Journal of Mathematical Research with Applications
Jul., 2020, Vol. 40, No. 4, pp. 405–414
DOI:10.3770/j.issn:2095-2651.2020.04.008
Http://jmre.dlut.edu.cn

Relaxation Methods for Systems of Linear Equations and Applications

Xinzhu ZHAO, Bo DONG*, Bo YU
School of Mathematical Sciences, Dalian University of Technology, Liaoning 116024, P. R. China

Abstract The relaxation methods have served as very efficient tools for solving linear system and have many important applications in the field of science and engineering. In this paper, we study an efficient relaxation method based on the well-known Gauss-Seidel iteration method. Theoretical analysis shows our method can converge to the unique solution of the linear system. In addition, our method is applied to solve the saddle point problem and PageRank problem, and the numerical results show our method is more powerful than the existent relaxation methods.

Keywords iterative methods; relaxation methods; linear systems; saddle point problem; PageRank problem

MR(2010) Subject Classification 65F10; 65N22

1. Introduction

In many applications, one often meets linear systems

$$Ax =b, \tag{1.1}$$

where $A \in \mathbb{C}^{n\times n}$ is nonsingular and $b \in \mathbb{C}^n$. Suppose the matrix A has the following splitting

$$A = D - L - U,$$

where D is the diagonal part of the matrix A, $-L$ is the strictly lower part of the matrix A, and $-U$ is the strictly upper part of the matrix A. It is generally assumed the matrix D is nonsingular, that is, the diagonal entries of A are all nonzero.

Generally speaking, the iterative method for solving linear system is a process of going from an iterate to the next by modifying some components of an approximate solution at a time. For example, let x_k be the kth iterate. The Jacobi iteration, which is of the form

$$x_{k+1} =D^{-1}(L + U)x_k + D^{-1}b, \tag{1.2}$$

derives all components of the next iterate simultaneously, however, the Gauss-Seidel iteration

$$x_{k+1} =(D - L)^{-1}Ux_k + (D - L)^{-1}b, \tag{1.3}$$

Received September 30, 2019; Accepted March 17, 2020
Supported by the National Natural Science Foundation of China (Grant Nos. 11871136; 11801382; 11971092) and the Fundamental Research Funds for the Central Universities (Grant No. DUT19LK06).
* Corresponding author
E-mail address: mathzhaoxinzhu@mail.dlut.edu.cn (Xinzhu ZHAO); dongbo@dlut.edu.cn (Bo DONG); yubo@dlut.edu.cn (Bo YU)

is a refinement of the Jacobi iteration and updates the next iterate one by one in the order $i = 1, 2, \ldots, n$.

Relaxation is an important technique in the design of iterative methods for solving linear systems. Each iterative method can be rewritten as the following form $x_{k+1} = x_k + \Delta x_k$. Therefore, if we introduce a relaxation factor $\omega \in \mathbb{R}$, we can get the following modification of the above iteration $x_{k+1} = x_k + \omega\Delta x_k$.

Specifically, introducing a relaxation factor to the Jacobi iterate (1.2), we can get the iterate

$$x_{k+1} = x_k + \omega(D^{-1}(L+U)x_k + D^{-1}b - x_k),$$

which is equivalent to

$$Dx_{k+1} =(1-\omega)Dx_k + \omega Lx_k + \omega Ux_k + \omega b, \tag{1.4}$$

then applying the similar technique as the derivation of Gauss-Seidel iteration to the recursion (1.4), we can obtain a new recursion

$$Dx_{k+1} = (1-\omega)Dx_k + \omega Lx_{k+1} + \omega Ux_k + \omega b,$$

which is equivalent to

$$x_{k+1} = (D-\omega L)^{-1}((1-\omega)D + \omega U)x_k + \omega(D-\omega L)^{-1}b.$$

This is the well-known Successive Over Relaxation (SOR) method. If we apply the relaxation technique to the Gauss-Seidel iteration (1.3), we can obtain a new iterate

$$\begin{aligned} x_{k+1} &= x_k + \omega\left((D-L)^{-1}Ux_k + (D-L)^{-1}b - x_k\right) \\ &= (D-L)^{-1}\left((1-\omega)(D-L) + \omega U\right)x_k + \omega(D-L)^{-1}b. \end{aligned}$$

This leads immediately to a new relaxation method, which is named GSR method and can be described as follows

GSR Method: Given an initial point x_0, the iterative sequence $x_k, k = 1, 2, \ldots$, of the GSR method is generated by

$$x_k =(D-L)^{-1}\left((1-\omega)(D-L) + \omega U\right)x_{k-1} + \omega(D-L)^{-1}b, \tag{1.5}$$

where ω is a positive number, and the matrix

$$I(\omega) :=(D-L)^{-1}\big((1-\omega)(D-L) + \omega U\big) \tag{1.6}$$

is the iteration matrix of the GSR method.

From the above derivation, we can see that the SOR method is generated by firstly applying the relaxation technique to the Jacobi iterate and then updating the components of the new iteration one by one. However, our method is obtained through changing the orders of those two operations in the derivation of the SOR method.

Actually, we can find the GSR method is a special case of the accelerated over-relaxation (AOR) method [1]

$$(D-\gamma L)x_{k+1} = ((1-\omega)D + (\omega-\gamma)L + \omega U)x_k + \omega b$$

when $\gamma = 1$. In this sense, this paper does not propose new algorithm. Rather, our point is to reveal some important techniques that we do not realize in the study of the basic iterative method for solving linear system, and call attention to the power of the AOR method.

This paper is organized as follows. The convergence and the complexity analysis are presented in Section 2. Applications to the saddle point problem and PageRank problem are given in Section 3 to show the power of our GSR method.

2. GSR method

In this section, we study the convergence of our GSR method and consider the computational complexity. On the convergence of the GSR method, some sufficient conditions are presented when the coefficient matrix A is symmetric positive definite, strictly (irreducibly) diagonally dominant.

Whether or not the sequence $x_k, k = 0, 1, 2, \dots$, converges to the unique solution $x_* = A^{-1}b$ depends upon the spectral radius of the iteration matrix $I(\omega)$ in (1.6). Let λ be the eigenvalue of the matrix $I(\omega)$. Then

$$\begin{aligned} I(\omega) - \lambda I =&(D-L)^{-1}\left((1-\omega)(D-L) + \omega U\right) - \lambda I \\ =&(D-L)^{-1}\left((1-\lambda)(D-L) - \omega A\right) \\ =&(1-\lambda)I - \omega(D-L)^{-1}A, \end{aligned}$$

therefore, $\dfrac{1-\lambda}{\omega}$ is the eigenvalue of $(D-L)^{-1}A$, and

$$\lambda = 1 - \omega\sigma = 1 - \omega(1-\Delta), \tag{2.1}$$

where σ and Δ are the eigenvalues of $(D-L)^{-1}A$ and $(D-L)^{-1}U$, respectively. Furthermore, if the spectral radius of the iteration matrix $I(\omega)$ is less than 1, then the GSR method converges.

2.1. Convergence analysis for the linear system with special coefficient matrix

Similar as the existent relaxation methods, the convergence of our GSR method is hardly guaranteed for general coefficient matrices, however, for symmetric positive definite matrices, strictly diagonally dominant matrices or irreducibly diagonally dominant matrices, the convergence holds.

Theorem 2.1 *Suppose A is a Hermitian and positive definite matrix and $0 < \omega \le 1$, then the convergence of the GSR method holds for any initial point.*

Proof Suppose λ is an eigenvalue of the iteration matrix $I(\omega)$ and the vector x is the corresponding eigenvector, then we have

$$(D-L)^{-1}\big((1-\omega)(D-L) + \omega U\big)x = \lambda x.$$

Let x^H denote the conjugate transpose of the vector x. Through multiplying both sides of the above equation by $x^H(D-L)$, we get

$$x^H((1-\omega)(D-L)+\omega U)x = \lambda x^H(D-L)x. \tag{2.2}$$

Since A is Hermitian and positive definite, then $U = L^H$, $x^H Lx = (x^H Ux)^H$ and $x^H Dx > 0$. Denote

$$\alpha := x^H Dx,\ \beta+\gamma\mathbf{i} := x^H Lx,\ \beta-\gamma\mathbf{i} := x^H Ux, \tag{2.3}$$

where $\mathbf{i} = \sqrt{-1}$. From the equation (2.2) and the notations (2.3), we obtain

$$\begin{aligned}|\lambda|^2 =& \Big|\frac{x^H((1-\omega)(D-L)+\omega U)x}{x^H(D-L)x}\Big|^2 = \frac{|(1-\omega)(\alpha-\beta-\gamma\mathbf{i})+\omega(\beta-\gamma\mathbf{i})|^2}{|\alpha-\beta-\gamma\mathbf{i}|^2}\\ =& \frac{|\alpha-\omega\alpha-\beta+2\omega\beta-\gamma\mathbf{i}|^2}{|\alpha-\beta-\gamma\mathbf{i}|^2} = \frac{|(\alpha-\beta+2\omega\beta-\omega\alpha)^2+\gamma^2|}{|(\alpha-\beta)^2+\gamma^2|}.\end{aligned}$$

Since A is positive definite, we have

$$x^H Ax = x^H Dx - x^H Lx - x^H L^H x = \alpha - 2\beta > 0. \tag{2.4}$$

Combining the relations $0 < \omega \le 1$ and (2.4), we can get

$$\begin{aligned}&|(\alpha-\beta+2\omega\beta-\omega\alpha)^2+\gamma^2| - |(\alpha-\beta)^2+\gamma^2|\\ &= \omega(2\beta-\alpha)(2\beta\omega-\alpha\omega+2\alpha-2\beta) = \omega(2\beta-\alpha)(\alpha+(1-\omega)(\alpha-2\beta)) < 0,\end{aligned}$$

which implies $|\lambda| < 1$, thus the theorem holds. □

A strictly diagonally dominant or irreducible diagonally dominant matrix is nonsingular. The following theorem shows the convergence of the GSR method for strictly diagonally dominant or irreducibly diagonally dominant matrix A.

Theorem 2.2 *Suppose A is strictly diagonally dominant or irreducibly diagonally dominant and $0 < \omega \le 1$. Then the GSR method converges for any initial point.*

Proof The proof is by contradiction. Suppose $\lambda = a + b\mathbf{i}$ is an eigenvalue of the iteration matrix $I(\omega)$ and satisfies $|\lambda| \ge 1$, then $a^2 + b^2 \ge 1$ and

$$\det(I(\omega)-\lambda I) = \det((D-L)^{-1})\det((1-\omega-\lambda)(D-L)+\omega U) = 0. \tag{2.5}$$

Consider the following two cases:

(1) If $a < 1$, we get

$$\begin{aligned}|1-\omega-\lambda|^2-\omega^2 &= (1-\omega-a)^2+b^2-\omega^2 \ge 1+a^2+b^2-2a+2(a-1)\\ &= a^2+b^2-1 \ge 0.\end{aligned}$$

(2) If $a \ge 1$, we get

$$|1-\omega-\lambda|^2-\omega^2 = (a-1)^2+2\omega(a-1)+b^2 \ge 0.$$

Thus the relation $|1-\omega-\lambda| \ge |\omega|$ holds and the matrix $(1-\omega-\lambda)(D-L)+\omega U$ is a strictly diagonally dominant or irreducibly diagonally dominant matrix, which indicates the matrix $(1-\omega-\lambda)(D-L)+\omega U$ is nonsingular and contradicts the equation (2.5). Therefore, we have $|\lambda| < 1$

and the theorem holds. □

2.2. Computational complexity

For the SOR method, in the calculation of the $(k+1)$th iterate

$$x_{k+1} = x_k + \omega(D - \omega L)^{-1} r_k = x_k + (D/\omega - L)^{-1} r_k,$$

we need n flops to get D/ω, then n^2 flops to get $(D/\omega - L)^{-1} r_k$, and finally n flops to get x_{k+1}, thus the total number of flops is $n^2 + n + n = n^2 + 2n$. To go further, we have to calculate

$$\begin{aligned} r_{k+1} &= b - Ax_{k+1} = b - A(x_k + (D/\omega - L)^{-1} r_k) \\ &= ((1-\omega)/\omega D + U)(D/\omega - L)^{-1} r_k, \end{aligned}$$

this requires $n^2 + n + 2$ flops.

Next, we consider the number of flops required in our GSR method. In the calculation of the $(k+1)$th iterate $x_{k+1} = x_k + \omega(D - L)^{-1} r_k$, we need n^2 flops to get $(D - L)^{-1} r_k$, then n flops to get $\omega(D - L)^{-1} r_k$, and finally n flops to get x_{k+1}, thus the total number of flops is $n^2 + n + n = n^2 + 2n$. To go further, we have to calculate

$$r_{k+1} = b - Ax_{k+1} = (1-\omega) r_k + \omega U (D - L)^{-1} r_k = r_k - \omega(r_k - U(D - L)^{-1} r_k),$$

this requires $(n-1)^2 + 3n$ flops.

3. Applications

In this section, we compare the performance of the GSR method with some existing algorithms by two practical problems: the saddle point problem and PageRank problem. All our computations were performed in MATLAB 2014a on a Intel 2.5GHZ computer with 48GB memory running Windows 8.

3.1. Saddle point problems

The saddle point system is of the form

$$\begin{pmatrix} A & B \\ B^\top & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} b \\ q \end{pmatrix}, \tag{3.1}$$

where $A \in \mathbb{R}^{m \times m}$ is a symmetric positive definite matrix and $B \in \mathbb{R}^{m \times n}$ $(m \geq n)$ is a matrix of full column rank, $b \in \mathbb{R}^m$ and $q \in \mathbb{R}^n$ are two real vectors. A linear system of the form (3.1) is also named as a KKT system, or an augmented system, or an equilibrium system.

The saddle point problem is common in many scientific and engineering applications [2], such as computational fluid dynamics [3], mixed finite element method of elliptic partial differential equations [4], least squares problems [5]. And there are many iterative methods presented to solve these problems, for example, (preconditioned) Uzawa-type algorithms [6–8], the SOR-like methods and its variants [9–13], and the preconditioned Krylov subspace methods [14–16], HSS splitting method [17–19] and many references contained therein.

The linear system in (3.1) can be rewritten as following

$$\begin{pmatrix} A & B \\ -B^\top & 0 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} b \\ -q \end{pmatrix}. \tag{3.2}$$

In this subsection, we mainly compare our method with the SOR-like method in [10], which is constructed based on the matrix partition

$$\begin{pmatrix} A & B \\ -B^\top & 0 \end{pmatrix} = D_Q - L_Q - U_Q,$$

where

$$D_Q = \begin{pmatrix} A & 0 \\ 0 & Q \end{pmatrix},\ L_Q = \begin{pmatrix} 0 & 0 \\ B^\top & 0 \end{pmatrix},\ U_Q = \begin{pmatrix} 0 & -B \\ 0 & Q \end{pmatrix},$$

and Q is a nonsingular and symmetric matrix. Sequentially applying the relaxation technique and Gauss-Seidel technique to the Jacobi iterate, we can get the following SOR-like iterate [10]:

$$\begin{pmatrix} x_{k+1} \\ y_{k+1} \end{pmatrix} = (D_Q - \omega L_Q)^{-1}\big((1-\omega)D_Q + \omega U_Q\big)\begin{pmatrix} x_k \\ y_k \end{pmatrix} + \omega(D_Q - \omega L_Q)^{-1}\begin{pmatrix} b \\ q \end{pmatrix}.$$

Changing the order of applying the relaxation technique and Gauss-Seidel technique, we can get our GSR-like iterate:

$$\begin{pmatrix} x_{k+1} \\ y_{k+1} \end{pmatrix} = (D_Q - L_Q)^{-1}\big((1-\omega)(D_Q - L_Q) + \omega U_Q\big)\begin{pmatrix} x_k \\ y_k \end{pmatrix} + \omega(D_Q - L_Q)^{-1}\begin{pmatrix} b \\ q \end{pmatrix}.$$

The following theorem states the convergence of the GSR-like method for some specially chosen Q.

Theorem 3.1 *For the linear system in (3.2), suppose the matrix $Q \in \mathbb{R}^{n\times n}$ is nonsingular and symmetric, the eigenvalues of the matrix $J := Q^{-1}B^\top A^{-1}B$ are all positive, then the GSR-like method for the saddle point problem is convergent when the relaxation factor ω satisfies*

$$0 < \omega < \min\big\{\frac{2}{\mu_{\max}}, 2\big\},$$

where $\mu_{\max}$ is the maximal eigenvalue of the matrix J.

Proof Suppose λ is the eigenvalue of the iteration matrix

$$I_Q(\omega) := (D_Q - L_Q)^{-1}\big((1-\omega)(D_Q - L_Q) + \omega U_Q\big),$$

then

$$\begin{aligned} I_Q(\omega) &= (1-\omega)I + \omega(D_Q - L_Q)^{-1}U_Q \\ &= (1-\omega)I + \omega\begin{pmatrix} A^{-1} & 0 \\ Q^{-1}B^\top A^{-1} & Q^{-1} \end{pmatrix}\begin{pmatrix} 0 & -B \\ 0 & Q \end{pmatrix} \\ &= (1-\omega)I + \omega\begin{pmatrix} 0 & -A^{-1}B \\ 0 & -Q^{-1}B^\top A^{-1}B + I \end{pmatrix}. \end{aligned}$$

Let μ be the eigenvalue of the matrix $Q^{-1}B^{\top}A^{-1}B$. Then we can get the following relation

$$\lambda = (1-\omega) + \omega(-\mu+1) = 1-\omega\mu \text{ or } \lambda = 1-\omega. \tag{3.3}$$

Thus, we can get

$$|\lambda| < 1 \Longleftrightarrow |1-\omega\mu| < 1, |1-\omega| < 1 \Longleftrightarrow 0 < \omega < \min\left\{\frac{2}{\mu_{\max}}, 2\right\},$$

and the theorem holds. □

Next, we consider the strategies for choosing the optimal relaxation factor. To find an optimal relaxation factor, we need to solve the following optimization problem

$$\omega = \arg\min_{\omega\in R}\max_{\mu\in\lambda(J)}\{|1-\omega|, |1-\mu\omega|\} = \arg\min_{\omega\in R}\max_{\mu\in\{\lambda(J),1\}}\{|1-\mu\omega|\}.$$

From the properties of Chebyshev polynomials, we can get the following optimal relaxation factor ω and the corresponding spectral radius of the iteration matrix:

$$\omega = \frac{2}{\widehat{\mu}_{\min} + \widehat{\mu}_{\max}}, \quad \rho = \frac{\widehat{\mu}_{\max} - \widehat{\mu}_{\min}}{\widehat{\mu}_{\max} + \widehat{\mu}_{\min}} < 1.$$

where $\widehat{\mu}_{\min} := \min\{1, \mu_{\min}\}$, $\widehat{\mu}_{\max} := \max\{1, \mu_{\max}\}$, and $\mu_{\min}$ and $\mu_{\max}$ are the minimal and maximal eigenvalues of the matrix J, respectively.

Finally, we compare our GSR-like method with the SOR-like method through an example in [20]. Consider the linear system in (3.2), where

$$A = \begin{pmatrix} I_p\otimes T + T\otimes I_p & 0 \\ 0 & I_p\otimes T + T\otimes I_p \end{pmatrix} \in \mathbb{R}^{2p^2\times 2p^2},$$

$$B = \begin{pmatrix} I_p\otimes F \\ F\otimes I_p \end{pmatrix} \in \mathbb{R}^{2p^2\times p^2},$$

$$T = \frac{1}{h^2}\text{Tridiag}(-1,2,-1) \in \mathbb{R}^{p\times p}, \quad F = \frac{1}{h}\text{Tridiag}(-1,1,0) \in \mathbb{R}^{p\times p},$$

$$b = (1,\ldots,1)^{\top} \in \mathbb{R}^{2p^2}, q = (1,\ldots,1)^{\top} \in \mathbb{R}^{p^2},$$

$h = 1/(p+1)$, Tridiag(a,b,c) is a matrix in which elements of main diagonal are b, elements of the first diagonal are c, elements of the negative first diagonal are a, and $\otimes$ is the Kronecker product.

Dimension	Method	IT	CPU	RES
3072	SOR-like	414	8.4108	9.795e-09
	GSR-like	340	7.1384	9.639e-09
4800	SOR-like	513	18.1664	9.950e-09
	GSR-like	423	15.7483	9.734e-09

Table 2 GSR-like method VS SOR-like method when Q =Tridiag$(B^{\top}\tilde{A}^{-1}B)$, $\tilde{A}$ =Tridiag(A)

The numerical results are summarized in Tables 2 and 3. SOR-like represents the SOR-like method with the optimal relaxation factor $\omega = \frac{2\sqrt{\mu_{\max}}-1}{\mu_{\max}}$ (see [10]). GSR-like represents the method with the optimal relaxation factor $\omega = \frac{2}{\widehat{\mu}_{\min}+\widehat{\mu}_{\max}}$. All methods are started from the initial vector $x_0 = (0, \dots, 0)^\top \in \mathbb{R}^{2p^2}, y_0 = (0, \dots, 0)^\top \in \mathbb{R}^{p^2}$.

Dimension	Method	IT	CPU	RES
3072	SOR-like	390	7.9715	9.598e-09
	GSR-like	247	5.4232	9.942e-09
4800	SOR-like	483	17.3768	9.911e-09
	GSR-like	305	11.7825	9.960e-09

Table 3 GSR-like method VS SOR-like method when Q =Tridiag$(B^\top A^{-1}B)$

From Tables 2 and 3, we can see that the GSR-like method is more efficient than the SOR-like method with the optimal relaxation factor.

3.2. PageRank problems

The PageRank problem [21] is a well-known web ranking problem and can be formulated as the linear system $(I - \alpha P)x = (1 - \alpha)v$ (see [22]), where P is a column-stochastic matrix, $\alpha < 1$ is the damping parameter, and v is a given probability vector.

Many numerical methods [23–28] are proposed to solve the PageRank problem. As pointed out in [29], basic stationary schemes, such as Jacobi iterative method, Gauss-Seidel iterative method and SOR method, are still very competitive for solving the linear system arising in the PageRank problem comparing with the Krylov subspace methods. The convergence and performance of the SOR method for solving the PageRank problem is studied in [30].

In this subsection, we consider to solve the PageRank problem by our GSR method, which is based on the splitting $I - \alpha P = D - L - U$. The following theorem presents the convergence of the GSR method if choosing the relaxation factor in an interval depending on the damping parameter.

Theorem 3.2 *If $\omega \in (0, \frac{2}{1+\alpha})$, then the GSR method for the PageRank problem is convergent for any initial point.*

Proof Suppose λ is the eigenvalue of the iteration matrix $I(\omega)$, then we have

$$\begin{aligned}
&\det((D - L)^{-1}((1 - \omega)(D - L) + \omega U) - \lambda I)\\
&\quad = \det((D - L)^{-1}) \det(((1 - \omega)(D - L) + \omega U) - \lambda(D - L))\\
&\quad = \det((D - L)^{-1}) \det((1 - \omega - \lambda)(D - L) + \omega U) = 0,
\end{aligned}$$

which implies the matrix $(1 - \omega - \lambda)(D - L) + \omega U$ is necessarily not strictly column diagonally dominant. Therefore, there exists one column, say the ith column, we have

$$|1 - \omega - \lambda|(1 - \alpha(1 - u - l)) \le |1 - \omega - \lambda|\alpha l + \omega\alpha u$$

or

$$\frac{|1 - \omega - \lambda|}{\omega} \le \frac{\alpha u}{1 - \alpha(1 - u - l) - \alpha l} = \frac{\alpha u}{1 - \alpha(1 - u)},$$

where u and l are the sum of the entries above and, respectively, below the diagonal in the ith column of the matrix P. The function $f(u) := \frac{\alpha u}{1-\alpha(1-u)}$ is monotonically increasing, then we have $\frac{|1-\omega-\lambda|}{\omega} \le \alpha$. Thus, the eigenvalue λ satisfies $|\lambda| \le \alpha\omega + |1-\omega|$. Therefore, if $0 < \omega < \frac{2}{1+\alpha}$, we can get $\alpha\omega + |1-\omega| < 1$, and the convergence of the GSR method is guaranteed. $\square$

We consider the performance of our GSR method through some test Web matrices: Ca-GrQc, P2P-Gnutella08 and Ca-HepTh. SOR represents the SOR method with the optimal relaxation factor in $(0, \frac{2}{1+\alpha})$. GSR represents the GSR method with the optimal relaxation factor in $(0, \frac{2}{1+\alpha})$. All methods are started from the initial point $x_0 = (0, \ldots, 0)^\top$.

Matrix	Method	IT	CPU	RES
P2P-Gnutella08	SOR	12	0.0237	5.657e-09
	GSR	14	0.0226	2.283e-09
Ca-HepTh	SOR	60	0.1037	7.792e-09
	GSR	64	0.0837	8.304e-09
Ca-GrQc	SOR	59	0.0456	8.799e-09
	GSR	63	0.0360	9.117e-09

Table 4 GSR method VS SOR method when $\alpha = 0.85$

From Table 4, we can see that the number of iterations and the elapsed time by the GSR method and those by the SOR method with the optimal relaxation factor are almost same.

4. Conclusion

In this paper, we present a new relaxation method for solving linear systems, prove its convergence, and apply it to solve two important practical problems: the saddle point problem and the PageRank problem.

Acknowledgements We thank the referees for their time and comments.

References

[1] A. HADJIDIMOS. *Accelerated overrelaxation method*. Math. Comp., 1978, **32**(141): 149–157.

[2] M. BENZI, G. H. GOLUB, J. LIESEN. *Numerical solution of saddle point problems*. Acta Numer., 2005, **14**: 1–137.

[3] H. C. ELMAN. *Preconditioners for saddle point problems arising in computational fluid dynamics*. Appl. Numer. Math., 2002, **43**(1-2): 75–89.

[4] F. BREZZI, M. FORTIN. *Mixed and Hybrid Finite Element Methods, Springer Series in Computational Mathematics*. Springer-Verlag, New York, 1991.

[5] Å. BJÖRCK. *Numerical Methods for Least Squares Problems*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1996.

[6] Zhongzhi BAI, Zengqi WANG. *On parameterized inexact Uzawa methods for generalized saddle point problems*. Linear Algebra Appl., 2008, **428**(11-12): 2900–2932.

[7] J. H. BRAMBLE, J. E. PASCIAK, A. T. VASSILEV. *Analysis of the inexact Uzawa algorithm for saddle point problems*. SIAM J. Numer. Anal., 1997, **34**(3): 1072–1092.

[8] H. C. ELMAN, G. H. GOLUB. *Inexact and preconditioned Uzawa algorithms for saddle point problems*. SIAM J. Numer. Anal., 1994, **31**(6): 1645–1661.

[9] Zhongzhi BAI, B. N. PARLETT, Zengqi WANG. *On generalized successive overrelaxation methods for augmented linear systems*. Numer. Math., 2005, **102**(1): 1–38.

[10] G. H. GOLUB, Xiaonan WU, Jinyun YUAN. *SOR-like methods for augmented systems*. BIT, 2001, **41**(1): 71–85.

[11] Changjun LI, Baojia LI, D. J. EVANS. *A generalized successive overrelaxation method for least squares problems*. BIT, 1998, **38**(2): 347–355.

[12] Guofeng ZHANG, Qunhua LU. *On generalized symmetric SOR method for augmented systems*. J. Comput. Appl. Math., 2008, **219**(1): 51–58.

[13] Bing ZHENG, Ke WANG, Yujiang WU. *SSOR-like methods for saddle point problems*. Int. J. Comput. Math., 2009, **86**(8): 1405–1423.

[14] Zhongzhi BAI. *Structured preconditioners for nonsingular matrices of block two-by-two structures*. Math. Comp., 2006, **75**(254): 791–815.

[15] Zhongzhi BAI, Guiqing LI. *Restrictively preconditioned conjugate gradient methods for systems of linear equations*. IMA J. Numer. Anal., 2003, **23**(4): 561–580.

[16] M. BENZI, G. H. GOLUB. *A preconditioner for generalized saddle point problems*. SIAM J. Matrix Anal. Appl., 2004, **26**(1): 20–41.

[17] Zhongzhi BAI, G. H. GOLUB. *Accelerated Hermitian and skew-Hermitian splitting iteration methods for saddle-point problems*. IMA J. Numer. Anal., 2007, **27**(1): 1–23.

[18] Zhongzhi BAI, G. H. GOLUB, M. K. NG. *Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems*. SIAM J. Matrix Anal. Appl., 2003, **24**(3): 603–626.

[19] M. BENZI, M. J. GANDER, G. H. GOLUB. *Optimization of the Hermitian and skew-Hermitian splitting iteration for saddle-point problems*. BIT, 2003, **43**(5): 881–900.

[20] Zhongzhi BAI, G. H. GOLUB, Jianyu PAN. *Preconditioned Hermitian and skew-Hermitian splitting methods for non-Hermitian positive semidefinite linear systems*. Numer. Math., 2004, **98**(1): 1–32.

[21] L. PAGE, S. BRIN, R. MOTWAMI, et al. *The PageRank Citation Ranking: Bring Order to the Web*. Technical report, Computer Science Department, Stanford University, 1998.

[22] A. N. LANGVILLE, C. D. MEYER. *Googles PageRank and Beyond: The Science of Search Engine Rankings*. Princeton University Press, Princeton, NJ, 2006.

[23] Zhongzhi BAI. *On convergence of the inner-outer iteration method for computing PageRank*. Numer. Algebra Control Optim., 2012, **2**(4): 855–862.

[24] C. BREZINSKI, M. REDIVO-ZAGLIA. *The PageRank vector: properties, computation, approximation, and acceleration*. SIAM J. Matrix Anal. Appl., 2006, **28**(2): 551–575.

[25] G. H. GOLUB, C. GREIF. *An Arnoldi-type algorithm for computing PageRank*. BIT, 2006, **46**(4): 759–771.

[26] Gang WU, Yanchun WANG, Xiaoqing JIN. *A preconditioned and shifted GMRES algorithm for the PageRank problem with multiple damping factors*. SIAM J. Sci. Comput., 2012, **34**(5): A2558–A2575.

[27] Gang WU, Yimin WEI. *A Power-Arnoldi algorithm for computing PageRank*. Numer. Linear Algebra Appl., 2007, **14**(7): 521–546.

[28] Junfeng YIN, Guojian YIN, M. NG. *On adaptively accelerated Arnoldi method for computing PageRank*. Numer. Linear Algebra Appl., 2012, **19**(1): 73–85.

[29] C. P. LEE, G. H. GOLUB, S. A. ZENIOS. *A two-stage algorithm for computing PageRank and multistage generalizations*. Internet Math., 2007, **4**(4): 299–327.

[30] C. GREIF, D. KUROKAWA. *A note on the convergence of SOR for the PageRank problem*. SIAM J. Sci. Comput., 2011, **33**(6): 3201–3209.