathematics and Informatics

A REFINEMENT OF SOME OVERRELAXATION ALGORITHMS FOR SOLVING A SYSTEM OF LINEAR EQUATIONS*

Nikolay Kyurkchiev, Anton Iliev

Abstract. In this paper we propose a refinement of some successive overrelaxation methods based on the reverse Gauss–Seidel method for solving a system of linear equations $Ax = b$ by the decomposition $A = T_m - E_m - F_m$, where T_m is a banded matrix of bandwidth $2m + 1$.

We study the convergence of the methods and give software implementation of algorithms in Mathematica package with numerical examples.

ACM Computing Classification System (1998): G.1.3.

Key words: reverse Gauss–Seidel method, or Nekrassov–Mehmke 2 method – (NM2), Successive Overrelaxation method with 1 parameter, based on (NM2) – (SOR1NM2), Successive Overrelaxation method with 2 parameters, based on (NM2) – (SOR2NM2), Refinement of (SOR1NM2) method – (RSOR1NM2), Refinement of (SOR2NM2) method – (RSOR2NM2).

*This paper is partly supported by project NI13 FMI–002 of Department for Scientific Research, Paisii Hilendarski University of Plovdiv.

1. Introduction. Let us consider the linear system:

$$Ax - b = 0. \tag{1}$$

Let $A = (a_{ij})$ be an $n \times n$ matrix and $T_m = (t_{ij})$ be a banded matrix of bandwidth $2m+1$ defined as:

$$t_{ij} = \begin{cases} a_{ij}, & |i-j| \le m, \\ 0 & \text{otherwise.} \end{cases}$$

Let

$$T_m = \begin{pmatrix} a_{11} & \cdots & a_{1,m+1} & & \\ \vdots & \ddots & & \ddots & \\ a_{m+1,1} & & \ddots & & a_{n-m,n} \\ & \ddots & & \ddots & \vdots \\ & & a_{n,n-m} & \cdots & a_{n,n} \end{pmatrix},$$

$$E_m = \begin{pmatrix} & & & \\ & & & \\ -a_{m+2,1} & & & \\ \vdots & \ddots & & \\ -a_{n,1} & \cdots & -a_{n,n-m-1} & \end{pmatrix}$$

and

$$F_m = \begin{pmatrix} & -a_{1,m+2} & \cdots & -a_{1,n} \\ & & \ddots & \vdots \\ & & & -a_{n-m-1,n} \\ & & & \\ & & & \end{pmatrix}.$$

In [15] Salkuyeh considers the following overrelaxation method, based on Gauss–Seidel (forward algorithm) [10]–[12]:

$$x^{k+1} = (T_m - \omega E_m)^{-1}[\omega F_m + (1-\omega)T_m]x^k + (T_m - \omega E_m)^{-1}\omega b,\ k = 0, 1, 2, \ldots, \tag{2}$$

where $A = T_m - E_m - F_m$.

In [22] the following iteration scheme, based on the reverse Gauss–Seidel method [1] is proposed:

$$\begin{aligned} x^{k+1} &= (T_m - \omega F_m)^{-1}[\omega E_m + (1-\omega)T_m]x^k + (T_m - \omega F_m)^{-1}\omega b \\ &= B^m_{SOR1NM2}x^k + cb, \quad k = 0, 1, 2, \ldots. \end{aligned} \tag{3}$$

Henceforth, we shall call the above scheme the *Successive Overrelaxation method with 1 parameter, based on (NM2) – (SOR1NM2).*

In [1] D. Faddeev and V. Faddeeva pointed out that such iteration processes in which cycles studied in Gauss–Seidel (forward and reverse) algorithms alternate.

The following theorem holds true:

Theorem A [22]**.** *Let A and T_m be a strictly diagonally dominant (SDD) matrix. Then for every $0 < \omega < 2$ the (SOR1NM2) method is convergent for any initial guess x^0.*

Salkuyeh in [17] proposed the following overrelaxation method, based on Gauss–Seidel (forward algorithm):

$$x^{k+1} = (T_m - \gamma E_m)^{-1}[(1-\omega)T_m + (\omega-\gamma)E_m + \omega F_m]x^k + (T_m - \gamma E_m)^{-1}\omega b, \quad k = 0, 1, 2, \dots . \tag{4}$$

In [22] Zaharieva and Malinova published the following iteration scheme, based on the reverse Gauss–Seidel method:

$$\begin{aligned} x^{k+1} &= (T_m - \gamma F_m)^{-1}[(1-\omega)T_m + (\omega-\gamma)F_m + \omega E_m]x^k + (T_m - \gamma F_m)^{-1}\omega b, \\ &= B^m_{SOR2NM2}x^k + c_1 b, \quad k = 0, 1, 2, \dots . \end{aligned} \tag{5}$$

We shall call the above scheme the *Successive Overrelaxation method with 2 parameters, based on (NM2) – (SOR2NM2).*

Definition. *A is an M- matrix if $a_{ij} \le 0$ for $i \ne j$, A is non-singular and $A^{-1} \ge 0$.*

The following theorem holds true:

Theorem B [22]**.** *If A is an M-matrix and $0 \le \gamma < \omega \le 1$ with $\omega \ne 0$, then the (SOR2NM2) method is convergent, i.e.:*

$$\rho\,(B^m_{SOR2NM2}) < 1.$$

For other results, see [3]–[6], [8], [9], [21], and [23].

2. Main results. In this paper, following the ideas given in [20] and [7], we propose a refinement of the methods (SOR1NM2) and (SOR2NM2).

I. Let x^1 be an initial approximation for the solution of system (1) and $b_i^1 = \sum_{j=1}^{n} a_{ij} x_j^1$, $i = 1, 2, \ldots, n$.

After k^{th} step we have: $b_i^{k+1} = \sum_{j=1}^{n} a_{ij} x_j^{k+1}$, $i = 1, 2, \ldots, n$.

Now we refine this obtained solution as $b_i^{k+1} \to b_i$.

Assume that $\tilde{x}^{k+1} = \left(\tilde{x}_1^{k+1}, \ldots, \tilde{x}_n^{k+1}\right)$ is good approximation for the solution of system (1), i.e., $\tilde{x}^{k+1} \to x$, where x is the exact solution of system (1) and $b_i = \sum_{j=1}^{n} a_{ij} \tilde{x}_j^{k+1}$, $i = 1, 2, \ldots, n$.

Since all $\tilde{x}_t^{k+1}$ are unknown, we define them as follows, $\tilde{x}^{k+1} = x^{k+1} + b^{k+1} - b$.

By the decomposition

$$\omega A = (T_m - \omega F_m) - [(1 - \omega) T_m + \omega E_m]$$

we have

$$\begin{aligned}
&[(T_m - \omega F_m) - [(1 - \omega) T_m + \omega E_m]]x = \omega b \\
&(T_m - \omega F_m)x = [\omega E_m + (1 - \omega) T_m]x + \omega b \\
&(T_m - \omega F_m)x = [T_m - \omega F_m - \omega A]x + \omega b \qquad (6)\\
&(T_m - \omega F_m)x = (T_m - \omega F_m)x + \omega(b - Ax) \\
&x = x + \omega (T_m - \omega F_m)^{-1}(b - Ax)
\end{aligned}$$

i.e.

$$\tilde{x}^{k+1} = x^{k+1} + \omega (T_m - \omega F_m)^{-1}(b - Ax^{k+1}).$$

For the method (3) we have

$$
\begin{aligned}
x^{k+1} &= (T_m - \omega F_m)^{-1}[\omega E_m + (1-\omega)T_m]x^k + (T_m - \omega F_m)^{-1}\omega b + \\
&+ (T_m - \omega F_m)^{-1}\left[\omega b - \omega A\left[(T_m - \omega F_m)^{-1}[\omega E_m + (1-\omega)T_m]x^k + \right.\right. \\
&+ \left.\left. (T_m - \omega F_m)^{-1}\omega b\right]\right] \\
&= \left[(T_m - \omega F_m)^{-1}[\omega E_m + (1-\omega)T_m]\right]^2 x^k + \\
&+ \left[I + (T_m - \omega F_m)^{-1}[\omega E_m + (1-\omega)T_m]\right](T_m - \omega F_m)^{-1}\omega b \\
&= B^m_{RSOR1NM2}x^k + c_2 b, \quad k = 0, 1, 2, \dots,
\end{aligned}
\tag{7}
$$

We shall call the above scheme the *Refinement of (SOR1NM2) method – (RSOR1NM2)*.

The following theorem holds true:

Theorem 1. *Let A be a strictly diagonally dominant (SDD) matrix.*

Then for any natural number $m < n$ the (RSOR1NM2) method is convergent for any initial guess x^0.

Proof. Assuming x is the real solution of (1), as A is a SDD matrix by Theorem A, a (SOR1NM2) method is convergent.

Let $x^{k+1} \to x$. Then

$$\|\tilde{x}^{k+1} - x\|_\infty \leq \|x^{k+1} - x\|_\infty + \omega\|(T_m - \omega F_m)^{-1}\|_\infty\|(b - Ax^{k+1})\|_\infty.$$

From the fact $\|x^{k+1} - x\|_\infty \to 0$, we have $\|(b - Ax^{k+1})\|_\infty \to 0$.

Therefore, $\|\tilde{x}^{k+1} - x\|_\infty \to 0$ and a (RSOR1NM2) method is convergent. □

II. By the decomposition

$$\omega A = (T_m - \gamma F_m) - [(1-\omega)T_m + (\omega - \gamma)F_m + \omega E_m]$$

we have

$$
\begin{aligned}
&[(T_m-\gamma F_m)-[(1-\omega)T_m+(\omega-\gamma)F_m+\omega E_m]]x=\omega b\\
&(T_m-\gamma F_m)x=[(1-\omega)T_m+(\omega-\gamma)F_m+\omega E_m]x+\omega b\\
&(T_m-\gamma F_m)x=[T_m-\gamma F_m-\omega A]x+\omega b \qquad (8)\\
&(T_m-\gamma F_m)x=(T_m-\gamma F_m)x+\omega(b-Ax)\\
&x=x+\omega(T_m-\gamma F_m)^{-1}(b-Ax)
\end{aligned}
$$

i.e.

$$\tilde{x}^{k+1}=x^{k+1}+\omega(T_m-\gamma F_m)^{-1}(b-Ax^{k+1}).$$

For the method (5) we have

$$
\begin{aligned}
x^{k+1} &= (T_m-\gamma F_m)^{-1}[(1-\omega)T_m+(\omega-\gamma)F_m+\omega E_m]x^k+\\
&+ (T_m-\gamma F_m)^{-1}\omega b+(T_m-\gamma F_m)^{-1}\left[\omega b-\omega A\left[(T_m-\gamma F_m)^{-1}[(1-\omega)T_m+\right.\right.\\
&+ \left.\left.(\omega-\gamma)F_m+\omega E_m]x^k+(T_m-\gamma F_m)^{-1}\omega b\right]\right] \qquad (9)\\
&= \left[(T_m-\gamma F_m)^{-1}[(1-\omega)T_m+(\omega-\gamma)F_m+\omega E_m]\right]^2x^k+\\
&+ \left[I+(T_m-\gamma F_m)^{-1}[(1-\omega)T_m+(\omega-\gamma)F_m+\omega E_m]\right](T_m-\gamma F_m)^{-1}\omega b\\
&= B^m_{RSOR2NM2}x^k+c_3b,\quad k=0,1,2,\dots,
\end{aligned}
$$

We shall call the above scheme the *Refinement of (SOR2NM2) method – (RSOR2NM2)*.

The following theorem holds true:

Theorem 2. *Let A be an M-matrix. Then for any natural number $m<n$ the (RSOR2NM2) method is convergent for any initial guess x^0.*

The proof follows the ideas given in [21], and will be omitted.

Remark. If the (SOR1NM2) method is convergent, then the (RSOR2NM2) method is also convergent.

Evidently, the (RSOR2NM2) method yields considerable improvement in the rate of convergence for iterative method (SOR2NM2).

III. We define the new Refinement Symmetric Successive Overrelaxation Nekrassov–Mehmke method (RSSOR2NM2) consists the cyclic procedures

$$x^{k+1/2} = \left[(T_m - \gamma E_m)^{-1}[(1-\omega)T_m + (\omega-\gamma)E_m + \omega F_m]\right]^2 x^k + \alpha b,$$

$$x^{k+1} = \left[(T_m - \gamma F_m)^{-1}[(1-\omega)T_m + (\omega-\gamma)F_m + \omega E_m]\right]^2 x^{k+1/2} + \beta b.$$

This gives the recurrence

$$x^{k+1} = B^m_{RSSOR2NM2} x^k + \delta b,$$

where

$$\begin{aligned} B^m_{RSSOR2NM2} &= \left[(T_m - \gamma E_m)^{-1}[(1-\omega)T_m + (\omega-\gamma)E_m + \omega F_m]\right]^2 \times \\ &\times \left[(T_m - \gamma F_m)^{-1}[(1-\omega)T_m + (\omega-\gamma)F_m + \omega E_m]\right]^2 . \end{aligned}$$

3. Numerical example. Let A is an M–matrix (example by Salkuyeh [16]):

$$\begin{pmatrix} 4 & -2 & -1 & -2 \\ -1 & 5 & -5 & -1 \\ -2 & -1 & 9 & -1 \\ -1 & -1 & -1 & 5 \end{pmatrix}.$$

Let $\gamma = 0.5$, $\omega = 0.9$.

For algorithms (5) and (9) and $m = 1$ we have (see Figure 2):

$$\rho\left(B^1_{RSOR2NM2}\right) = 0.4927 < 0.7019 = \rho\left(B^1_{SOR2NM2}\right) < 1.$$

For $m = 2$ we obtain:

$$\rho\left(B^2_{RSOR2NM2}\right) = 0.245 < 0.495 = \rho\left(B^2_{SOR2NM2}\right) < 1.$$

These results show that the method (9) is more appropriate in this case.

For an implementation of algorithms (5) and (9) in the Mathematica package ([19]), see Figure 1. The results for m = 1 are shown, see Figure 2.

For other results, see [2], [13], and [14]. For other iteration schemes with increased speed of convergence, see [18].

Acknowledgements. The authors would like to thank the anonymous reviewers for their helpful and constructive comments that contributed to improving the final version of the paper.

```
A = ( 4  -2 -1 -2
     -1   5 -5 -1
     -2  -1  9 -1
     -1  -1 -1  5 );

Det[A] ≠ 0
True
Module[{g, w, m, Tm, Fm, Em, Mm, Nm, e, e1},
  (*g=Input["Give the value of the parameter γ:"];
  w=Input["Give the value of the parameter w:"];*)
  g = 0.5; w = 0.9;
  m = Input["Give the value of the parameter m:"];
  Tm = SparseArray[
    {Band[{1, 1}] → Diagonal[A], {i_, j_} /; Abs[i - j] ≤ m → Part[A, i, j]}, {4, 4}];
  Print["\nT", m, " = ", Tm // MatrixForm];
  Fm = (-1) * UpperTriangularize[A, m + 1];
  Print["\nF", m, " = ", Fm // MatrixForm];
  Em = (-1) * LowerTriangularize[A, -1 - m];
  Print["\nE", m, " = ", Em // MatrixForm];
  Mm = Tm - g Fm;
  Nm = (1 - w) Tm + (w - g) Fm + w Em;
  BSOR2NM2m = Inverse[Mm].Nm;
  Print["\nBSOR2NM2", m, " = ", BSOR2NM2m // MatrixForm];
  e = Eigenvalues[BSOR2NM2m];
  Print["\neigenvalues of BSOR2NM2", m, " = ", e // MatrixForm];
  Print["\nspectral radius of BSOR2NM2", m, " = ", Style[Max[Abs[e]], 18, Orange]];
  BRSOR2NM2m = BSOR2NM2m.BSOR2NM2m;
  Print["\nBRSOR2NM2", m, " = ", BRSOR2NM2m // MatrixForm];
  e1 = Eigenvalues[BRSOR2NM2m];
  Print["\neigenvalues of BRSOR2NM2", m, " = ", e1 // MatrixForm];
  Print["\nspectral radius of BRSOR2NM2", m, " = ", Style[Max[Abs[e1]], 18, Orange]];
 ];
```

Fig. 1

$$T1 = \begin{pmatrix} 4 & -2 & 0 & 0 \\ -1 & 5 & -5 & 0 \\ 0 & -1 & 9 & -1 \\ 0 & 0 & -1 & 5 \end{pmatrix}$$

$$F1 = \begin{pmatrix} 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$E1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

$$BSOR2NM21 = \begin{pmatrix} 0.360561 & 0.0809541 & 0.127495 & 0.314967 \\ 0.338893 & 0.162272 & 0.028842 & 0.173052 \\ 0.263511 & 0.027531 & 0.103277 & 0.019665 \\ 0.232702 & 0.185506 & 0.000655499 & 0.103933 \end{pmatrix}$$

$$\text{eigenvalues of } BSOR2NM21 = \begin{pmatrix} 0.701942 \\ 0.132076 \\ -0.0519868 + 0.0406157\, i \\ -0.0519868 - 0.0406157\, i \end{pmatrix}$$

$$\text{spectral radius of } BSOR2NM21 = 0.701942$$

$$BRSOR2NM21 = \begin{pmatrix} 0.264329 & 0.104264 & 0.0616782 & 0.162817 \\ 0.225054 & 0.0866633 & 0.0509794 & 0.153375 \\ 0.136132 & 0.0322911 & 0.0450693 & 0.0918363 \\ 0.171128 & 0.068239 & 0.0351544 & 0.116211 \end{pmatrix}$$

$$\text{eigenvalues of } BRSOR2NM21 = \begin{pmatrix} 0.492722 \\ 0.017444 \\ 0.00105299 + 0.00422296\, i \\ 0.00105299 - 0.00422296\, i \end{pmatrix}$$

$$\text{spectral radius of } BRSOR2NM21 = 0.492722$$

Fig. 2

REFERENCES

[1] Faddeev D., V. Faddeeva. Numerical Methods of Linear Algebra. 2nd ed., Fizmatgiz, M., 1963.

[2] Golev A., A. Malinova, D. Zaharieva. Software implementation of modifications of iterative algorithms for solving linear systems of equations. *Int. J. of Pure and Appl. Math.*, **76** (2012), No 4, 489–500.

[3] Iliev A., N. Kyurkchiev. Nontrivial Methods in Numerical Analysis: Selected Topics in Numerical Analysis. LAP LAMBERT Acad. Publ., Saarbrucken, 2010.

[4] Iliev A., N. Kyurkchiev, M. Petkov. On some modifications of the Nekrassov method for numerical solution of linear systems of equations. *Serdica Journal of Computing*, **3** (2009), No 4, 371–380.

[5] Jin X., Y. Wei, H. Tam. Preconditioning technique for symmetric M-matrices. *Calcolo*, **42** (2005), 105–113.

[6] Kinashi Y., H. Sawami, H. Niki. An iterative method applied to nonsymmetric linear systems. *Japan J. Indust. Appl. Math.*, **13** (1996), 235–241.

[7] Kyurkchiev N. Refinement of some iterative algorithms for solving system of linear equations. *Compt. rend. Acad. bulg. Sci.*, **65** (2012), No 12, 1657–1664.

[8] Kyurkchiev N., M. Petkov, A. Iliev. A modification of Richardson method for numerical solution of linear system of equations. *Compt. rend. Acad. bulg. Sci.*, **61** (2008), No 10, 1257–1264.

[9] Li W. A note on the preconditioned Gauss–Seidel (GS) method for linear systems. *J. Comput. Appl. Math.*, **182** (2005), 81–90.

[10] Mehmke R. On the Seidel scheme for iterative solution of linear system of equations with a very large number of unknowns by successive approximations. *Math. Sb.*, **16** (1892), No 2, 342–345 (in Russian).

[11] Mehmke R., P. Nekrassov. Solution of linear system of equations by means of successive approximations. *Math. Sb.*, **16** (1892), 437–459 (in Russian).

[12] Nekrassov P. Determination of the unknowns by the least squares when the number of unknowns is considerable. *Math. Sb.*, **12** (1885), 189–204 (in Russian).

[13] Popova E. Connectivity to Interval Libraries filib++ and C-XSC. In: Numerical Validation in Current Hardware Architectures (Eds A. Cuyt, W. Kramer, W. Luther, P. Markstein), Lecture Notes in Computer Science, Vol.**5492** , Springer Berlin/Heidelberg, 2009, 117–132.

[14] Popova E., L. Kolev, W. Kramer. A solver for complex-valuated parametric linear systems. *Serdica Journal of Computing*, **4** (2010), No 1, 123–132.

[15] Salkuyeh D. A generalization of the SOR method for solving linear system of equations. *J. Appl. Math.*, **4** (2007), No 15, 31–38.

[16] Salkuyeh D. Generalized AOR method for solving system of linear equations. *Australian J. of Basic and Appl. Sci.*, **5** (2011), No 3, 351–358.

[17] Salkuyeh D. Generalized Jacobi and Gauss–Seidel methods for solving linear system of equations. *Numer. Math. A J. of Chinese Univ. (English Ser.)*, **16** (2007), No 2, 164–170.

[18] Soleymani F. A new method for solving ill–conditioned linear systems. *Opuscula Math.*, **33** (2013), No 2, 337–344.

[19] Trott M. The MATHEMATICA GuideBook for Numerics. Springer Science+Business Media, Inc., 2006.

[20] Vatti V., G. Gonfa. Refinement of generalized Jacobi (RGJ) method for solving system of linear equations. *Int. J. Contemp. Math. Sci.*, **6** (2011), No 3, 109–116.

[21] Zaharieva D., N. Kyurkchiev, A. Iliev. Generalized Nekrassov-Mehmke procedures for solving linear system of equations. *Compt. rend. Acad. bulg. Sci.*, **64** (2011), No 4, 487–496.

[22] Zaharieva D., A. Malinova. On some (AOR) iterative algorithms for solving system of linear equations. *Plovdiv Univ. "P. Hilendarski" Sci. Works – Math.*, **38** (2011), No 3, 133–144.

[23] Zaharieva D., N. Kyurkchiev, A. Iliev. A SOR-Nekrassov-Mehmke procedure for numerical solution of linear system of equations. *Plovdiv Univ. "P. Hilendarski" Sci. Works – Math.*, **37** (2010), No 3, 121–134.

Nikolay Kyurkchiev
Faculty of Mathematics, Informatics
and Information Technology
Paisii Hilendarski University of Plovdiv
24, Tsar Assen Str.
4000 Plovdiv, Bulgaria
e-mail: nkyurk@uni-plovdiv.bg
and
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Acad. G. Bonchev Str., Bl. 8
1113 Sofia, Bulgaria
e-mail: nkyurk@math.bas.bg

Anton Iliev
Faculty of Mathematics, Informatics
and Information Technology
Paisii Hilendarski University of Plovdiv
24, Tsar Assen Str.
4000 Plovdiv, Bulgaria
e-mail: aii@uni-plovdiv.bg
and
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Acad. G. Bonchev Str., Bl. 8
1113 Sofia, Bulgaria
e-mail: anton.iliev@gmail.com

Received September 16, 2013
Final Accepted November 7, 2013