J. KSIAM Vol.11, No.4, 39-46, 2007

Solving a Matrix Polynomial by Conjugate Gradient Methods *

Hyun Ji Ko† Hyun-Min Kim‡

Abstract

One of well known and much studied nonlinear matrix equations is the matrix polynomial which has the form $G(X) = A_0X^m + A_1X^{m-1} + \cdots + A_m$, where $A_0, A_1, \cdots, A_m$ and X are $n \times n$ real matrices. We show how the minimization methods can be used to solve the matrix polynomial $G(X)$ and give some numerical experiments. We also compare Polak and Ribiére version and Fletcher and Reeves version of conjugate gradient method.

keywords. matrix polynomial, solvent, gradient, Hessian, conjugate gradient method
AMS subject classifications. 65F30, 65H10

1 Introduction

Nonlinear matrix equations often occur in applications and modelling of scientific problems. In this work we specially consider one of the nonlinear matrix equations which is called the matrix polynomial

$$G(X) = A_0X^m + A_1X^{m-1} + \cdots + A_m, \quad \text{where} \quad A_0, \cdots, A_m, X \in \mathbb{R}^{n\times n}. \tag{1.1}$$

A matrix S satisfying the equation $G(S) = 0$ is called a solvent, more precisely, a right solvent of $G(X)$ to distinguish it from a left solvent, which is a solution of the related matrix equation

$$X^mA_0 + X^{m-1}A_1 + \cdots + A_m = 0.$$

For solving the quadratic matrix equation

$$AX^2 + BX + C = 0, \quad \text{where} \quad A, B, C, X \in \mathbb{R}^{n\times n},$$

Davis [1], [2] considered Newton's method and Higham and Kim [4], [5] incorporated exact line searches into Newton's method and gave the generalized Schur decomposition approach and Bernoulli's method. Also, the conjugate gradient method was suggested by Kim [8].

In matrix polynomials Newton's method with and without line searches and Berlloulli's method were considered [6], [7], [9], [11]. Our work is to extend the conjugate gradient method for solving the matrix polynomial (1.1). Defining the object function and finding the gradient of the object function we can apply both conjugate gradient methods which are suggested by Fletcher and Reeves [3] and Polak and Rebiére [10] for minimizing the nonlinear equations.

*This work was supported for two years by Pusan National University Research Grant.
†Department of Mathematics, Pusan National University, Busan, Republic of Korea.
‡Department of Mathematics, Pusan National University, Busan, Republic of Korea. Corresponding author(hyunmin@pnu.edu).

2 Conjugate Gradient Methods for Solving a Matrix Polynomial

For solving matrix polynomials by adapting nonlinear conjugate gradient methods, first we define the gradient and Hessian of the objective function

$$f(X) = \frac{1}{2} \parallel G(X) \parallel_F^2$$

where $f : \mathbb{R}^{n\times n} \to \mathbb{R}$. To derive the gradient of f

$$\nabla f(X) = \left[\frac{\partial f}{\partial x_{11}} \cdots \frac{\partial f}{\partial x_{nn}}\right]^T \in \mathbb{R}^{n^2}$$

and the Hessian of f

$$\nabla^2 f(X) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_{11}\partial x_{11}} & \frac{\partial^2 f}{\partial x_{11}\partial x_{12}} & \cdots & \frac{\partial^2 f}{\partial x_{11}\partial x_{1n}} \\ \frac{\partial^2 f}{\partial x_{21}\partial x_{11}} & \frac{\partial^2 f}{\partial x_{21}\partial x_{22}} & \cdots & \frac{\partial^2 f}{\partial x_{21}\partial x_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_{n1}\partial x_{11}} & \frac{\partial^2 f}{\partial x_{n1}\partial x_{22}} & \cdots & \frac{\partial^2 f}{\partial x_{n1}\partial x_{nn}} \end{bmatrix} \in \mathbb{R}^{n^2\times n^2}$$

suppose that the nonlinear matrix equation $G(X)$ is twice continuously differentiable. By expanding $G(X + E)$ we obtain

$$G(X + E) = G(X) + G'_X(E) + N_X(E),$$

where $G'_X(E) : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$ is the Fréchet derivative of G at X in the direction E and $N_X(E) = O(\|E\|^2)$. Using vec operator

$$\begin{aligned} \text{vec}\{G(X+E)\} &= \text{vec}\{G(X) + G'_X(E) + N_X(E)\} \\ &= \text{vec}\{G(X)\} + \text{vec}\{G'_X(E)\} + \text{vec}\{N_X(E)\}. \end{aligned}$$

Now, the function $f(X + E)$ can be written

$$\begin{aligned} f(X+E) &= \tfrac{1}{2} \parallel G(X+E) \parallel_F^2 \\ &= \tfrac{1}{2}\Big[\text{vec}\{G(X)\} + \text{vec}\{(G'_X(E)\} + \text{vec}\{N_X(E)\}\Big]^T \\ &\qquad \Big[\text{vec}\{G(X)\} + \text{vec}\{(G'_X(E)\} + \text{vec}\{N_X(E)\}\Big] \\ &= \tfrac{1}{2}\Big[\text{vec}\{G(X)\}^T\text{vec}\{G(X)\} + 2\text{vec}\{G(X)\}^T\text{vec}\{(G'_X(E)\} + O(\|E\|_F^2)\Big], \end{aligned} \tag{2.1}$$

where

$$\begin{aligned} O(\|E\|^2) &= \text{vec}(G(X))^T\text{vec}(N_X(E)) + \text{vec}(G'_X(E))^T\left\{\text{vec}(G'_X(E)) + \text{vec}(N_X(E))\right\} \\ &\quad + \text{vec}(N_X(E))^T\left\{\text{vec}(G(X)) + \text{vec}(G'_X(E)) + \text{vec}(N_X(E))\right\}. \end{aligned}$$

By Taylor series of $f(X + E)$ we have

$$\begin{aligned} f(X+E) &= f(X) + \text{vec}(\nabla f(X))^T\text{vec}(E) + \text{vec}(E)^T\nabla^2 f(X)\text{vec}(E) + O(\|E\|_F^3) \\ &= f(X) + \text{trace}(\nabla f(X)^T E) + \text{vec}(E)^T\nabla^2 f(X)\text{vec}(E) + O(\|E\|_F^3). \end{aligned} \tag{2.2}$$

From the equations (2.1) and (2.2),

$$\text{vec}(G(X))^T\text{vec}((G'_X(E)) = \text{trace}(\nabla f(X)^T E) \tag{2.3}$$

and writing quad(y) for the quadratic part of y in the variable E,

$$\begin{aligned}\mathrm{vec}(E)^T\nabla^2 f(X)\mathrm{vec}(E) &= \frac{1}{2}\left[\mathrm{vec}(G_X'(E))^T\mathrm{vec}(G_X'(E) + 2\mathrm{vec}(G(X))^T\mathrm{quad}(\mathrm{vec}(N_X'(E))\right] \\ &= \frac{1}{2}\mathrm{trace}(G_X'(E)^T G_X'(E)) + (G(X))^T\mathrm{quad}(N_X(E)).\end{aligned}$$

By the way, we can transform these representations into different forms. In the left side of the equation (2.3) we have

$$\mathrm{vec}(G(X))^T\mathrm{vec}((G_X'(E)) = \mathrm{trace}\left(G(X)^T G_X'(E)\right)$$

and from the right side of the equation (2.3) we obtain

$$\mathrm{trace}(\nabla f(X)^T E) = \mathrm{trace}(E^T\nabla f(X)).$$

So, by setting $E = e_i e_j^T$ and using trace(AB) = trace(BA), we can get the gradient of the function f which is

$$\begin{aligned}(\nabla f)_{ij} &= \mathrm{trace}(e_j e_i^T\nabla f) \\ &= \mathrm{trace}\left(G(X)^T G_X'(E)\right).\end{aligned}$$

There is no necessity for expressing more. But it is important to know the positive definiteness of the Hessian of $f(X)$ at a solution. At a solution S, $G(S) = 0$, so

$$\mathrm{vec}(E)^T\nabla^2 f(X)\mathrm{vec}(E) = \frac{1}{2}\mathrm{trace}(G_X'(E)^T G_X'(E)). \tag{2.4}$$

Hence, the Hessian of $f(X)$ is positive definite at X if and only if $\mathrm{trace}(G_X'(E)^T G_X'(E)) > 0$ for all nonzero E if and only if $G'(X)$ is nonsingular. We can now apply minimization method for solving the equation $G(X) = 0$. Again, the gradient of $f(X)$ can be written

$$\begin{aligned}(\nabla f)_{ij} &= \mathrm{trace}\left[G(X)^T G_X'(e_i e_j^T)\right] \\ &= \mathrm{trace}\left[G(X)^T\sum_{p=1}^{m}\{(\sum_{q=0}^{m-p} A_q X^{m-(q+p)})e_i e_j^T X^{p-1}\}\right] \\ &= \mathrm{trace}\left[\sum_{p=1}^{m}\{(\sum_{q=0}^{m-p} G(X)^T A_q X^{m-(q+p)})e_i e_j^T X^{p-1}\}\right] \\ &= \sum_{p=1}^{m}\sum_{q=0}^{m-p} e_j^T X^{p-1} G(X)^T A_q X^{m-(q+p)} e_i\end{aligned}$$

hence

$$\begin{aligned}\nabla f &= \sum_{p=1}^{m}\sum_{q=0}^{m-p}\left(X^{p-1}G(X)^T A_q X^{m-(q+p)}\right)^T \\ &= \sum_{p=1}^{m}\sum_{q=0}^{m-p}(A_q X^{m-(q+p)})^T G(X)(X^{p-1})^T.\end{aligned}$$

Finally, by (2.4), the Hessian of $f(X)$ can be obtained by

$$\begin{aligned}&\mathrm{vec}(E)^T\nabla^2 f(X)\mathrm{vec}(E) \\ &= \frac{1}{2}\mathrm{trace}\sum_{l=1}^{m}\sum_{k=0}^{m-l}\left[\sum_{p=1}^{m}\sum_{q=0}^{m-p}(X^{p-1})^T E(X^{m-q-p})^T A^T\right] AX^{m-l-k}EX^{l-1}.\end{aligned}$$

3 Algorithms for Solving $G(X) = 0$ by Conjugate Gradient Methods

We now develop the algorithm of the conjugate gradient for solving a matrix polynomial. Let X_0 be given and $f : \mathbb{R}^{n\times n} \to \mathbb{R}$. To construct the approximations X_i and the search directions $\mathcal{D}_i$ for $i = 1, 2, \cdots$, we use the steepest decent direction $\mathcal{D}_i = -\nabla f(X_i)$. The following algorithm attempts to minimize $f(X)$.

Algorithm 3.1. *Evaluate* $f_0 = f(X_0)$, $\nabla f_0 = \nabla f(X_0)$

$i = 0$; $\mathcal{D}_0 = -\nabla f(X_0)$

while $\nabla f_i \neq 0$

find α_i *that minimize* $\|G(X_i + \alpha_i \mathcal{D}_i)\|_F^2$

$X_{i+1} = X_i + \alpha_i \mathcal{D}_i$

$\mathcal{D}_{i+1} = -\nabla f(X_{i+1}) + \beta_i \mathcal{D}_i$

end

The two forms of β_i are suggested by Fletcher and Reeves [3] and Polak and Ribiére [8],

$$\beta_i^{FR} = \frac{\|\nabla f_{i+1}\|_F^2}{\|\nabla f_i\|_F^2},$$

$$\beta_i^{PR} = \frac{\text{trace}((\nabla f_{i+1} - \nabla f_i)^T \nabla f_{i+1})}{\|\nabla f_i\|_F^2}.$$

We now call the conjugate gradient method with β_i^{FR} the CG_{FR} method and the conjugate gradient method with β_i^{PR} the CG_{PR} method. Algorithm 3.1 can be considered with exact line searches for a step length α_i. First, we apply exact line searches for the quadratic matrix equation

$$Q(X) = AX^2 + BX + C$$

where A, B, C, X are $n \times n$ matrices [8]. From $Q(X + \alpha\mathcal{D}) = Q(X) + \alpha D_X(\mathcal{D}) + \alpha^2 A\mathcal{D}^2$ we have a quartic polynomial

$$\begin{aligned} p_{CG}(t) &= \|Q(X + \alpha\mathcal{D})\|_F^2 \\ &= a_4\alpha^4 + a_3\alpha^3 + a_2\alpha^2 + a_1\alpha^1 + a_0, \end{aligned}$$

where

$$\begin{aligned} a_4 &= \|A\mathcal{D}^2\|_F^2, \\ a_3 &= \text{trace}(D_X(\mathcal{D})^T A\mathcal{D}^2 + (A\mathcal{D}^2)^T(D_X(\mathcal{D})), \\ a_2 &= \text{trace}(Q^T A\mathcal{D}^2 + (A\mathcal{D}^2)^T Q) + \|D_X(\mathcal{D})\|_F^2, \\ a_1 &= \text{trace}(Q^T D_X(\mathcal{D}) + D_X(\mathcal{D})^T Q), \\ a_0 &= \|Q\|_F^2. \end{aligned}$$

Since $p_{CG}(t)$ is quartic and the coefficient a_4 is positive it has a minimization.

Here we will generalize the exact line searches for the quadratic matrix equation to the matrix polynomial. Let matrix polynomial $G(X)$ in (1.1) be given. For implementation of $p_{CG}(t) = \|G(X + \alpha\mathcal{D})\|_F^2$, we must find exact expansion of $G(X + \alpha\mathcal{D})$. So we introduce some notion of repeated permutation [11]. A function $\Phi_{X,Y}[n, m-n]$ is the sum of the products of all repeated permutation of which the number of matrices X is n and the number of matrices Y is $m-n$. For example,

$$\begin{aligned}\Phi_{X,Y}[2,3] = \Phi_{X,Y}[2,5-2] \\ &= XXYYY + XYXYY + XYYXY + XYYYX + YXXYY \\ &\quad + YXYXY + YXYYX + YYXYX + YYXXY + YYYXX.\end{aligned}$$

Also,

$$\Phi_{X,Y}[0,0] = I, \quad \Phi_{X,Y}[m,0] = X^m, \quad \Phi_{X,Y}[0,m] = Y^m \quad \text{for all nonnegative integer } m.$$

And we can easily verify

$$(\Phi_{X,Y}[n,m-n])^T = \Phi_{X^T,Y^T}[n,m-n].$$

By using this notation, we can describe the expansion of $G(X+\alpha\mathcal{D})$ which has the form

$$\begin{aligned}G(X+\alpha\mathcal{D}) &= A_0(X+\alpha\mathcal{D})^m + A_1(X+\alpha\mathcal{D})^{m-1} + \cdots + A_m \\ &= A_0\sum_{i=0}^{m}\alpha^i\Phi_{X,\mathcal{D}}[m-i,i] + A_1\sum_{i=0}^{m-1}\alpha^i\Phi_{X,\mathcal{D}}[m-1-i,i] + \cdots + A_m \\ &= \sum_{j=0}^{m}\sum_{i=0}^{j}\alpha^i A_{m-j}\Phi_{X,\mathcal{D}}[j-i,i] \\ &= G(X) + \alpha D_X(\mathcal{D}) + \sum_{j=2}^{m}\sum_{i=2}^{j}\alpha^i A_{m-j}\Phi_{X,\mathcal{D}}[j-i,i].\end{aligned}$$

Finally, we obtain

$$\begin{aligned}\|G(X+\alpha\mathcal{D})\|_F^2 &= \operatorname{trace}\,(G(X+\alpha\mathcal{D}))^T G(X+\alpha\mathcal{D}) \\ &= \operatorname{trace}\left(G^T(X) + \alpha D_X(\mathcal{D})^T + \sum_{j=2}^{m}\sum_{i=2}^{j}\alpha^i\Phi_{X^T,\mathcal{D}^T}[j-i,i]A_{m-j}^T\right) \\ &\qquad \left(G(X) + \alpha D_X(\mathcal{D}) + \sum_{j=2}^{m}\sum_{i=2}^{j}\alpha^i A_{m-j}\Phi_{X,\mathcal{D}}[j-i,i]\right) \\ &= \|G(X)\|_F^2 + \cdots + \alpha^{2m}\|A_0\mathcal{D}^m\|_F^2.\end{aligned}$$

Note that exact line searches always satisfy the equation.

$$\operatorname{trace}((\nabla f_{i+1})^T\mathcal{D}_i) = \operatorname{vec}(\nabla f_{i+1})^T\operatorname{vec}\mathcal{D}_i = 0. \tag{3.1}$$

By applying the vec operation to $\mathcal{D}_{i+1} = -\nabla f(X_{i+1}) + \beta_i\mathcal{D}_i$ and premultiplying by $\operatorname{vec}(\nabla f_{i+1})^T$ we have

$$\operatorname{vec}(\nabla f_{i+1})^T\operatorname{vec}(\mathcal{D}_{i+1}) = -\|\nabla f_{i+1}\|_F^2 + \beta_i\operatorname{vec}(\nabla f_{i+1})^T\operatorname{vec}(\mathcal{D}_i).$$

Therefore, by (3.1), we have $\operatorname{vec}(\nabla f_{i+1})^T\operatorname{vec}(\mathcal{D}_{i+1}) < 0$, which means that $\mathcal{D}_{i+1}$ is descent direction.

4 Numerical Experiment and Conclusion

In this section we show and compare some experimental results CG_{FR} method and CG_{PR} method. Our experiment were done in MATLAB 7.1, which has the unit roundoff $u = 2^{-53} \approx 1.1 \times 10^{-16}$. Iterations for two methods are terminated when the relative residual $\rho(X_i)$ satisfies

$$\rho(X_i) = \frac{||fl(G(X_i))||_F}{||A_0||_F||X_i||_F^m + \cdots + ||A_m||_F} \leq nu.$$

First, we consider two examples

$$G_1(X) = X^2 + X + \begin{bmatrix} -6 & -5 \\ 0 & -6 \end{bmatrix} = 0, \tag{4.1}$$

$$G_2(X) = X^2 + \begin{bmatrix} -1 & -1 \\ 1 & -1 \end{bmatrix} X + \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} = 0 \tag{4.2}$$

from [1], [8]. Figure 4.1 and Table 4.1 show that we have exactly same results in [8] and our default matrix is, as in [1],

$$\text{Default\_}X_0 = \left(\frac{||B||_F + \sqrt{||B||_F^2 + 4||A||_F||C||_F}}{2||A||_F} \right) I.$$

Figure 4.1: Convergence for problem $G_1(X)$ in (4.1) with $\mathrm{CG_{FR}}$ and $\mathrm{CG_{PR}}$ methods.

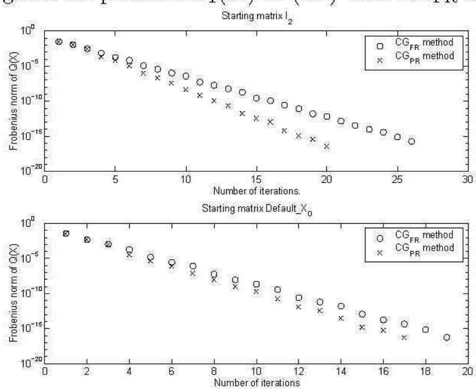

Table 4.1: Number of iterations for convergence for problem $G_2(X)$ in (4.2) with conjugate gradient methods.

X_0	$\mathrm{CG_{FR}}$	$\mathrm{CG_{PR}}$
DdfaultX_0	17	7
$10I$	83	8
$10^5 I$	34	8
$10^{10} I$	39	10

We now consider cubic matrix equations. Two examples of matrix polynomial with degree 3 are

$$G_3(X) = X^3 + X^2 + X + \begin{bmatrix} -6 & -5 \\ 0 & -6 \end{bmatrix} = 0 \tag{4.3}$$

and

$$G_4(X) = X^3 + \begin{bmatrix} 0 & -1 \\ -1 & 1 \end{bmatrix} X^2 + X + \begin{bmatrix} -10 & -7 \\ 4 & 0 \end{bmatrix} = 0. \tag{4.4}$$

Starting matrices are both I_3 and we can see the convergence results in Figures 4.2 and 4.3. Note that using CG_{FR} method we could not find the solvent of the equation (4.4).

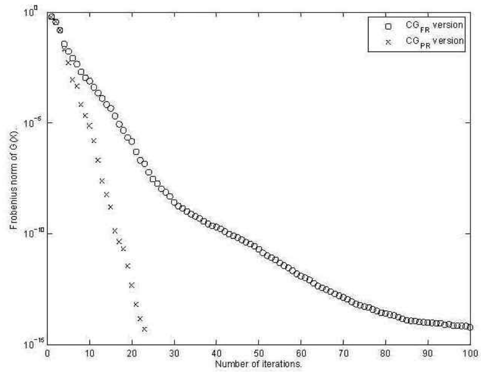

Figure 4.2: Convergence for problem (4.3) with conjugate gradient methods.

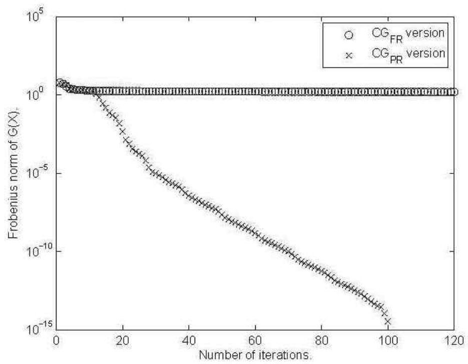

Figure 4.3: Convergence for problem (4.4) with conjugate gradient methods.

Figure 4.1, 4.2 and 4.3 show that the CG_{PR} method gives better results than the CG_{FR} method.

Finally, we give a summary of our results and compare experimental results. For solving matrix polynomials Newton's method and functional iterations were introduced. Also the conjugate gradient method for solving the quadratic matrix equations was considered. So we generalized the conjugate gradient method to the matrix polynomial. We also experimented with some examples and compare the CG_{FR} method and CG_{PR} method. Although not all of them converge, we find some examples to converge. The examples in this paper, CG_{PR} method is more efficient for converging.

References

[1] George J. Davis. Numerical solution of a quadratic matrix equation. *SIAM J. Sci. Stat. Comput.*, 2(2):164–175, 1981.

[2] George J. Davis. Algorithm 598: An algorithm to compute solvents of the matrix equation $AX^2 + BX + C = 0$. *ACM Trans. Math. Software*, 9(2):246–254, 1983.

[3] R. Fletcher and C. M. Reeves. Function minimization by conjugate gradients. *Computer J.*, 7:149–154, 1964.

[4] Nicholas J. Higham and Hyun-Min Kim. Numerical analysis of a quadratic matrix equation. *IMA J. Numer. Anal.*, 20(4):499–519, 2000.

[5] Nicholas J. Higham and Hyun-Min Kim. Solving a quadratic matrix equations by Newton's method with exact line searches. *SIAM J. Matrix Anal. Appl.*, 23(2):303–316, 2001.

[6] J. E. Dennis. Jr., J. F. Traub, and R. P. Weber. The algebraic theory of matrix polynomials. *Numerical Algorithms*, 13(6):831–845, 1976.

[7] J. E. Dennis. Jr., J. F. Traub, and R. P. Weber. Algorithms for solvents of matrix polynomials. 13(3):523–533, 1976.

[8] Hyun-Min Kim. Minimization method for solving a quadratic matrix equation. *Kyungpook Math. J.*, 47(2):239–251, 2007.

[9] W. Kratz and E. Stickel. Numerical solution of matrix polynomial equations by Newton's method. *IMA J. Numer. Anal.*, 7:355–369, 1987.

[10] E. Polak and G. Ribiére. Note sur la convergence des méthodes de directions conjugées. *Revue Fr. Inf. Rech. Oper.*, 16-R1:35–43, 1969.

[11] Jong-Hyeon Seo and Hyun-Min Kim. Solving matrix polynomials by newton's method. Preprint.