Research Article

A New GMRES(m) Method for Markov Chains

Bing-Yuan Pu,^{1,2} Ting-Zhu Huang,¹ and Chun Wen¹

¹ School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu 611731, China

² Department of Basic Courses, Chengdu Textile College, Chengdu 611731, China

Correspondence should be addressed to Bing-Yuan Pu; skypuby@163.com

Received 11 August 2013; Accepted 2 September 2013

Academic Editor: Zheng-Guang Wu

Copyright © 2013 Bing-Yuan Pu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a class of new accelerated restarted GMRES method for calculating the stationary probability vector of an irreducible Markov chain. We focus on the mechanism of this new hybrid method by showing how to periodically combine the GMRES and vector extrapolation method into a much efficient one for improving the convergence rate in Markov chain problems. Numerical experiments are carried out to demonstrate the efficiency of our new algorithm on several typical Markov chain problems.

1. Introduction

The Markov chain is a very robust tool for studying stochastic systems overtime and is in a wide range of applications including queueing systems [1, 2], computer and communication systems [3], information retrieval, and Web ranking [4–7]. For both discrete and continuous Markov chains, it has always been one of the central work to get their numerical solutions by adopting appropriate methods.

In this paper, we consider a class of Krylov subspace methods, restarted GMRES (GMRES(m)) method, for the numerical solutions of the stationary probability vector of an irreducible Markov chain. Let $B = (b_{ij}) \in \mathbb{R}^{m \times n}$ be a column stochastic matrix; that is, $0 \leq b_{ij} \leq 1$ ($\forall i, j$) and $e^T B = e^T$, with e being the column vector of all ones. We seek the stationary vector $x \in \mathbb{R}^n$ that satisfies

$$Bx = x, \quad x_i \geq 0 \quad (\forall i), \quad \|x\| = 1. \quad (1)$$

By the Perron-Frobenius theorem [8, 9], if B is irreducible, then there exists a unique solution x to (1), which is strictly positive ($x_i > 0, \forall i$).

Equation (1) is equivalent to the following singular linear problem:

$$Ax = 0, \quad x_i \geq 0, \quad \forall i, \quad \|x\| = 1, \quad (2)$$

where $A = I - B$, with I being the identity matrix, and A is a singular M -matrix with the diagonal elements being negative column sums of its off-diagonal elements.

Now there are many numerical methods for solving (2), such as the traditional iterative methods, the power method or weighted Jacobi method [5, 10–12], the aggregation/disaggregation algorithm [13–16], and the Krylov subspace methods like the Arnoldi method and the GMRES [3, 17–21]. However, these iteration methods for calculating x may converge very slowly when the subdominant eigenvalue of B satisfies $\lambda_2 \sim 1$ [3, 11]. Thus, it becomes quite required to accelerate the calculation in Markov chain problems.

In this paper, we consider the restarted GMRES method and propose a new way to accelerate its numerical calculation by use of the polynomial-type vector extrapolation methods. In fact, seeking the vector extrapolation method as the accelerator is very popular; see [5, 12, 22] for details. In discussing this issue, we start from the mechanisms of the GMRES(m) method and vector extrapolation method and then illustrate how to periodically knit these two methods. In numerical experiments, the proposed extrapolation-accelerated GMRES(m) methods are tested on several Markov chain problems, and the experimental results show its effectiveness.

This paper is organized as follows. Section 2 briefly provides the mechanism of the GMRES methods and some acceleration ones. In Section 3, we first review the vector extrapolation method and then consider how to periodically

combine the extrapolation method with the GMRES method for numerical calculation of Markov chains. Section 4 provides experimental evidence of the effectiveness of our approach. Section 5 summarizes the thesis and points out the direction of our future research.

2. Background

In this section, we briefly introduce the mechanism of the GMRES methods and describe some existing modifications to the standard GMRES aimed at accelerating its convergence.

The GMRES method, proposed by Saad and Schultz in [20], is a popular method for the iterative solution of sparse linear systems with an unsymmetrical matrix,

$$Ax = b, \quad A \in \mathbb{R}^{n \times n}. \quad (3)$$

If x_0 is an initial guess for the true solution of (3) and $r_0 = b - Ax_0$ is the initial residual, we have the equivalent system:

$$Az = r_0, \quad A \in \mathbb{R}^{n \times n}, \quad (4)$$

where $x = x_0 + z$. Let $\mathcal{K}_k$ be the *Krylov subspace*:

$$\mathcal{K}_k \equiv \text{span} \{r_0, Ar_0, \dots, A^{k-1}r_0\}. \quad (5)$$

GMRES is to find an approximate solution

$$x_k = x_0 + z_k \quad \text{with } z_k \in \mathcal{K}_k, \quad (6)$$

such that

$$\|b - Ax_k\|_2 = \min_{x \in x_0 + \mathcal{K}_k} \|b - Ax\|_2 = \min_{z \in \mathcal{K}_k} \|r_0 - Az\|_2. \quad (7)$$

Here, $\|\cdot\|_2$ denotes the Euclidean norm on $\mathbb{R}^n$, as well as the associated-induced matrix norm on $\mathbb{R}^{n \times n}$.

GMRES is often referred to as an “optimal” method in the sense that it finds the approximate solution in the Krylov subspace that minimizes the residual [20]. However, the amount of storage and computational work grow quadratically with the number of steps. So the restarted version of the algorithm is often used as suggested in [20]. In restarted GMRES (GMRES(m)), the method is restarted after each cycle of m iteration steps, and the current approximate solution becomes the new initial guess for the next m iterates. The mechanism of the GMRES(m) for Markov chains as system (2) is described as follows. For more details, see [20, 23].

Algorithm 1. The Restarted GMRES.

(1) Start: choose x_0, m, tol

$$r_0 \leftarrow -Ax_0, \quad v_1 \leftarrow \frac{r_0}{\|r_0\|_2}, \quad \beta = \|r_0\|_2. \quad (8)$$

(2) For $j = 1, 2, \dots, m$, do

$$h_{ij} = (Av_j, v_i), \quad i = 1, 2, \dots, j. \quad (9)$$

$$\widehat{v_{j+1}} = Av_j - \sum_{i=1}^j h_{ij}v_i,$$

$$h_{j+1,j} = \|\widehat{v_{j+1}}\|_2, \quad v_{j+1} = \frac{\widehat{v_{j+1}}}{h_{j+1,j}}.$$

(3) Form the approximate solution: $x_m = x_0 + V_m y_m$, where y_m minimizes $\|\beta e_1 - \bar{H}_m y\|_2$, $y \in \mathbb{R}^m$.

(4) Restart: while $\|r_{k+1}\|_2 > \text{tol}$, do

$$r_m = -Ax_m \quad (10)$$

$x_0 \leftarrow x_m, \quad v_1 = r_m / \|r_m\|_2$ and go to step 2.

In general, the convergence rate depends on the restart parameter [24, 25]. Even in the situation in which the appropriate restart parameter results in the satisfactory convergence, the convergence behavior may not be the optimal; since an iterate is restarted, the current approximation space is discarded at each restart, and the orthogonality between approximation spaces generated at successive restarts is not preserved at each restart. For that reason, slow convergence can occur. Therefore, it is necessary to develop more efficient algorithms for enhancing the robustness of restarted GMRES, and thereof now its improvements and modifications continue to receive considerable attentions. Augmented methods are a class of acceleration techniques which seek to avoid stalling by improving information in GMRES at the time of the restart. These acceleration methods are presented by Morgan [19, 26], Saad [27] and Chapman and Saad [28]. The precondition technique is also often used to accelerate the numerical calculation of GMRES; see [17, 23, 29–32] for more details. Meanwhile, this paper focuses on undertaking the similar works, that is, speeding up the process and improving the robustness of the restarted GMRES.

3. The Main Algorithm and Practical Implementations

Now let us first present the rationale and theory behind vector extrapolation method. When considering the solutions of systems of linear or nonlinear equations by an iterative method, a sequence of vectors (approximation solutions) is yielded. As the classical iteration process may converge slowly, extrapolation strategy can often be applied to enhance its convergence rates. A detailed review of extrapolation methods can be found in the works of Smith et al. [22], Sidi [12], and Jbilou and Sadok [33]. Now there are mainly two classes of vector extrapolation methods: (1) polynomial-type methods, namely, the minimal polynomial extrapolation (MPE) of Cabay and Jackson [34], the reduced rank extrapolation (RRE) of Eddy [35] and Mešina [36], and the modified minimal polynomial extrapolation (MMPE) method of Sidi et al. [37], Brezinski [38], and Pugachev [39]; and (2) epsilon algorithms, namely, the topological epsilon algorithm of Brezinski [38] and the scalar and vector epsilon algorithms of Wynn [40]. Numerical experiences have illustrated that the polynomial type vector extrapolation methods are in general more economical than the epsilon ones, with respect to the computing time and storage requirements. And for the special touch, Kamvar et al. have recently proposed a new extrapolation, quadratic extrapolation, for computing the PageRank in [5], and Sidi [12] has generalized this method (the generalization one is marked by GQE) and proved that the resulting generalization is very closely related to MPE.

Therefore, we focus on the two polynomial extrapolation methods, namely, RRE and GQE, in this paper. Now let us display the implementations of these two algorithms. For more details, we refer the reader to the paper [12] by Sidi.

Vector extrapolation algorithms are derived by considering vector sequences x_i , generated from a fixed-point iterative method of the following form:

$$x_{n+1} = Ax_n + b, \quad n = 0, 1, \dots, \quad (11)$$

where A is a fixed $N \times N$ matrix, b is a fixed N -dimensional vector, and x_0 is an initial vector.

Suppose that we have produced a sequence of iterates $\{x_i\}_{i=1}^{\infty}$, where $x_i \geq 0$. Then at the k th outer iteration, let

$$X = [x_k, x_{k-1}, \dots, x_{k-m+1}] \in \mathbb{R}^{n \times m} \quad (12)$$

be a matrix consisting of the last m iterates with x_k being the newest, where m is called the window size in usual. It is evident that X has the following properties:

$$x_i \geq 0, \quad \|x_i\|_1 = 1, \quad i = 1, 2, \dots \quad (13)$$

The problem to be solved is transformed into obtaining a vector z satisfying $\sum_{i=1}^m z_i = 1$, and thus we have an updated probability vector

$$\hat{x}_k = Xz = z_1 x_k + z_2 x_{k-1} + \dots + z_m x_{k-m+1}, \quad (14)$$

a linear combination of the last m iterates. Assume $\rho(A) < 1$, which results in the unique solution s of the linear system

$$x = Ax + b. \quad (15)$$

As illustrated in [12], the efficient implementations of the vector extrapolation methods (RRE and GQE) are presented as follows.

Algorithm 2. The Reduced Rank Extrapolation Method (RRE).

- (1) Input vectors $x_0, x_1, \dots, x_{k+1}$.
- (2) Compute $u_i = x_{i+1} - x_i$, $i = 0, 1, \dots, k$, set $U_k = [u_0, u_1, \dots, u_k]$, and compute the QR-factorization of U_k , namely, $U_k = Q_k R_k$.
- (3) Solve the linear system $R_k^T R_k d = e$, $d = [d_0, d_1, \dots, d_k]^T$, $e = [1, 1, \dots, 1]^T \in \mathbb{R}^{k+1}$. It amounts to solving two triangular systems: $R_k^T a = e$ for a and $R_k d = a$ for d . Set $\lambda = 1 / \sum_{i=0}^k d_i$, and compute $\gamma = [\gamma_0, \gamma_1, \dots, \gamma_k]^T$, with $\gamma_i = \lambda d_i$, $i = 0, 1, \dots, k$.
- (4) Compute $\xi = [\xi_0, \xi_1, \dots, \xi_{k-1}]^T$ by $\xi_0 = 1 - \gamma_0$, $\xi_j = \xi_{j-1} - \gamma_j$, $j = 1, 2, \dots, k-1$.
- (5) Compute $\eta = [\eta_0, \eta_1, \dots, \eta_{k-1}]^T = R_{k-1} \xi$. Compute the RRE approximation for the exact solution of system (15) by $\hat{x}_{k+1} = x_0 + Q_{k-1} \eta = x_0 + \sum_{i=0}^{k-1} \eta_i q_i$.

Algorithm 3. The Generalization of Quadratic Extrapolation (GQE).

- (1) Input vectors $x_0, x_1, \dots, x_{k+1}$.
- (2) Compute $u_i = x_{i+1} - x_0$, $i = 0, 1, \dots, k$, set $U_k = [u_0, u_1, \dots, u_k]$, and compute the QR-factorization of U_k , namely, $U_k = Q_k R_k$.
- (3) Solve the linear system $R_{k-1} d = -Q_{k-1}^T u_k$, $d = [d_0, d_1, \dots, d_k]^T$.
- (4) Set $d_k = 1$ and compute $c = [c_0, c_1, \dots, c_k]^T$ by $c_i = \sum_{j=i}^k d_j$, $i = 0, 1, \dots, k$ (thus, $c_k = d_k = 1$).
- (5) Compute $\hat{x}_{k+1} = (\sum_{i=0}^k c_i) x_0 + Q_k (R_k c)$.

Clearly, Algorithms 2 and 3 share the common feature: they both contain a QR factorization in step 2 for $U_k = Q_k R_k$, where $Q_k \in \mathbb{R}^{n \times (k+1)}$ is unitary and $R_k \in \mathbb{R}^{(k+1) \times (k+1)}$ is an upper triangular matrix with positive diagonal elements. In addition, the QR-factorization is always carried out inexpensively by applying the modified Gram-Schmidt process (MGS) to the vectors $u_0, u_1, \dots, u_k$; see [12] for more details. The MGS algorithm is given as follows.

MGS Algorithm

- (1) Compute $r_{00} = (u_0, u_0)^{1/2}$, and set $q_0 = u_0 / r_{00}$.
- (2) For $j = 1 : k$, set $u_j^{(0)} = u_j$.
- (3) For $i = 1 : j$, compute $r_{ij} = (q_i, u_j^{(i)})$ and $u_j^{(i+1)} = u_j^{(i)} - r_{ij} q_i$.
- (4) Compute $r_{jj} = (u_j^{(j)}, u_j^{(j)})^{1/2}$ and $q_j = u_j^{(j)} / r_{jj}$.

As previously mentioned, the restarting of GMRES may entail the slow convergence since the current approximation space is discarded at each restart cycle. Meanwhile, the change of the restart parameter m will affect the number of iterations and also the execution time in the run time of GMRES. Furthermore, the increasing m will decrease the number of iteration needed to converge while increasing the computing work and storage amount required per iteration, especially for large systems of equations, like large-scale Markov chains, and PageRank computation in information retrieval. So it is very necessary to enhance the robustness of GMRES(m). And our work falls into the category of accelerating techniques. Next, we will discuss the idea and implementation of the new method.

Recall that in restarted GMRES, once the Krylov subspace reaches dimension m , the current approximate solution becomes the new initial guess for the next m iterations. The motivation of our new method comes from the successive restarts every m iterations. To be specific, we will implement the current approximation, after m iterations in a restart cycle, as the initial vector for the extrapolation procedure. And furthermore, the resulting vector by extrapolation method, in turn, can be the new improved starting vector for the next m iterations by GMRES(m). In summary, the mechanism of our new restarted GMRES method can be characterized as follows.

TABLE 1: Numerical results of the three algorithms for one-dimensional Markov chain with identical transition rates.

Rest	GMRES			RGMRES			GGMRES		
	CPU	it	Res	CPU	it	Res	CPU	it	Res
5	0.1458	227	$9.9857e-7$	0.2071	97	$1.0240e-6$	0.0544	32	$5.6827e-6$
10	0.2708	78	$9.7092e-7$	0.1832	76	$9.8590e-7$	0.0612	30	$4.0731e-6$
15	0.5075	59	$9.9114e-7$	0.1317	40	$1.0653e-6$	0.0752	23	$2.2496e-6$
20	0.6750	45	$9.8919e-7$	0.1762	33	$1.0609e-6$	0.1113	21	$1.5214e-6$
30	0.6617	23	$9.2978e-7$	0.1963	18	$8.5427e-7$	0.1219	12	$2.3510e-6$

Algorithm 4. The Extrapolation-Accelerated GMRES(m) (EGMRES)

- (1) Choose x_0 , m , n , tol, and set $k = 1$.
- (2) Compute x_k using Algorithm 1 (GMRES(m)).
- (3) Set $X \leftarrow [x_{(j)}, x_{(j-1)}, \dots, x_{(j-n+1)}]$.
- (4) Compute $\hat{x}_k$ from X by applying Algorithm 2 (RRE) and Algorithm 3 (GQE), respectively.
- (5) If $\|A\hat{x}_k\|_1 > \|Ax_k\|_1$, then $\hat{x}_k \leftarrow x_k$.
- (6) Check convergence. If $\|A\hat{x}_k - \hat{x}_k\|_1 / \|\hat{x}_k\|_1 < \text{tol}$, stop. Otherwise, set $k \leftarrow k + 1$ and go to step 2.

Note that, in step 1 tol is the user described tolerance for residual vector 1-norms by GMRES(m), m is the restart number in GMRES, and n is the window size for extrapolation. From Algorithms 2 and 3, both the RRE and GQE methods require $n + 2$ vectors $x_0, x_1, \dots, x_{n+1}$ as inputs. For instance, when $n = 2$, four approximate vectors are needed as input in the extrapolation procedure. Step 5 is to check the efficiency of extrapolation strategy in current iterations. Step 6 is devised to determine when to flip flop between the extrapolation procedure and GMRES procedure. The performances of our new algorithm and comparisons with the other original algorithms will be discussed in detail below.

4. Numerical Results in Markov Chain Problems

In this section, we will report the numerical experiments to examine the efficiency of our new accelerated algorithm in numerical solution of stationary probability vector for Markov chains. Three typical Markov chain problems have been used in our experiments. All the numerical results are obtained with a MATLAB R2010a implementation on Windows 8 with 2.5 GZ i5 processor and 4 GB memory.

For the sake of justice, the same starting vector $x_0 = 1/n \times e$, with e being the column vector of all ones, is used for all the algorithms. All the iterations are terminated when $\|A\hat{x}_k - \hat{x}_k\|_1 / \|\hat{x}_k\|_1 < \text{tol}$, where $\hat{x}_k$ are the approximations obtained by the current iteration. For convenience, in all the tables and figures below, we have abbreviated the RRE-GMRES algorithm and the GQE-GMRES algorithm as RGMRES and GGMRES, respectively. We denote by “rest” the restart number in GMRES, “CPU” the CPU time used in seconds, “it” the iteration counts, and “Res” the residual 1-norms.

FIGURE 1: Graph for one-dimensional Markov chain with identical transition rates.

Example 1. In this example, we compare the GMRES(m) algorithm, the RGMRES algorithm, and the GGMRES algorithm for the Markov chain problem. This test problem is a one-dimensional (1D) Markov chains often resulting from a $M/M/1$ queueing system. The simplest graph of this problem is displayed in Figure 1, where the transition rates are identical. Numerical results are presented in Table 1.

It is easy to see from Table 1 that the accelerated GMRES is more effective than the unaccelerated one, both in terms of CPU time and the number of iteration and regardless of the restart number size for GMRES. In particular, the GMRES accelerated by GQE (GGMRES) performs best in most cases. For instance, when the restart number is 20, in comparison with the results of unaccelerated GMRES, the CPU time has been saved by 74% for FGMRES and 85% for GGMRES, and the iteration count has been reduced by 27% for RGMRES and 54% for GGMRES. In Figure 2, we compare the converging speed among the three algorithms. It is clear that the convergence rate has been enhanced greatly when using the accelerating technique in GMRES, and especially that the GGMRES has faster convergence speed than the other algorithms for meeting the convergence precision quickly.

Example 2. This test problem is displayed in Figure 3, which is a 1D chain with uniform weights, except for two weak links with weight μ in the middle of the chain. This is a typical nearly completely decomposable (NCD) Markov chain problem, and it has been discussed in [41–43]. We run the GMRES algorithm, the accelerated GMRES algorithm by vector extrapolation method, namely, RGMRES and GGMRES, on this problem, when the window size in extrapolation takes different values. All the algorithms will be stopped as soon as the residual 2-norms are below $\text{tol} = 10^{-7}$. Let $\mu = 1e-3$ in the numerical simulation, and the experimental results are listed in Table 2.

It is seen from Table 2 that the accelerated GMRES methods by vector extrapolation method perform better than the unaccelerated GMRES method both in terms of iteration counts and CPU time, regardless of their window

TABLE 2: Numeric comparisons of three algorithms for the uniform chain with two weak links in the middle ($\mu = 1e - 3$).

	Window size							
	2		3		4		5	
	CPU	it	CPU	it	CPU	it	CPU	it
GMRES	0.2708	78	0.4917	98	0.4807	98	0.4763	98
RGMRES	0.1832	76	0.1924	76	0.1526	61	0.1593	65
GGMRES	0.0612	30	0.0915	40	0.0679	28	0.0927	38

TABLE 3: Numerical results of the three algorithms with various problem sizes for uniform 2D lattice.

N	GMRES			RGMRES			GGMRES		
	CPU	it	Res	CPU	it	Res	CPU	it	Res
50	0.0393	20	$9.2658e - 7$	0.0515	20	$9.2658e - 7$	0.0423	17	$7.4865e - 7$
100	0.3117	55	$9.6876e - 7$	0.2056	27	$8.8903e - 7$	0.1850	25	$1.5432e - 6$
200	1.6468	71	$9.5389e - 7$	1.5973	56	$9.9972e - 7$	1.104	35	$2.1738e - 6$
300	8.0600	118	$9.7536e - 7$	7.2827	77	$1.3757e - 6$	5.5989	59	$1.9483e - 6$

FIGURE 2: Convergence curves of the three algorithms for Example 1, $\text{tol} = 10^{-7}$. FIGURE 3: Graph for a uniform chain with two weak links in the middle ($\mu = 1e - 3$).

size. Obviously the GMRES method accelerated by GQE outperforms the other two methods. For instance, when the widow size is 4, the iteration number in GGMRES is only 46% of that in RGMRES and less than 28% of that in GMRES. Especially from the view point of CPU time, the GGMRES method has more obvious advantages than the GMRES method and the RGMRES method, for it reduces the convergence time by 55% compared with RGMRES method and by nearly 86% compared with GMRES method.

FIGURE 4: Graph for a 2D lattice with uniform weights.

Example 3. This problem is a 2D lattice (grid) with uniform weights, which has been discussed in [43, 44]. Aggregates of size 3×3 are used and the gauge variables are defined on the lattice edges and are scalars valued as one, as shown in Figure 4.

In this problem, we run the GMRES, RGMRES and GGMRES, and compare their convergence rates while changing the problem size, with the restart number being 10 and the window size being 2. Numerical results are presented in Table 3. The numerical simulation given in Table 3 for the 2D lattice problem clearly demonstrates the effectiveness of acceleration by vector extrapolation methods. It is seen that the RGMRES algorithm converges faster than the GMRES method, while the GGMRES algorithm performs the best.

5. Conclusions

In this paper, we have presented a new GMRES method accelerated by vector extrapolation techniques to get the numerical solutions of the stationary probability vector of an irreducible Markov chain, using vector extrapolation techniques. Experimental results on several typical Markov chain problems demonstrate that the extrapolation method is an attractive option for accelerating the convergence of calculation of Markov chain, especially with the GQE (proposed by Sidi in [12]) being the accelerator. We note that, as mentioned previously, preconditioning technique may also

be an appropriate strategy for improving convergence rate in Markov chains and would be one of our research topics of this field in future.

Acknowledgments

This research is supported by Chinese Universities Specialized Research Fund for the Doctoral Program (20110185110020), NSFC (61170309), Sichuan Province Sci. and Tech. Research Project (2012GZX0080), and Scientific Research Fund of Sichuan Provincial Education Department (T11008).

References

- [1] W. K. Ching, "Iterative methods for queuing systems with batch arrivals and negative customers," *BIT*, vol. 43, no. 2, pp. 285–296, 2003.
- [2] B. Meini, "Solving M/G/1 type Markov chains: recent advances and applications," *Communications in Statistics*, vol. 14, no. 1-2, pp. 479–496, 1998.
- [3] W. J. Stewart, *An Introduction to the Numerical Solution of Markov Chains*, Princeton University Press, Princeton, NJ, USA, 1994.
- [4] A. Z. Broder, R. Lempel, F. Maghoul, and J. Pedersen, "Efficient PageRank approximation via graph aggregation," *Information Retrieval*, vol. 9, no. 2, pp. 123–138, 2006.
- [5] S. D. Kamvar, T. H. Haveliwala, C. D. Manning, and G. H. Golub, "Extrapolation methods for accelerating PageRank computations," in *Proceedings of the 12th International World Wide Web Conference*, Budapest, Hungary, May 2003.
- [6] A. N. Langville and C. D. Meyer, "A survey of eigenvector methods for Web information retrieval," *SIAM Review*, vol. 47, no. 1, pp. 135–161, 2005.
- [7] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: bringing order to the web," Tech. Rep. 1999-0120, Computer Science Department, Stanford, Calif, USA, 1999.
- [8] A. Berman and R. J. Plemmons, *Nonnegative Matrices in the Mathematics Science*, SIAM, Philadelphia, Pa, USA, 1987.
- [9] H. De Sterck, T. A. Manteuffel, S. F. McCormick, Q. Nguyen, and J. Ruge, "Multilevel adaptive aggregation for Markov chains, with application to web ranking," *SIAM Journal on Scientific Computing*, vol. 30, no. 5, pp. 2235–2262, 2008.
- [10] S. Kamvar, T. Haveliwala, and G. Golub, "Adaptive methods for the computation of PageRank," *Linear Algebra and Its Applications*, vol. 386, pp. 51–65, 2004.
- [11] B. Philippe, Y. Saad, and W. J. Stewart, "Numerical methods in Markov Chain modeling," *Operations Research*, vol. 40, pp. 1156–1179, 1992.
- [12] A. Sidi, "Vector extrapolation methods with applications to solution of large systems of equations and to PageRank computations," *Computers & Mathematics with Applications*, vol. 56, no. 1, pp. 1–24, 2008.
- [13] I. Marek and P. Mayer, "Convergence analysis of an iterative aggregation/disaggregation method for computing stationary probability vectors of stochastic matrices," *Numerical Linear Algebra with Applications*, vol. 5, no. 4, pp. 253–274, 1998.
- [14] I. Marek and P. Mayer, "Convergence theory of some classes of iterative aggregation/disaggregation methods for computing stationary probability vectors of stochastic matrices," *Linear Algebra and Its Applications*, vol. 363, pp. 177–200, 2003.
- [15] P. J. Schweitzer and K. W. Kindle, "An iterative aggregation-disaggregation algorithm for solving linear equations," *Applied Mathematics and Computation*, vol. 18, no. 4, pp. 313–354, 1986.
- [16] W. J. Stewart and W. L. Cao, "Iterative aggregation/disaggregation techniques for nearly uncoupled Markov chains," *Journal of the Association for Computing Machinery*, vol. 32, no. 3, pp. 702–719, 1985.
- [17] A. H. Baker, E. R. Jessup, and T. Manteuffel, "A technique for accelerating the convergence of restarted GMRES," *SIAM Journal on Matrix Analysis and Applications*, vol. 26, no. 4, pp. 962–984, 2005.
- [18] G. H. Golub and C. Greif, "An Arnoldi-type algorithm for computing PageRank," *BIT*, vol. 46, no. 4, pp. 759–771, 2006.
- [19] R. B. Morgan, "Implicitly restarted GMRES and Arnoldi methods for nonsymmetric systems of equations," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1112–1135, 2000.
- [20] Y. Saad and M. H. Schultz, "GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems," *Society for Industrial and Applied Mathematics*, vol. 7, no. 3, pp. 856–869, 1986.
- [21] G. Wu and Y. Wei, "An Arnoldi-extrapolation algorithm for computing PageRank," *Journal of Computational and Applied Mathematics*, vol. 234, no. 11, pp. 3196–3212, 2010.
- [22] D. A. Smith, W. F. Ford, and A. Sidi, "Extrapolation methods for vector sequences," *SIAM Review*, vol. 29, no. 2, pp. 199–233, 1987.
- [23] J. Baglama, D. Calvetti, G. H. Golub, and L. Reichel, "Adaptively preconditioned GMRES algorithms," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 243–269, 1999.
- [24] M. Embree, "The tortoise and the hare restart GMRES," Tech. Rep. 01/22, Oxford University Computing Laboratory Numerical Analysis, Oxford, UK, 2001.
- [25] W. Joubert, "On the convergence behavior of the restarted GMRES algorithm for solving nonsymmetric linear systems," *Numerical Linear Algebra with Applications*, vol. 1, no. 5, pp. 427–447, 1994.
- [26] R. B. Morgan, "A restarted GMRES method augmented with eigenvectors," *SIAM Journal on Matrix Analysis and Applications*, vol. 16, no. 4, pp. 1154–1171, 1995.
- [27] Y. Saad, "Analysis of augmented Krylov subspace methods," *SIAM Journal on Matrix Analysis and Applications*, vol. 18, no. 2, pp. 435–449, 1997.
- [28] A. Chapman and Y. Saad, "Deflated and augmented Krylov subspace techniques," *Numerical Linear Algebra with Applications*, vol. 4, no. 1, pp. 43–66, 1997.
- [29] M. Benzi and B. Uçar, "Product preconditioning for Markov chain problems," in *Proceedings of the 2006 Markov Anniversary Meeting*, Charleston, SC, USA, A. N. Langville and W. J. Stewart, Eds., pp. 239–256, Bosc Books, Raleigh, NC, USA.
- [30] M. Benzi and B. Uçar, "Block triangular preconditioners for M-matrices and Markov chains," *Electronic Transactions on Numerical Analysis*, vol. 26, pp. 209–227, 2007.
- [31] J. Erhel, K. Burrage, and B. Pohl, "Restarted GMRES preconditioned by deflation," *Journal of Computational and Applied Mathematics*, vol. 69, no. 2, pp. 303–318, 1996.
- [32] S. A. Kharchenko and A. Yu. Yeremin, "Eigenvalue translation based preconditioners for the GMRES(k) method," *Numerical Linear Algebra with Applications*, vol. 2, no. 1, pp. 51–77, 1995.

- [33] K. Jbilou and H. Sadok, "Vector extrapolation methods. Applications and numerical comparison," *Journal of Computational and Applied Mathematics*, vol. 122, no. 1-2, pp. 149–165, 2000.
- [34] S. Cabay and L. W. Jackson, "A polynomial extrapolation method for finding limits and antilimits of vector sequences," *SIAM Journal on Numerical Analysis*, vol. 13, no. 5, pp. 734–752, 1976.
- [35] R. P. Eddy, "Extrapolating to the limit of a vector sequence," in *Information Linkage between Applied Mathematics and Industry*, P. C. C. Wang, Ed., pp. 387–396, Academic Press, New York, NY, USA, 1979.
- [36] M. Mešina, "Convergence acceleration for the iterative solution of the equations $X = AX + f$," *Computer Methods in Applied Mechanics and Engineering*, vol. 10, no. 2, pp. 165–173, 1977.
- [37] A. Sidi, W. F. Ford, and D. A. Smith, "Acceleration of convergence of vector sequences," *SIAM Journal on Numerical Analysis*, vol. 23, no. 1, pp. 178–196, 1986.
- [38] C. Brezinski, "Généralisations de la transformation de Shanks, de la table de Padé et de l' ϵ -algorithme," *Calcolo*, vol. 11, no. 4, pp. 317–360, 1975.
- [39] B. P. Pugachev, "Acceleration of the convergence of iterative processes and a method of solving systems of non-linear equations," *USSR Computational Mathematics and Mathematical Physics*, vol. 17, no. 5, pp. 199–207, 1978.
- [40] P. Wynn, "Acceleration techniques for iterated vector and matrix problems," *Mathematics of Computation*, vol. 16, pp. 301–322, 1962.
- [41] C. Isensee and G. Horton, "A multi-level method for the steady state solution of Markov chains," in *Simulation and Visualization*, SCS, Magdeburg, Germany, 2004.
- [42] C. Isensee and G. Horton, "A multi-level method for the steady state solution of discretetime Markov chains," in *Proceedings of the 2nd Balkan conference in informatics*, pp. 413–420, Ohrid, Macedonia, November 2005.
- [43] H. De Sterck, T. A. Manteuffel, S. F. McCormick et al., "Smoothed aggregation multigrid for Markov chains," *SIAM Journal on Scientific Computing*, vol. 32, no. 1, pp. 40–61, 2010.
- [44] H. De Sterck, K. Miller, G. Sanders, and M. Winlaw, "Recur-sively accelerated multilevel aggregation for Markov chains," *SIAM Journal on Scientific Computing*, vol. 32, no. 3, pp. 1652–1671, 2010.