**Universidad Central de Venezuela**
**Facultad de Ciencias**
**Escuela de Computación**

***Lecturas en Ciencias de la Computación***
***ISSN 1316-6239***

# A secant method for nonlinear matrix problems

M. Monsalve y M. Raydan.

**RT 2007-06**

Centro de Cálculo Científico y Tecnológico de la UCV
CCCT-UCV
Caracas, Junio, 2007.

# A secant method for nonlinear matrix problems

Marlliny Monsalve [*] Marcos Raydan [†]

June 06, 2007

**Abstract**

Nonlinear matrix equations arise in different scientific topics, such as applied statistics and control theory, among others. Standard approaches to solve them include and combine some variations of Newton's method, matrix factorizations, and reduction to generalized eigenvalue problems. In this paper we explore the use of secant methods in the space of matrices, that represent a new approach with interesting features. For the special problem of computing the inverse or the pseudoinverse of a given matrix, we propose a specialized secant method for which we establish stability and q-superlinear convergence, and for which we also present some numerical results. In addition, for solving quadratic matrix equations, we discuss several issues, and present preliminary and encouraging numerical experiments.

**Keywords:** Secant method, Newton's method, nonlinear matrix problems, Schulz method.

## 1 Introduction

The aim of this paper is to present a secant method for solving the following matrix nonlinear problem:

$$\text{given } F : \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n} \text{ find } X_* \in \mathbb{C}^{n\times n} \text{ such that } F(X_*) = 0, \tag{1}$$

where $F$ is a Fréchet differentiable map. In what follows, we denote by $F'$ the Fréchet derivative of $F$.

This problem appears in different applications. For instances, given the matrices $A$, $B$, and $C$, the quadratic matrix equation $AX^2 + BX + C = 0$ arises in control theory [1, 2, 6, 10]. Another application is to compute the inverse or the pseudoinverse of any given matrix $A$. Indeed, if we find the root of $F(X) = X^{-1} - A$, we obtain the inverse of $A$, and iterative schemes based on Newton's method can be applied for finding the inverse or the pseudoinverse of any matrix [14, 16]. For additional applications and further results concerning nonlinear matrix problems see [9, 12].

A useful tool for solving equation (1) is the well-known Newton's method:

[*]Departamento de Computación, Facultad de Ciencias, Universidad Central de Venezuela, Ap. 47002, Caracas 1041-A, Venezuela (mmonsalv@kuaimare.ciens.ucv.ve). Supported by the Scientific Computing Center at UCV.

[†]Departamento de Computación, Facultad de Ciencias, Universidad Central de Venezuela, Ap. 47002, Caracas 1041-A, Venezuela (mraydan@kuaimare.ciens.ucv.ve). Supported by the Scientific Computing Center at UCV.

**Algorithm 1** Newton's method

1: Given $X_0 \in \mathbb{C}^{n\times n}$
2: **for** $k = 0, 1, \cdots$ **do**
3: Solve $F'(X_k)S_k = -F(X_k)$
4: $X_{k+1} = X_k + S_k$
5: **end for**

Note that we need $F'$ to find $S_k$ in each step of Algorithm (1) and in order to obtain $F'$ we can use the Taylor series for $F$ about $X$, $F(X + S) = F(X) + F'(X)S + R(S)$, where $R(S)$ is such that

$$\lim_{\|S\|\to 0} \frac{\|R(S)\|}{\|S\|} = 0,$$

The Taylor series allows us to identify the application of $F'(X)$ on $S$ which is required to solve the linear equation of step (3) in Algorithm 1. In many cases, solving that linear equation is computationally expensive (see, e.g., [17]).

As a general principle, whenever a Newton's method is applicable, a suitable secant method can be obtained, that hopefully have interesting features to exploit. For example, in the well-known scalar case, the secant method does not require the derivative, and only uses function evaluations. In that case ($f : \mathbb{C} \to \mathbb{C}$), the secant method can be written as follow:

$$x_{k+1} = x_k - \frac{f(x_k)}{a_k},$$

where $a_k$ satisfies that $f(x_k) = f(x_{k-1}) + a_k(x_k - x_{k-1})$ for $k \geq 0$, and $x_{-1}, x_0 \in \mathbb{C}$ are given.

Moreover, an extension for nonlinear systems of equations ($F : \mathbb{C}^n \to \mathbb{C}^n$), can be written as

$$x_{k+1} = x_k - A_k F(x_k)$$

where $A_k \in \mathbb{C}^{n\times n}$ satisfies that $F(x_k) = F(x_{k-1}) + A_k(x_k - x_{k-1})$ for $k \geq 1$, and the vector $x_0$ and the matrix $A_0$ are given. There are infinitely many options for building the matrix $A_k$ at every iteration. In particular, the Broyden's family of quasi-Newton methods avoids the knowledge of the Jacobian and has produced a significant body of research for many different problems (see, e.g., [5, 13]).

In this work we develop secant methods for nonlinear matrix problems that inherited, as much as possible, the features of the classical secant methods in previous scenarios (e.g., scalar equations, nonlinear algebraic systems of equations). The rest of this document is organized as follows. In section 2 we propose a general secant method for matrix problems which is based on the standard secant method, and we also describe some of its variations. In section 3 we propose a specialized secant method for approximating the inverse or the pseudoinverse of a matrix. The global convergence and the stability are proved for this specialized secant method. We present numerical experiments for computing the inverse of some given nonsingular matrices, and for computing the pseudoinverse of a singular matrix. In section 4 we consider the application of the general secant algorithms for solving quadratic matrix equations, and we also present some encouraging preliminary numerical results. Finally, in section 5, we present some conclusions and perspectives.

## 2 A secant equation for matrix problems

A general secant method for solving (1) should be given by the following iteration

$$X_{k+1} = X_k - A_k^{-1} F(X_k), \tag{2}$$

where $X_{-1} \in \mathbb{C}^{n\times n}$ and $X_0 \in \mathbb{C}^{n\times n}$ are given, and $A_{k+1}$ is a suitable linear operator that satisfies

$$A_{k+1} S_k = Y_k, \tag{3}$$

where $S_k = X_{k+1} - X_k$ and $Y_k = F(X_{k+1}) - F(X_k)$. Equation (3) is known as the *secant equation.*

Once $X_{k+1}$ has been obtained, we observe in (3) that $A_{k+1}$ can be computed at each iteration by solving a linear system of $n^2$ equations. Therefore, there is a resemblance with the scalar case, in which one equation is required to find one unknown. Similarly, we notice that one $n \times n$ matrix is enough to satisfy the matrix secant equation (3). Hence, we force the operator $A_k$ to be a matrix of the same dimension of the step $S_k$ and the map-difference $Y_k$, as in the scalar case. The proposed algorithm, and some important variants, can be summarized as follows:

**Algorithm 2** General secant method for matrix problems

Given $X_{-1} \in \mathbb{C}^{n\times n}$, $X_0 \in \mathbb{C}^{n\times n}$
**Set** $S_{-1} = X_0 - X_{-1}$
**Set** $Y_{-1} = F(X_0) - F(X_{-1})$
**Solve** $A_0 S_{-1} = Y_{-1}$ ▷ for $A_0$
**for** $k = 0, 1, \cdots$ until convergence **do**
  **Solve** $A_k S_k = -F(X_k)$ ▷ for $S_k$
  **Set** $X_{k+1} = X_k + S_k$
  **Set** $Y_k = F(X_{k+1}) - F(X_k)$
  **Solve** $A_{k+1} S_k = Y_k$ ▷ for $A_{k+1}$
**end for**

We can generate the sequence $B_k = A_k^{-1}$, instead of $A_k$, and obtain an inverse version that solves only one linear system of equations per iteration:

**Algorithm 3** Inverse secant method

Given $X_{-1} \in \mathbb{C}^{n\times n}$, $X_0 \in \mathbb{C}^{n\times n}$
**Set** $S_{-1} = X_0 - X_{-1}$
**Set** $Y_{-1} = F(X_0) - F(X_{-1})$
**Solve** $B_0 Y_{-1} = S_{-1}$ ▷ for $B_0$
**for** $k = 0, 1, \cdots$ until convergence **do**
  **Set** $S_k = -B_k F(X_k)$
  **Set** $X_{k+1} = X_k + S_k$
  **Set** $Y_k = F(X_{k+1}) - F(X_k)$
  **Solve** $B_{k+1} Y_k = S_k$ ▷ for $B_{k+1}$
**end for**

Solving a secant method that deals with $n \times n$ matrices is the most attractive feature of our proposal, and represents a sharp contrast with the standard extension of quasi-Newton methods for general Hilbert spaces, (see e.g. [7, 15]), that in this context would involve $n^2 \times n^2$ linear operators to approximate the derivative of $F$. Clearly, dealing with $n \times n$ matrices for solving the related linear systems significantly reduces the computational cost associated with the linear algebra of the algorithm.

In order to discuss some theoretical issues of the proposed general secant methods, let us consider the standard assumptions for problem (1): $F : \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$ is continuously differentiable in an open and convex set $D \subseteq \mathbb{C}^{n\times n}$. There exists $X_* \in \mathbb{C}^{n\times n}$ and $r > 0$, such that $N(X_*, r) \subset D$ is an open neighborhood of radius $r$ around $X_*$, $F(X_*) = 0$, and $F'(X_*)$ is nonsingular, and $F'(X) \in Lip_\gamma(N(X_*, r))$.

We begin by noticing that the operator $A_k$ does not approximate $F'(X_k)$ as in previous scenarios due to dimensional discrepancies. Indeed, $F'(X_k) \in \mathbb{C}^{n^2\times n^2}$ and $A_k \in \mathbb{C}^{n\times n}$. However, fortunately, $F'(X_k)S_k$ and $A_k S_k$ both live in $\mathbb{C}^{n\times n}$, which turns out to be the suitable approximation since, using the secant equation (3), we have that

$$A_{k+1}S_k = Y_k = F(X_{k+1}) - F(X_k) = F'(X_k)S_k + R(S_k). \qquad (4)$$

Subtracting $F'(X_*)S_k$ in both sides of (4), and taking norms we obtain

$$\|A_{k+1}S_k - F'(X_*)S_k\| \le \|F'(X_k) - F'(X_*)\|\|S_k\| + \|R(S_k)\|,$$

for any subordinate norm $\| \, . \, \|$. Using now that $F'(X) \in Lip_\gamma(N(X_*, r))$, and dividing by $\|S_k\|$ we have

$$\frac{\|A_{k+1}S_k - F'(X_*)S_k\|}{\|S_k\|} \le \gamma\|E_k\| + \frac{\|R(S_k)\|}{\|S_k\|}, \qquad (5)$$

where $E_k = X_k - X_*$ represents the error matrix.

Form this inequality we observe that, if convergence is attained, the left hand side tends to zero when $k$ goes to infinity, and so the sequence $\{A_k\}$, generated by Algorithm 2, tends to the Fréchet derivative, $F'(X_*)$, when they are both applied to the direction of the step $S_k$. Concerning local convergence, we have from Step 7 in Algorithm 2 that

$$\begin{aligned} E_{k+1} &= E_k - A_k^{-1}F(X_k) \\ &= E_k - A_k^{-1}F'(X_*)E_k - O(E_k^2), \end{aligned}$$

which implies that

$$\|E_{k+1}\| \leq \|E_k - A_k^{-1}(F'(X_*)E_k)\| + O(\|E_k\|^2). \tag{6}$$

Consequently, if $A_k$ in our secant algorithms is such that $A_k^{-1}(F'(X_*)E_k)$ approximates $E_k$ in a neighborhood of $X_*$, as expected, then $\|E_{k+1}\|$ is reduced with respect to $\|E_k\|$. Inequalities (5) and (6), somehow, explain the convergence behavior we have observed in our numerical results. In our next section, though, we will establish formally the stability and also the local and q-superlinear convergence of the proposed secant methods for the special case of computing the inverse or the pseudoinverse of a given matrix.

## 3 Special case: Inverse or pseudoinverse of a matrix

For computing the inverse of a given matrix $A$ we will consider iterative methods to find the root of

$$F(X) = X^{-1} - A, \tag{7}$$

and for the sake of clarity let us assume, for a while, that $A$ is nonsingular.

Newton's method from an initial guess $X_0$, for solving (7), also known as Schulz method [16], is given by

$$X_{k+1} = 2X_k - X_kAX_k. \tag{8}$$

It has been established that if $X_0 = \frac{A^T}{\|A\|_2^2}$, then Schulz method possesses global convergence [8, 18]. Moreover, if $A$ does not have an inverse, it converges to the pseudoinverse (also known as the generalized inverse) of $A$ [8, 9, 18].

First, let us consider the general secant method applied to (7)

$$\begin{aligned} X_{k+1} &= X_k - S_{k-1}(F(X_k) - F(X_{k-1}))^{-1}F(X_k) \\ &= X_k - (X_k - X_{k-1})(X_k^{-1} - X_{k-1}^{-1})^{-1}(X_k^{-1} - A). \end{aligned} \tag{9}$$

Let us assume that $A$ is *diagonalizable*, that is, there exists a nonsingular matrix $V$ such that

$$V^{-1}AV = \Lambda = diag(\lambda_1, \lambda_2 \cdots, \lambda_n),$$

where $\lambda_1, \lambda_2 \cdots, \lambda_n$ are the eigenvalues of $A$, and let us define $D_k = V^{-1}X_kV$. From (9) we have that

$$\begin{aligned} D_{k+1} &= D_k - (V^{-1}X_k - V^{-1}X_{k-1})VV^{-1}(X_k^{-1} - X_{k-1}^{-1})^{-1}VV^{-1}(X_k^{-1}V - AV) \\ &= D_k - (D_k - D_{k-1})(D_k^{-1} - D_{k-1}^{-1})^{-1}(D_k^{-1} - \Lambda). \end{aligned} \tag{10}$$

Note that if we choose $X_{-1}$ and $X_0$ such that $D_{-1} = V^{-1}X_{-1}V$ and $D_0 = V^{-1}X_0V$ are diagonal matrices, then all successive $D_k$ are diagonal too, and in this case $D_iD_j = D_jD_i$ for all $i, j$. Therefore (10) can be written as

$$\begin{aligned} D_{k+1} &= D_k - (D_k - D_{k-1})(D_k^{-1}D_{k-1}^{-1}D_{k-1} - D_{k-1}^{-1}D_k^{-1}D_k)^{-1}(D_k^{-1} - \Lambda) \\ &= D_k - (D_k - D_{k-1})(D_k^{-1}D_{k-1}^{-1}D_{k-1} - D_k^{-1}D_{k-1}^{-1}D_k)^{-1}(D_k^{-1} - \Lambda) \\ &= D_k - (D_k - D_{k-1})((D_{k-1}D_k)^{-1}(D_{k-1} - D_k))^{-1}(D_k^{-1} - \Lambda) \\ &= D_k + (D_k - D_{k-1})(D_k - D_{k-1})^{-1}(D_{k-1}D_k)(D_k^{-1} - \Lambda) \\ &= D_{k-1} + D_k - D_{k-1}\Lambda D_k. \end{aligned} \tag{11}$$

Motivated by (11) we now consider the specialized secant method for (7),

$$X_{k+1} = X_{k-1} + X_k - X_{k-1}AX_k, \tag{12}$$

that avoids the inverse matrix calculations per iteration associated with iteration (9). Notice the resemblance between (12) and Schulz method for solving the same problem. Therefore, in what follows (12) will be denoted as the secant-Schulz method. Our next result establishes that if $A$ is diagonalizable and the two initial guesses are chosen properly, then the secant-Schulz method converges locally and q-superlinearly to the inverse of $A$.

**Theorem 3.1** *Let $A \in \mathbb{C}^{n\times n}$ be a nonsingular diagonalizable matrix, that is, there exists a nonsingular matrix $V$ such that*

$$V^{-1}AV = \Lambda = diag(\lambda_1, \lambda_2 \cdots, \lambda_n),$$

*where $\lambda_1, \lambda_2 \cdots, \lambda_n$ are the eigenvalues of $A$. Let $X_{-1}$ and $X_0$ be such that $V^{-1}X_{-1}V$ and $V^{-1}X_0V$ are diagonal matrices. Then the secant-Schulz method converges locally and q-superlinearly to the inverse of $A$.*

**Proof.** Let us define $D_k = V^{-1}X_kV$ for all $k \geq -1$. From (12) we have that

$$D_{k+1} = D_{k-1} + D_k - D_{k-1}\Lambda D_k. \tag{13}$$

Since $D_{-1}$ and $D_0$ are diagonal matrices, then all successive $D_k$ are diagonal too, and in this case $D_iD_j = D_jD_i$ for all $i, j$. Moreover, since $D_k = diag(d_k^1, d_k^2, \cdots d_k^n)$ we see from (13) that

$$d_{k+1}^i = d_{k-1}^i + d_k^i - d_{k-1}^i d_k^i \lambda_i, \text{ for all } 1 \leq i \leq n, \tag{14}$$

where (14) represents $n$ uncoupled scalar secant iterations converging to $1/\lambda_i$, $1 \leq i \leq n$. Indeed, subtracting $1/\lambda_i$ in both sides of (14) and letting $e_k^i = d_k^i - 1/\lambda_i$ we have that

$$\begin{aligned} e_{k+1}^i &= d_k^i + d_{k-1}^i - d_{k-1}^i d_k^i \lambda_i - 1/\lambda_i \\ &= -\lambda_i(d_k^i d_{k-1}^i - d_k^i/\lambda_i - d_{k-1}^i/\lambda_i + 1/\lambda_i^2) \\ &= -\lambda_i(d_k^i - 1/\lambda_i)(d_{k-1}^i - 1/\lambda_i) \\ &= -\lambda_i e_k^i e_{k-1}^i. \end{aligned} \tag{15}$$

From (15) we conclude that each scalar secant iteration (14) converges locally and q-superlinearly to $1/\lambda_i$. Therefore, equivalently [5], there exists a sequence $\{c_k^i\}$, for each $1 \le i \le n$, such that $c_k^i > 0$ for all $k$, $\lim_{k\to\infty} c_k^i = 0$, and

$$|e_{k+1}^i| \le c_k^i |e_k^i|. \tag{16}$$

Using (16) we now obtain in the Frobenius norm

$$\|D_{k+1} - \Lambda^{-1}\|_F^2 = \sum_{i=1}^{n} (e_{k+1}^i)^2 \le \sum_{i=1}^{n} (c_k^i)^2 (e_k^i)^2 \le n\widehat{c}_k^2 \sum_{i=1}^{n} (e_k^i)^2 \le n\widehat{c}_k^2 \|D_k - \Lambda^{-1}\|_F^2, \tag{17}$$

where $\widehat{c}_k = \max_{1\le i\le n}\{c_k^i\}$.

Finally, we have that

$$\begin{aligned}
\|X_{k+1} - A^{-1}\|_F &= \|VV^{-1}(X_{k+1} - A^{-1})VV^{-1}\|_F \\
&= \|V(D_{k+1} - \Lambda^{-1})V^{-1}\|_F \\
&\le \kappa_F(V)\|D_{k+1} - \Lambda^{-1}\|_F \\
&\le \kappa_F(V)\sqrt{n}\widehat{c}_k\|D_k - \Lambda^{-1}\|_F \\
&= \kappa_F(V)\sqrt{n}\widehat{c}_k\|V^{-1}V(D_k - \Lambda^{-1})V^{-1}V\|_F \\
&\le \kappa_F(V)^2\sqrt{n}\widehat{c}_k\|X_k - A^{-1}\|_F, \qquad (18)
\end{aligned}$$

where $\kappa_F(V)$ is the Frobenius condition number of $V$. Hence, the secant-Schulz method converges locally and q-superlinearly to the inverse of $A$. □

When $A$ has no inverse, we can prove that the secant-Schulz method converges locally and q-superlinearly to the pseudoinverse of $A$, denoted by $A\dagger$. For this case, let $A \in \mathbb{C}^{m\times n}$ be a matrix of rank $r$, and let us assume that its singular value decomposition is given by

$$A = U \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix} V^*, \tag{19}$$

where $U \in \mathbb{C}^{m\times m}$, $V \in \mathbb{C}^{n\times n}$ are unitary matrices and $\Sigma = diag(\sigma_1, \sigma_2, \cdots \sigma_r)$ where $\sigma_1, \sigma_2 \cdots, \sigma_r$ are the singular values of $A$.

**Corollary 3.1** *Let $A \in \mathbb{C}^{m\times n}$ be a matrix of rank $r$, and let $X_{-1}$ and $X_0$ be such that $V^*X_{-1}U = \begin{pmatrix} D_{-1} & 0 \\ 0 & 0 \end{pmatrix}$ and $V^*X_0U = \begin{pmatrix} D_0 & 0 \\ 0 & 0 \end{pmatrix}$ where $V^*, U$ are defined in (19) and $D_{-1}, D_0 \in \mathbb{C}^{r\times r}$ are diagonal matrices. Then the secant-Schulz method converges locally and q-superlinearly to the pseudoinverse of $A$.*

**Proof.** From iteration (12) and defining $\begin{pmatrix} D_k & 0 \\ 0 & 0 \end{pmatrix} = V^*X_kU$, with $D_k \in \mathbb{C}^{r\times r}$, we have that

$$\begin{pmatrix} D_{k+1} & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} D_{k-1} + D_k - D_{k-1}\Sigma D_k & 0 \\ 0 & 0 \end{pmatrix}. \tag{20}$$

Since $X_{-1}$ and $X_0$ are such that $D_{-1}$ and $D_0$ are diagonal matrices then, using the same arguments as in the proof of Theorem 3.1, we obtain that

$$D_{k+1} = D_{k-1} + D_k - D_{k-1}\Sigma D_k,$$

represents $r$ uncoupled scalar secant iterations that converges locally and q-superlinearly to $1/\sigma_i$, $1 \le i \le r$, that is,

$$\|D_{k+1} - \Sigma^{-1}\|_F^2 \le r\widehat{c}_k^2 \|D_k - \Sigma^{-1}\|_F^2, \tag{21}$$

where $\widehat{c}_k = \max_{1\le i\le r}\{c_k^i\}$ and the sequences $\{c_k^i\}$ are such that $c_k^i > 0$ and $\lim_{k\to\infty} c_k^i = 0$ for each $0 \le i \le r$. Finally using the same arguments used for obtaining (18) we have that

$$\begin{aligned}
\|X_{k+1} - A^\dagger\|_F &= \|VV^*(X_{k+1} - A^\dagger)UU^*\|_F \\
&= \|V \begin{pmatrix} D_k - \Sigma^{-1} & 0 \\ 0 & 0 \end{pmatrix} U^*\|_F \\
&\le \sqrt{mn}\|D_{k+1} - \Sigma^{-1}\|_F \\
&\le \sqrt{mn}\sqrt{r}\widehat{c}_k\|D_k - \Sigma^{-1}\|_F \\
&= \sqrt{mn}\sqrt{r}\widehat{c}_k\|V^*V \begin{pmatrix} D_k - \Sigma^{-1} & 0 \\ 0 & 0 \end{pmatrix} U^*U\|_F \\
&\le mn\sqrt{r}\widehat{c}_k\|X_k - A^\dagger\|_F.
\end{aligned}$$

□

It is important to note that Theorem 3.1 implies the well-known Dennis-Moré condition [4, 5]

$$\lim_{k\to\infty} \frac{\|A_k S_k - F'(X_*)S_k\|}{\|S_k\|} = 0,$$

that establishes the most important property of the sequence $\{A_k\}$ generated by the secant-Schulz method.

We now discuss the stability of our specialized secant method for the inverse matrix. First, let us recall the suitable definition from [9]. The fixed point iteration $Y_{k+1} = G(Y_k)$ is stable in a neighborhood of a fixed point $Y_*$ if the Fréchet derivative $G'(Y_*)$ has bounded powers.

**Theorem 3.2** *The secant-Schulz method generates a stable iteration.*

**Proof.** The secant-Schulz method, as a fixed point iteration, can be obtained setting

$$Y_{k+1} = \begin{pmatrix} X_{k+1} \\ X_k \end{pmatrix} \quad , \quad Y_* = \begin{pmatrix} A^{-1} \\ A^{-1} \end{pmatrix}$$

and

$$G(Y_k) = G\begin{pmatrix} X_k \\ X_{k-1} \end{pmatrix} = \begin{pmatrix} X_{k-1} + X_k - X_{k-1}AX_k \\ X_k \end{pmatrix}.$$

Therefore, the map we need to study is given by

$$G\begin{pmatrix} W \\ Z \end{pmatrix} = \begin{pmatrix} Z + W - ZAW \\ W \end{pmatrix},$$

for $W$ and $Z$ in $\mathbb{C}^{n\times n}$. Now we will use Taylor series for identifying $G'$:

$$G(Y+P) = G(Y) + G'(Y)P + R(P), \tag{22}$$

where $P = (E_1, E_2)^T$, $E_1$ and $E_2$ are perturbation matrices, and $R$ is such that

$$\lim_{\|P\|\to 0} \frac{\|R(P)\|}{\|P\|} = 0.$$

We have that

$$G\begin{pmatrix} W + E_1 \\ Z + E_2 \end{pmatrix} = \begin{pmatrix} ((Z+E_2) + (W+E_1)) - (Z+E_2)A(W+E_1) \\ W + E_1 \end{pmatrix}$$

$$= \begin{pmatrix} (Z + W - ZAW) + (E_1 + E_2 - ZAE_1 - E_2AW) - E_2AE_1 \\ W + E_1 \end{pmatrix}. \tag{23}$$

Comparing equations (22) and (23) we conclude that

$$G'(Y)P = \begin{pmatrix} E_1 + E_2 - ZAE_1 - E_2AW \\ E_1 \end{pmatrix}. \tag{24}$$

When $Y = Y_* = (A^{-1}, A^{-1})^T$ from (24) we obtain that

$$G'(Y_*)P = \begin{pmatrix} 0 \\ E_1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ \mathrm{I} & 0 \end{pmatrix}\begin{pmatrix} E_1 \\ E_2 \end{pmatrix}.$$

Therefore, $G'(Y_*)$ is an idempotent matrix, and the iteration is stable. □

We now present a set of experiments to compute the inverse of a given matrix using the secant-Schulz iterative method. We choose as initial guesses $X_{-1} = \alpha A^T/\|A\|_2^2$ and $X_0 = \beta A^T/\|A\|_2^2$, with $\alpha, \beta \in (0, 1]$. When $A$ is symmetric and positive definite we can also choose $X_{-1} = \alpha I$ and $X_0 = \beta I$ with $\alpha > 0$ and $\beta > 0$. For these initial choices global convergence can be established. Indeed, from (12) we obtain that

$$\|I - AX_{k+1}\|_2 \le \|I - AX_{-1}\|_2^{\alpha_k} \|I - AX_0\|_2^{\beta_k} \tag{25}$$

where $\alpha_k > 0$ and $\beta_k > 0$. Note that the matrices $I - AX_{-1}$ and $I - AX_0$ are symmetric, and for this case (25) can be written as

$$\|I - AX_{k+1}\|_2 \le \rho(I - AX_{-1})^{\alpha_k} \rho(I - AX_0)^{\beta_k}$$

where $\rho(B)$ represents the spectral radius of the matrix $B$. Finally using similar arguments to the ones used to prove the global convergence of Schulz method [8, 18], we can prove that $\rho(I - AX_{-1}) < 1$ and $\rho(I - AX_0) < 1$.

In our implementation we stop all considered algorithms when

$$\|X_k - X_*\|/\|X_*\| \leq 0.5D - 14.$$

All experiments were run on a Pentium Centrino Duo, 2.0GHz, using Matlab 7. We report the number of required iterations (Iter) and the relative error ($\|X_k - X_*\|/\|X_*\|$) when the process is stopped. For our first experiment we consider the symmetric and positive definite matrix `poisson` from the Matlab gallery with $n = 400$. We compare the performance of the secant-Schulz method with the Newton-Schulz method described in (8). For the secant-Schulz method we choose $X_{-1} = 0.5 * I$, and $X_0 = A^T/\|A\|_2^2$, and for the Newton-Schulz we choose the same $X_0$. We report the results in Table 1, and the semilog of the relative error in Figure 1.

Table 1: Performance of secant-Schulz and Newton-Schulz for finding the inverse of $A =$ `gallery('poisson',20)` when $n = 400$, $X_{-1} = 0.5 * I$, and $X_0 = A^T/\|A\|_2^2$.

| Method | Iter | $\|X_k - X_*\|/\|X_*\|$ |
|---|---|---|
| Secant-Schulz | 18 | 1.95e-15 |
| Newton-Schulz | 22 | 1.87e-15 |

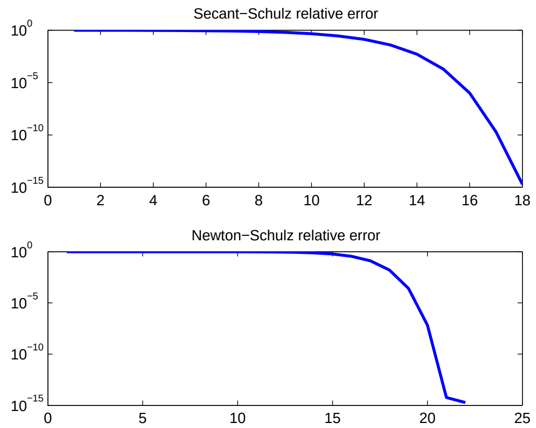


Figure 1: Semilog of the relative error for finding the inverse of $A =$ `gallery('poisson',20)` when $n = 400$, $X_{-1} = 0.5 * I$, and $X_0 = A^T/\|A\|_2^2$.

For our second experiment we consider the nonsymmetric matrix `grcar` from the Matlab gallery with $n = 200$. For the secant-Schulz method we choose $X_{-1} = 0.2 * A^T/\|A\|_2^2$, and $X_0 = A^T/\|A\|_2^2$, whereas for the Newton-Schulz we choose $X_0 = A^T/\|A\|_2^2$. We compare the performance of the secant-Schulz method with the Newton-Schulz. We report the results in Table 2.

Table 2: Performance of secant-Schulz and Newton-Schulz for finding the inverse of $A =$ `gallery('grcar',200)` with $X_{-1} = 0.2 * A^T/\|A\|_2^2$ and $X_0 = A^T/\|A\|_2^2$.

| Method | Iter | $\|X_k - X_*\|/\|X_*\|$ |
|---|---|---|
| Secant-Schulz | 14 | 2.69e-15 |
| Newton-Schulz | 10 | 4.32e-16 |

As in the Newton-Schulz method, the secant-Schulz method also converges to the pseudo-inverse of any given matrix. For our next experiment we consider the rectangular matrix `cycol` from the Matlab gallery with $n = [100 \quad 10]$ to compute its pseudoinverse, starting from the same initial choices of the previous experiment. We report the results in Table 3.

Table 3: Performance of secant-Schulz and Newton-Schulz for finding the pseudo-inverse of $A =$ `gallery('cycol',n,8)` with $X_{-1} = 0.2 * A^T/\|A\|_2^2$ and $X_0 = A^T/\|A\|_2^2$.

| Method | Iter | $\|X_k - X_*\|/\|X_*\|$ |
|---|---|---|
| Secant-Schulz | 9 | 1.85e-15 |
| Newton-Schulz | 8 | 1.86e-15 |

In all the experiments we observe the typical q-superlinear behavior of the proposed secant-Schulz method as compared with the q-quadratic behavior associated with the Newton-Schulz method.

## 4 Quadratic matrix equation

We now consider the application of Algorithms 2 and 3 for solving quadratic matrix equations of the form $AX^2 + BX + C = 0$, where $A$, $B$, and $C$ are $n \times n$ matrices. For a recent globalized implementation of Newton's method see [11]. For our secant algorithms we set $F(X) = AX^2 + BX + C$ and seek roots of $F$. For this special case, the general secant algorithm can be simplified as follows:

**Algorithm 4** Secant method for quadratic problems

Given $X_{-1} \in \mathbb{C}^{n\times n}$, $X_0 \in \mathbb{C}^{n\times n}$
**Set** $S_{-1} = X_0 - X_{-1}$
**Solve** $W_0 S_{-1} = A(X_0^2 - X_{-1}^2)$ ▷ for $W_0$
**Set** $A_0 = W_0 + B$
**for** $k = 0, 1, \cdots$ until convergence **do**
  **Solve** $A_k S_k = -F(X_k)$ ▷ for $S_k$
  **Set** $X_{k+1} = X_k + S_k$
  **Solve** $W_{k+1} S_k = A(X_{k+1}^2 - X_k^2)$ ▷ for $W_{k+1}$
  **Set** $A_{k+1} = W_{k+1} + B$
**end for**

and the inverse version of the secant algorithm can be written as follows:

**Algorithm 5** Inverse secant method for quadratic problems

Given $X_{-1} \in \mathbb{C}^{n\times n}$, $X_0 \in \mathbb{C}^{n\times n}$
**Set** $S_{-1} = X_0 - X_{-1}$
**Set** $Y_{-1} = A(X_0^2 - X_{-1}^2) + B(X_0 - X_{-1})$
**Solve** $B_0 Y_{-1} = S_{-1}$ ▷ for $B_0$
**for** $k = 0, 1, \cdots$ until convergence **do**
  **Set** $S_k = -B_k F(X_k)$
  **Set** $X_{k+1} = X_k + S_k$
  **Set** $Y_k = A(X_{k+1}^2 - X_k^2) + BS_k$
  **Solve** $B_{k+1} Y_k = S_k$ ▷ for $B_{k+1}$
**end for**

We now present some experiments to illustrate the advantages of using Algorithms 4 and 5 for solving quadratic matrix equations. For that we choose two examples already studied and described in [10] and [11]. We choose as initial guesses $X_{-1} = 0.1I$ and $X_0 = \beta I$, as in [2] for Newton's method, where $\beta = \left(\|B\|_F + \sqrt{\|B\|_F^2 + 4\|A\|_F\|C\|_F}\right)/(2\|A\|_F)$.

In our implementation we stop the algorithms when $Res(X_k) \leq n{*}eps$, where $Res(X_k) = \|F(X_k)\|_F/(\|A\|_F\|X_k\|_F^2 + \|B\|_F\|X_k\|_F + \|C\|_F)$ and $eps = 2.2D-16$. This stopping criterion is also suggested in [11]. These experiments were also run on a Pentium Centrino Duo, 2.0GHz, using Matlab 7. We report the number of required iterations (Iter) and the value of $Res(X_k)$ when the process is stopped. For our first experiment we consider the problem described by the following matrices $A = I$,

$$B = \begin{pmatrix} -1 & -1 \\ 1 & -1 \end{pmatrix} \quad , \quad C = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}. \tag{26}$$

Problem (26) has a solvent at $X_* = I$.

We compare the performance of the direct secant method (Algorithm 4) with the inverse secant method (Algorithm 5). We report the results in Table 4.

Table 4: Performance of secant and inverse secant for solving problem (26).

| $X_0$ | Method | Iter | $Res(X_k)$ |
|---|---|---|---|
| $\beta I$ | Secant | 10 | 4.15e-17 |
| $\beta I$ | Inverse secant | 11 | 2.22e-17 |
| $10I$ | Secant | 13 | 2.22e-17 |
| $10I$ | Inverse secant | 14 | 3.14e-17 |
| $10^5 I$ | Secant | 15 | 1.57e-17 |
| $10^5 I$ | Inverse secant | 16 | 5.02e-19 |
| $10^{10} I$ | Secant | 15 | 2.74e-19 |
| $10^{10} I$ | Inverse secant | 16 | 2.22e-17 |

For our second experiment we consider the problem described in [10] which is given by the following matrices $A = I$, $B = tridiag[-10, 30, -10]$ except $B(1,1) = B(n,n) = 20$, and $C = tridiag[-5, 15, -5]$, for $n = 100$. We compare the performance of the direct secant method (Algorithm 4) with the inverse secant method (Algorithm 5). We report the results in Table 5 and Figure 2. We observe in both experiments that the secant algorithms show a robust behavior converging from initial guesses either close or far away from the solution. In contrast, as reported in [11], Newton's method requires an exact line search globalization strategy to avoid the increase in number of iterations for convergence.

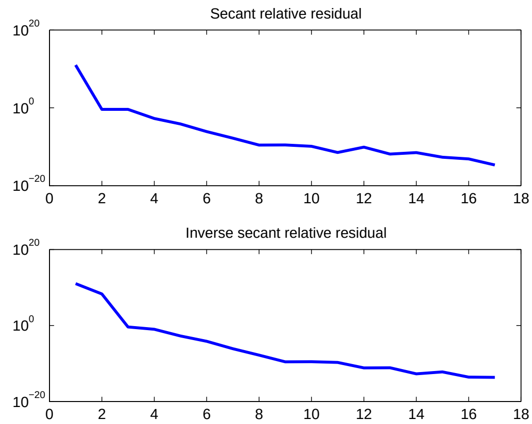


Figure 2: Semilog of the relative residual for solving our second quadratic experiment when $n = 100$ and $X_0 = 10^5 I$.

Table 5: Performance of secant and inverse secant for solving our second quadratic experiment.

| $X_0$ | Method | Iter | $Res(X_k)$ |
|---|---|---|---|
| $\beta I$ | Secant | 12 | 1.62e-14 |
| $\beta I$ | Inverse secant | 18 | 9.93e-15 |
| $10^2 I$ | Secant | 15 | 3.76e-15 |
| $10^2 I$ | Inverse secant | 18 | 1.23e-14 |
| $10^5 I$ | Secant | 17 | 1.92e-15 |
| $10^5 I$ | Inverse secant | 17 | 2.2e-14 |
| $10^{10} I$ | Secant | 18 | 1.71e-15 |
| $10^{10} I$ | Inverse secant | 16 | 7.55e-15 |
| $10^{20} I$ | Secant | 15 | 1.62e-14 |
| $10^{20} I$ | Inverse secant | 17 | 2.05e-14 |

## 5 Conclusions and perspectives

Whenever a Newton's method is applicable to a general nonlinear problem, a suitable secant method should be obtained for the same problem. In this work we present an interpretation of the classical secant method for solving nonlinear matrix problems. In the special case of computing the inverse of a given matrix, we present and fully analyze a specialized version, the secant-Schulz method, that resembles the well-known Schulz method which is a specialized version of Newton's method.

For solving quadratic matrix problems, we explore the use of the direct and also the inverse secant method. Our preliminary numerical experiments show the expected q-superlinear convergence, and indicate that these secant schemes seems to have interesting properties that remain to be established.

Finally, we hope that our specialized secant methods, for solving some simple cases, stimulate further extensions and research for solving additional and more complicated nonlinear matrix problems.

## References

[1] B. Datta [2003], *Numerical Methods for Linear Control Systems Design and Analysis*, Elsevier Academic Press.

[2] G. J. Davis [1981], Numerical solution of a quadratic matrix equation, *SIAM J. Sci. Stat. Comput.* 2, pp. 164–175.

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

[4] J.E. Dennis and J.J. Moré [1974], A characterization of superlinear convergence and its application to quasi-Newton methods, *Math. Comp.* 28, pp. 549–560.

[5] J.E. Dennis and R. Schnabel [1996], *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Classics in Applied Mathematics. SIAM Publications, Philadelphia.

[6] Y.-H. Gao [2006], Newton's method for the quadratic matrix equation, *Applied Mathematics and Computation* 2, pp.1772–1779.

[7] M. A. Gomes-Ruggiero and J. M. Martínez [1992], The column-updating method for solving nonlinear equations in Hilbert space, *RAIRO Mathematical Modelling and Numerical Analysis* 26, pp. 309–330.

[8] B. Héron, F. Issard-Roch and C. Picard [1999], *Analyse numérique*, Dunod.

[9] N. J. Higham [2006], *Functions of Matrices: Theory and Computation*, Book in preparation.

[10] N. J. Higham and H.-M. Kim [2000], Numerical analysis of a quadratic matrix equation, *IMA J. Numer. Anal.* 20, pp. 499–519.

[11] N. J. Higham and H.-M. Kim [2001], Solving a quadratic matrix equation by Newton's method with exact line searches, *SIAM J. Matrix Anal.* 23, pp. 303–316.

[12] R. A. Horn and C. R. Johnson [1991], *Topics in Matrix Analysis*, Cambridge University Press.

[13] J. M. Martínez [1990], A family of quasi-Newton methods with direct secant updates of matrix factorizations, *SIAM Journal on Numerical Analysis* 27, pp. 1034–1049.

[14] V. Pan and R. Schreiber [1991], An improved Newton iteration for the generalized inverse of a matrix, with applications, *SIAM J. Sci. Stat. Comput.* 12, pp. 1109–1130.

[15] E. Sachs [1986], Broyden's method in Hilbert space, *Math. Programming* 35, pp. 71–81.

[16] G. Schulz [1933], Iterative berechnung del reziproken matrix, *Z. Angew. Math. Mech.* 13, pp. 57–59.

[17] M. I. Smith [2003], A Schur algorithm for computing matrix $p$th roots, *SIAM J. Matrix Anal. Appl.* 24, pp. 971–989.

[18] T. Söderström and G. W. Stewart [1974], On the numerical properties of an iterative method for computing the Moore-Penrose generalized inverse, *SIAM J. Numer. Anal.* 11, pp. 61–74.