TAIWANESE JOURNAL OF MATHEMATICS
Vol. 22, No. 5, pp. 1161–1171, October 2018
DOI: 10.11650/tjm/180303

A Modified Newton Method for Multilinear PageRank

Pei-Chang Guo*, Shi-Chen Gao and Xiao-Xia Guo

Abstract. When studying the multilinear PageRank problem, a system of polynomial equations needs to be solved. In this paper, we propose a modified Newton method and develop a monotone convergence theory for a third-order tensor when $\alpha < 1/2$. In this parameter regime, the sequence of vectors produced by the Newton-like method is monotonically increasing and converges to the solution. When $\alpha > 1/2$ we present an always-stochastic modified Newton iteration. Numerical results illustrate the effectiveness of this method.

1. Introduction

When receiving a search query, Google' search engine could find an immense set of web pages that contained virtually the same words as the user entered. To determine the importance of web pages, a system of scores called PageRank is devised and developed by Google [1]. The methodology can be briefly described as follows. Let α be a probability less than 1. A random web surfer, with probability α randomly transitions according to a column stochastic matrix P, which represents the link structure of the web, and with probability $1-\alpha$ randomly transitions according to the fixed distribution, a column stochastic vector v [2]. The PageRank vector x, which is the stationary distribution of the PageRank Markov chain, is unique and solves the linear system

$$x = \alpha P x + (1-\alpha)v.$$

Recently Gleich et al. extended PageRank to higher-order Markov chains and proposed multilinear PageRank [3]. The limiting probability distribution vector of a transition probability tensor discussed in [10] can be seen as a special case of multilinear PageRank. We recall that an order-m Markov chain S is a stochastic process that satisfies

$$\Pr(S_t = i_1 \mid S_{t-1} = i_2, \ldots, S_1 = i_t) = \Pr(S_t = i_1 \mid S_{t-1} = i_2, \ldots, S_{t-m} = i_{m+1}),$$

Received August 10, 2017; Accepted March 15, 2018.
Communicated by Ren-Cang Li.
2010 *Mathematics Subject Classification*. 65F30, 65H10.
Key words and phrases. multilinear PageRank, tensor, Newton-like method, monotone convergence.
The research is supported by the Fundamental Research Funds for the Central Universities in China University of Geosciences, Beijing (2652017140).
*Corresponding author.

where the future state only relies on the past m states. For a second-order n-state Markov chain S, its transition probabilities are $\underline{P}_{ijk} = \Pr(S_{t+1} = i \mid S_t = j, S_{t-1} = k)$. Through modelling a random surfer on a higher-order Markov chain, Higher-order PageRank is introduced. With probability α, the surfer transitions according to the higher-order chain, and with probability $1-\alpha$, the surfer teleports according to the distribution v.

Let $\underline{P}$ be an order-m tensor representing an $(m-1)$th order Markov chain, α be a probability less than 1, and v be a stochastic vector. Then the multilinear PageRank vector is a nonnegative, stochastic solution of the following polynomial system:

$$x = \alpha \underline{P} x^{(m-1)} + (1-\alpha)v. \tag{1.1}$$

Here $\underline{P}x^{(m-1)}$ for a vector $x \in \mathbb{R}^n$ is a vector in $\mathbb{R}^n$, whose ith component is

$$\sum_{i_2,\ldots,i_m=1}^{n} \underline{P}_{ii_2\ldots i_m} x_{i_2} \cdots x_{i_m}.$$

Gleich et al. proved that when $\alpha < 1/(m-1)$, the multilinear PageRank equation (1.1) has a unique solution. Five different methods are studied to compute the multilinear PageRank vector [3]. They are a fixed-point iteration, a shifted fixed-point iteration, a nonlinear inner-outer iteration, an inverse iteration and the Newton iteration. It's proved that the first four of them converge for an order-m tensor when $\alpha < 1/(m-1)$ and the Newton iteration converges for a third-order tensor when $\alpha < 1/2$. The fixed point iteration

$$x_{k+1} = \alpha \underline{P} x_k^{(m-1)} + (1-\alpha)v$$

converges linearly to the unique solution of (1.1) and the convergence rate is $\alpha(m-1)$. The shifted fixed-point iteration

$$x_{k+1} = \frac{\alpha}{1+\gamma}\underline{P}x_k^{(m-1)} + \frac{1-\alpha}{1+\gamma}v + \frac{\gamma}{1+\gamma}x_k$$

has a convergence rate $\frac{\alpha(m-1)+\gamma}{1+\gamma}$. Let R be the n-by-n^{m-1} flattening of $\underline{P}$ along the first index (see [4] for more on flattening of a tensor) and let $\overline{R} = \alpha R + (1-\alpha)ve^T$, where e is n^{m-1}-by-1. The inner-outer iteration is as follows

$$x_{k+1} = \frac{\alpha}{m-1}\overline{R}(x_{k+1} \otimes \cdots \otimes x_{k+1}) + \left(1 - \frac{\alpha}{m-1}\right)x_k,$$

where at each iteration step a multilinear PageRank problem with $\overline{R}$, $\alpha/(m-1)$ and x_k is solved. Although it is more expensive than (shifted) fixed-point iteration, the inner-outer iteration can converge to a solution in some cases where the (shifted) fixed-point iteration doesn't converge. The inverse iteration is

$$x_{k+1} = \alpha S(x_k)x_{k+1} + (1-\alpha)v,$$

where $S(x)$ denotes

$$S(x) = \frac{1}{m-1} R(I \otimes x_k \otimes \cdots \otimes x_k + x_k \otimes I \otimes x_k \otimes \cdots \otimes x_k + x_k \otimes \cdots \otimes x_k \otimes I)$$

and at each iteration step a PageRank problem is solved. The inverse iteration and the inner-outer iteration have similar convergence behaviors on some problems.

For the case of a third-order tensor, $\underline{P}$ is a third-order stochastic tensor. Let R be the n-by-n^2 flattening of $\underline{P}$ along the first index:

$$R = \begin{bmatrix} \underline{P}_{111} & \cdots & \underline{P}_{1n1} & \underline{P}_{112} & \cdots & \underline{P}_{1n2} & \cdots & \underline{P}_{11n} & \cdots & \underline{P}_{1nn} \\ \underline{P}_{211} & \cdots & \underline{P}_{2n1} & \underline{P}_{212} & \cdots & \underline{P}_{2n2} & \cdots & \underline{P}_{21n} & \cdots & \underline{P}_{2nn} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \underline{P}_{n11} & \cdots & \underline{P}_{nn1} & \underline{P}_{n12} & \cdots & \underline{P}_{nn2} & \cdots & \underline{P}_{n1n} & \cdots & \underline{P}_{nnn} \end{bmatrix}.$$

Here R is a matrix with column sums equal to 1. Then (1.1) is

$$\mathcal{F}(x) = x - \alpha R(x \otimes x) - (1-\alpha)v = 0. \tag{1.2}$$

The Newton iteration for (1.2) is

$$x_{k+1} = x_k - [I - \alpha R(x_k \otimes I + I \otimes x_k)]^{-1} \mathcal{F}(x_k), \quad x_0 = 0.$$

When $\alpha > 1/2$, the Newton iteration often converges to a solution that is not stochastic, so an always-stochastic Newton iteration is proposed in [3]. The Newton iteration can converge on some third-order problems where neither the inner-out iteration nor the inverse iteration converges. That is to say, among the five methods, the Newton iteration performs well on some tough problems. In [9], the authors proposed a bigger domain of α than $\alpha < 1/(m-1)$ as the uniqueness condition for the multilinear PageRank vector. Moreover, the new uniqueness condition in [9] also ensures convergence for some algorithms in [3] such as the fixed point iteration.

In this paper, we give a modified Newton method for solving the multilinear PageRank vector. We show that, for a third-order tensor when $\alpha < 1/2$, starting with a suitable initial guess, the sequence of the iterative vectors generated by the modified Newton method is monotonically increasing and converges to the unique solution of equation (1.2). When $\alpha > 1/2$ we present an always-stochastic modified Newton iteration. Numerical experiments show that the modified Newton method can be faster than the Newton method.

We introduce some necessary notation for the paper. For any matrices $B = [b_{ij}] \in \mathbb{R}^{n \times n}$, we write $B \geq 0$ ($B > 0$) if $b_{ij} \geq 0$ ($b_{ij} > 0$) holds for all i, j. For any matrices $A, B \in \mathbb{R}^{n \times n}$, we write $A \geq B$ ($A > B$) if $a_{ij} \geq b_{ij}$ ($a_{ij} > b_{ij}$) for all i, j. For any vectors $x, y \in \mathbb{R}^n$, we write $x \geq y$ ($x > y$) if $x_i \geq y_i$ ($x_i > y_i$) holds for all $i = 1, \ldots, n$. The vector of all ones is denoted by e, i.e., $e = (1, 1, \ldots, 1)^T$. The identity matrix is denoted by I.

The rest of the paper is organized as follows. In Section 2 we recall Newton's method and present a modified Newton iterative procedure. In Section 3 we prove the monotone convergence for the modified Newton method. In Section 4 we present some numerical results, which show that our new algorithm can be faster than the Newton method. In Section 5, we give our conclusions.

2. A modified Newton method

The function $\mathcal{F}$ defined in (1.2) is a mapping from $\mathbb{R}^n$ into itself and the Fréchet derivative of $\mathcal{F}$ at x is a linear map $\mathcal{F}'_x\colon \mathbb{R}^n \to \mathbb{R}^n$ given by

$$\mathcal{F}'_x\colon z \mapsto [I - \alpha R(x \otimes I + I \otimes x)]z = z - \alpha R(x \otimes z + z \otimes x).$$

To suppress the technical details, later we will consider $\mathcal{F}'_x$ and the matrix $[I - \alpha R(x \otimes I + I \otimes x)]$ as equal. The second derivative of $\mathcal{F}$ at x, $\mathcal{F}''_x\colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$, is given by

$$\mathcal{F}''_x(z_1, z_2) = -\alpha R(z_1 \otimes z_2 + z_2 \otimes z_1).$$

For a given x_0, the Newton sequence for the solution of $\mathcal{F}(x) = 0$ is

$$\begin{aligned} x_{k+1} &= x_k - (\mathcal{F}'_{x_k})^{-1}\mathcal{F}(x_k) \\ &= x_k - [I - \alpha R(x_k \otimes I + I \otimes x_k)]^{-1}\mathcal{F}(x_k) \end{aligned} \tag{2.1}$$

for $k = 0, 1, \ldots$, provided that $\mathcal{F}'_{x_k}$ is invertible for all k.

As we see, for the nonlinear equation $\mathcal{F}(x) = 0$, the sequence generated by the Newton iteration will converge quadratically to the solution [3]. However, there is a disadvantage with the Newton method. At every Newton iteration step, we need to compute the Fréchet derivative and perform an LU factorization. See more about the Newton method for other matrix equations in [5–8, 11]. In order to save the overall cost, we present a modified Newton method for (1.2) as follows.

The modified Newton method for equation (1.2)

Given initial value $x_{0,0}$, for $i = 0, 1, \ldots$

$$\begin{aligned} x_{i,s} &= x_{i,s-1} - (\mathcal{F}'_{x_{i,0}})^{-1}\mathcal{F}(x_{i,s-1}) \\ &= x_{i,s-1} - (I - \alpha R(x_{i,0} \otimes I + I \otimes x_{i,0}))^{-1}\mathcal{F}(x_{i,s-1}), \quad s = 1, 2, \ldots, n_i, \end{aligned} \tag{2.2}$$

$$x_{i+1,0} = x_{i,n_i} \tag{2.3}$$

From (2.2) we see that the method reduces to Newton's method when $n_i = 1$ for $i = 0, 1, \ldots$, and the chord method [8] when $n_0 = \infty$. The chord method needs, in total, one LU factorization so the cost for each iteration step is low. But the convergence rate of the chord method is very slow.

When $\alpha > 1/2$, the Newton iteration in [3] and the modified Newton iteration (2.2), (2.3) often converges to a solution that is not stochastic. For problems when $\alpha > 1/2$, the authors proposed a practical always-stochastic Newton iteration in [3]. At each iteration step, a projection operator is introduced, which sets negative elements of a vector to zero and normalizes the new vector to have sum one. Similarly, for problems when $\alpha > 1/2$, we present an always-stochastic modified Newton iteration as follows.

The always-stochastic modified Newton method for equation (1.2)

Given initial value $x_{0,0} = (1-\alpha)v$, for $i = 0, 1, \dots$

$$x_{i,s} = \operatorname{proj}(x_{i,s-1} - (\mathcal{F}'_{x_{i,0}})^{-1}\mathcal{F}(x_{i,s-1})), \quad s = 1, 2, \dots, n_i,$$
$$x_{i+1,0} = x_{i,n_i},$$

where $\operatorname{proj}(x) = \max(x, 0)/e^T \max(x, 0)$ is a stochastic normalization after each iteration step.

3. Convergence analysis

In this section, we prove a monotone convergence result for the modified Newton method for equation (1.2) when $\alpha < 1/2$.

3.1. Preliminaries

We first recall that a real square matrix A is called a Z-matrix if all its off-diagonal elements are nonpositive. Note that any Z-matrix A can be written as $sI - B$ with $B \geq 0$. A Z-matrix A is called an M-matrix if $s \geq \rho(B)$, where $\rho(\cdot)$ is the spectral radius; it is a singular M-matrix if $s = \rho(B)$ and a nonsingular M-matrix if $s > \rho(B)$. We will make use of the following result (see [12]).

Lemma 3.1. *For a Z-matrix A, the following are equivalent:*

(a) *A is a nonsingular M-matrix.*

(b) *$A^{-1} \geq 0$.*

(c) *$Av > 0$ for some vector $v > 0$.*

(d) *All eigenvalues of A have positive real parts.*

The next result is also well known and also can be found in [12].

Lemma 3.2. *Let A be a nonsingular M-matrix. If $B \geq A$ is a Z-matrix, then B is also nonsingular M-matrix. Moreover, $B^{-1} \leq A^{-1}$.*

3.2. Monotone convergence

The next lemma displays the monotone convergence properties of the Newton iteration (2.1).

Lemma 3.3. *Let $\alpha < 1/2$ and suppose that a vector x is such that*

(i) $\mathcal{F}(x) \le 0$,

(ii) $0 \le x$, *and* $e^T x \le 1$.

Then

$$y = x - (\mathcal{F}'_x)^{-1}\mathcal{F}(x) \tag{3.1}$$

satisfies

(a) $\mathcal{F}(y) \le 0$,

(b) $0 \le x \le y$, *and* $e^T y \le 1$.

Proof. Note that R is with all column sums equal to 1, so both $R(x \otimes I)$ and $R(I \otimes x)$ are nonnegative matrices whose column sums are $e^T x$. If $0 \le x$ and $e^T x \le 1$, then $\mathcal{F}'_x = I - \alpha R(x \otimes I + I \otimes x)$ is strictly diagonally dominant and thus a nonsingular M-matrix. So from Lemma 3.1 and condition (i), y is well-defined and $y - x \ge 0$.

From (3.1) and the Taylor formula, we have

$$\begin{aligned}\mathcal{F}(y) &= \mathcal{F}(x) + \mathcal{F}'_x(y-x) + \frac{1}{2}\mathcal{F}''_x(y-x, y-x)\\ &= \frac{1}{2}\mathcal{F}''_x(y-x, y-x)\\ &= -\alpha R[(y-x)\otimes(y-x)] \le 0.\end{aligned}$$

We now prove the second term of (b). A mathematically equivalent form of (3.1) is

$$[I - \alpha R(x \otimes I + I \otimes x)](y - x) = \alpha R(x \otimes x) + (1-\alpha)v - x. \tag{3.2}$$

Taking summations on both sides of equation (3.2), we get

$$[1 - 2\alpha(e^T x)](e^T y - e^T x) = \alpha(e^T x)^2 + (1-\alpha) - (e^T x),$$

which yields

$$e^T y = \frac{1 - \alpha - \alpha(e^T x)^2}{1 - 2\alpha(e^T x)}. \tag{3.3}$$

Combining (3.3) and $e^T x \le 1$, we know $e^T y > 1$ doesn't hold, and thus $e^T y \le 1$. □

The next lemma is an extension of Lemma 3.3, which will be the theoretical basis of our monotone convergence result of the Newton-like method for (1.2).

Lemma 3.4. *Let $\alpha < 1/2$ and suppose there is a vector x such that*

(i) $\mathcal{F}(x) \leq 0$,

(ii) $0 \leq x$, *and* $e^T x \leq 1$.

Then for any vector z with $0 \leq z \leq x$,

$$y = x - (\mathcal{F}_z')^{-1}\mathcal{F}(x)$$

satisfies

(a) $\mathcal{F}(y) \leq 0$,

(b) $0 \leq x \leq y$, *and* $e^T y \leq 1$.

Proof. Here let

$$\widehat{y} = x - (\mathcal{F}_x')^{-1}\mathcal{F}(x).$$

First, from Lemma 3.3, we know that $\mathcal{F}_x'$ is a nonsingular M-matrix. Because $0 \leq z \leq x$ and Lemma 3.2, we know that $\mathcal{F}_z'$ is also a nonsingular M-matrix and

$$0 \leq [\mathcal{F}_z']^{-1} \leq [\mathcal{F}_x']^{-1}.$$

So the vector y is well defined and $0 \leq x \leq y \leq \widehat{y}$. From Lemma 3.3, we know $e^T\widehat{y} \leq 1$, so $e^T y \leq 1$. So (b) is true. We have

$$\begin{aligned}
\mathcal{F}(y) &= \mathcal{F}(x) + \mathcal{F}_x'(y-x) + \frac{1}{2}\mathcal{F}_x''(y-x, y-x) \\
&= \mathcal{F}(x) + \mathcal{F}_z'(y-x) + (\mathcal{F}_x' - \mathcal{F}_z')(y-x) + \frac{1}{2}\mathcal{F}_x''(y-x, y-x) \\
&= \mathcal{F}_x''(x-z, y-x) + \frac{1}{2}\mathcal{F}_x''(y-x, y-x) \\
&\leq 0,
\end{aligned}$$

where the last inequality holds because $x - z \geq 0$ and $y - x \geq 0$. So (a) is true. □

Using Lemma 3.4, we can get the following monotone convergence result of the Newton-like method for (1.2). For $i = 0, 1, \ldots$, we will use x_i to denote $x_{i,0}$ in the Newton-like method (2.3), thus $x_i = x_{i,0} = x_{i-1,n_{i-1}}$.

Theorem 3.5. *Let $\alpha < 1/2$ and assume that a vector $x_{0,0}$ is such that*

(i) $\mathcal{F}(x_{0,0}) \leq 0$,

(ii) $0 \le x_{0,0}$, *and* $e^T x_{0,0} \le 1$.

Then the Newton-like method (2.2), (2.3) *generates a sequence* $\{x_k\}$ *such that* $x_k \le x_{k+1}$ *for all* $k \ge 0$, *and* $\lim_{k\to\infty} \mathcal{F}(x_k) = 0$.

Proof. We prove the theorem by mathematical induction. From Lemma 3.4, we have

$$x_{0,0} \le \cdots \le x_{0,n_0} = x_1, \quad \mathcal{F}(x_1) \le 0, \quad \text{and} \quad e^T x_1 \le 1.$$

Assume $e^T x_i \le 1$, $\mathcal{F}(x_i) \le 0$, and

$$x_{0,0} \le \cdots \le x_{0,n_0} = x_1 \le \cdots \le x_{i-1,n_{i-1}} = x_i.$$

Again by Lemma 3.4 we have

$$\mathcal{F}(x_{i+1}) \le 0, \quad x_{i,0} \le \cdots \le x_{i,n_i} = x_{i+1},$$

and $e^T x_{i+1} \le 1$. Therefore we have proved inductively the sequence $\{x_k\}$ is monotonically increasing and bounded above. So it has a limit x_*. Next we show that $\mathcal{F}(x_*) = 0$. Since $x_0 \le x_k$, from Lemma 3.2 we have

$$0 \le (\mathcal{F}'_{x_0})^{-1} \le (\mathcal{F}'_{x_k})^{-1}.$$

Letting $i \to \infty$ in $x_{i+1} \ge x_{i,1} = x_i - (\mathcal{F}'_{x_i})^{-1}\mathcal{F}(x_i) \ge x_i - (\mathcal{F}'_{x_0})^{-1}\mathcal{F}(x_i) \ge 0$, we get

$$\lim_{i\to\infty} (\mathcal{F}'_{x_0})^{-1}\mathcal{F}(x_i) = 0.$$

$\mathcal{F}(x)$ is continuous at x_*, so $(\mathcal{F}'_{x_0})^{-1}\mathcal{F}(x_*) = 0$, and thus we get $\mathcal{F}(x_*) = 0$. □

4. Numerical experiments

We remark that the modified Newton method differs from Newton's method in that the evaluation and factorization of the Fréchet derivative are not done at every iteration step. So, while more iterations will be needed than Newton's method, the overall cost of the modified Newton method can be much less. Our numerical experiments confirm the efficiency of the modified Newton method for equation (1.2).

About how to choose the optimal scalars n_i in the Newton-like method (2.2), we have no theoretical results for the moment. This is a goal for our future research. In our extensive numerical experiments, we update the Fréchet derivative every four iteration steps. That is, for $i = 0, 1, \ldots$ we choose $n_i = 4$ in the Newton-like method (2.2).

We define the number of the factorization of the Fréchet derivative in the algorithm as the outer iteration steps, which is $i + 1$ when $s > 0$ or i when $s = 0$ for an approximate solution $x_{i,s}$ in the modified Newton algorithm.

The outer iteration steps (denoted as "iter"), the elapsed CPU time in seconds (denoted as "time"), and the normalized residual (denoted as "NRes") are used to measure the effectiveness of our new method, where "NRes" is defined as

$$\text{NRes} = \frac{\|\widetilde{x} - \alpha R(\widetilde{x} \otimes \widetilde{x}) - (1-\alpha)v\|_1}{(1-\alpha)\|v\|_1 + \alpha\|R(\widetilde{x} \otimes \widetilde{x})\|_1 + \|\widetilde{x}\|_1},$$

where $\|\cdot\|_1$ denotes the vector 1-norm and $\widetilde{x}$ is an approximate solution to (1.2). We use $x = (1-\alpha)v$ as the initial iteration value of the Newton-like method. If we choose $x_{0,0} = 0$, which satisfies the assumptions of Theorem 3.5, then we have $x_{0,1} = (1-\alpha)v$ according to the modified Newton algorithm. According to Lemma 3.4, $x_{0,1} = (1-\alpha)v$ satisfies the assumptions of Theorem 3.5. So in numerical experiments we choose the vector $(1-\alpha)v$ as the initial vector. The numerical tests were performed on a laptop (2.4 Ghz and 2G Memory) with MATLAB R2013b. Numerical experiments show that the the modified Newton method can be more efficient than the Newton iteration in [3]. We present the numerical results for a random-generated problem in Table 4.1. The MATLAB code used for its generation is reported here. The problem size is $n = 300$ in Table 4.1.

```
function [R, v] = page(n)
v = ones(n, 1);
N = n * n;
rand('state',0);
R = rand(n, n * n);
s = v' * R;
for i = 1 : N
R(:, i) = R(:, i)/s(i);
end
v = v/n;
```

α	Method	Time	NRes	Iter
0.490	Newton	24.648	5.19e-13	9
	modified Newton	18.580	8.79e-13	5
0.495	Newton	28.782	1.29e-13	10
	modified Newton	19.656	3.11e-12	5
0.499	Newton	32.745	3.07e-13	12
	modified Newton	23.275	9.63e-12	6

Table 4.1: Comparison of the numerical results.

For this random-generated problem, we let $\alpha \geq 1/2$ and compare the elapsed CPU time of the always-stochastic Newton iteration in [3] and the always-stochastic modified Newton iteration in this paper. The following condition

$$\|\widetilde{x} - \alpha R(\widetilde{x} \otimes \widetilde{x}) - (1-\alpha)v\|_{\infty} < 1e - 12$$

is chosen as the stopping criterion for the two algorithms, where $\|\cdot\|_{\infty}$ denotes the infinity norm. For the random-generated problem just now, we vary α and plot the CPU time of the two algorithms in seconds for different parameters α in Figure 4.1. Here the parameter in the always-stochastic modified Newton algorithm is also chosen to be $n_i = 4$. From this figure we can see that the always-stochastic modified Newton iteration is effective and can outperform the always-stochastic Newton iteration.

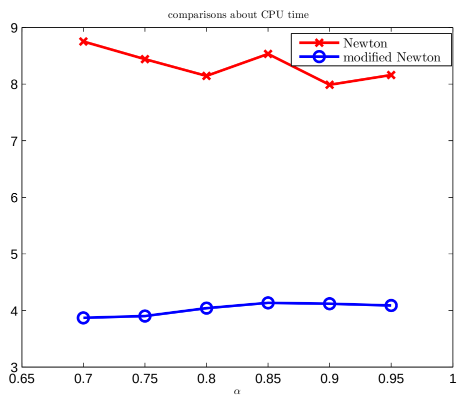

Figure 4.1: CPU time versus α.

5. Conclusions

In this paper, we consider the modified Newton method for the polynomial system of equations arising from the multilinear PageRank problem. The convergence analysis shows that this modified Newton method is feasible in the regime when $\alpha < 1/2$. We also present the always-stochastic modified Newton iteration when $\alpha > 1/2$. Numerical experiments show that the modified Newton method is effective and can outperform Newton's method. In the future, we will refer to the work of Li et al. [9] and try to improve the convergence regime of α for the modified Newton method.

References

[1] P. Fernández, *Google's Pagerank and beyond: The science of search engine rankings*, Math. Intelligencer **30** (2008), no. 1, 68–69.

[2] D. F. Gleich, *PageRank beyond the web*, SIAM Rev. **57** (2015), no. 3, 321–363.

[3] D. F. Gleich, L.-H. Lim and Y. Yu, *Multilinear PageRank*, SIAM J. Matrix Anal. Appl. **36** (2015), no. 4, 1507–1541.

[4] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Fourth edition, Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, Baltimore, MD, 2013.

[5] C.-H. Guo, *Monotone convergence of Newton-like methods for M-matrix algebraic Riccati equations*, Numer. Algorithms **64** (2013), no. 2, 295–309.

[6] P.-C. Guo and S.-F. Xu, *The modified Newton-Shamanskii method for the solution of a quadratic vector equation arising in Markovian binary trees*, Calcolo **52** (2015), no. 3, 317–325.

[7] J. He, T.-Z. Huang and L.-J. Deng, *A modified Newton method for the quadratic vector equation arising in Markovian binary trees*, Ukrainian Math. J. **68** (2016), no. 5, 812–822.

[8] C. T. Kelley, *Solving Nonlinear Equations with Newton's Method*, Fundamentals of Algorithms **1**, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2003.

[9] W. Li, D.-Liu, M. K. Ng and S.-W. Vong, *The uniqueness of multilinear PageRank vectors*, Numer. Linear Algebra Appl. **24** (2017), no. 6, e2017, 12 pp.

[10] W. Li and M. K. Ng, *On the limiting probability distribution of a transition probability tensor*, Linear Multilinear Algebra **62** (2014), no. 3, 362–385.

[11] Y. Lin and L. Bao, *Convergence analysis of the Newton-Shamanskii method for a nonsymmetric algebraic Riccati equation*, Numer. Linear Algebra Appl. **15** (2008), no. 6, 535–546.

[12] R. S. Varga, *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, N.J., 1962.

Pei-Chang Guo and Shi-Chen Gao
School of Science, China University of Geosciences, Beijing, 100083, China
E-mail address: `peichang@cugb.edu.cn`, `gsc2039@163.com`

Xiao-Xia Guo
School of Mathematical Sciences, Ocean University of China, Qingdao, 266100, China
E-mail address: `guoxiaoxia@ouc.edu.cn`