Functions of a Matrix: Theory and Computation

Nick Higham
School of Mathematics
The University of Manchester

higham@ma.man.ac.uk
http://www.ma.man.ac.uk/~higham/

Landscapes in Mathematical Science,
University of Bath, November 24 2006
Outline

1. Definition of $f(A)$
2. Motivation and MATLAB
3. $e^A$ and its Frechét derivative
4. $A^{1/2}$: Modified Newton Methods
Function of a Matrix

$f : \mathbb{C}^{n \times n} \rightarrow \mathbb{C}^{n \times n}$ for an underlying scalar function $f$.

These are not matrix functions:

- trace($A$), det($A$).
- The adjugate (or adjoint) matrix.
- Transfer function $f(t) = B(tI - A)^{-1}C$.
- sin $A = (\sin a_{ij})$.

These are matrix functions:

- $e^A = I + A + \frac{A^2}{2!} + \cdots$.
- $\log(I + A) = A - \frac{A^2}{2} + \frac{A^3}{3} + \cdots$, $\rho(A) < 1$.
- $A^{-1}, A^{1/2}$. 
There have been proposed in the literature since 1880 eight distinct definitions of a matric function, by Weyr, Sylvester and Buchheim, Giorgi, Cartan, Fantappiè, Cipolla, Schwerdtfeger and Richter.

Jordan Canonical Form

\[ Z^{-1} A Z = J = \text{diag}(J_1, \ldots, J_p), \quad J_k = \begin{bmatrix} \lambda_k & 1 \\ & \lambda_k & \ddots \\ & & \ddots & 1 \\ & & & \lambda_k \end{bmatrix} \]

Definition

\[ f(A) = Z f(J) Z^{-1} = Z \text{diag}(f(J_k)) Z^{-1}, \]

\[ f(J_k) = \begin{bmatrix} f(\lambda_k) & f'(\lambda_k) & \cdots & \frac{f(m_k-1)(\lambda_k)}{(m_k-1)!} \\ f(\lambda_k) & \ddots & \ddots & \vdots \\ \ddots & \ddots & f'(\lambda_k) & f(\lambda_k) \end{bmatrix} \]
The Formula for $f(J_k)$

Write $J_k = \lambda_k I + E_k \in \mathbb{C}^{m_k \times m_k}$. For $m_k = 3$ we have

$$E_k = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad E_k^2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad E_k^3 = 0.$$ 

Assume $f$ has Taylor expansion

$$f(t) = f(\lambda_k) + f'(\lambda_k)(t - \lambda_k) + \cdots + \frac{f^{(j)}(\lambda_k)(t - \lambda_k)^j}{j!} + \cdots.$$ 

Then

$$f(J_k) = f(\lambda_k) I + f'(\lambda_k)E_k + \cdots + \frac{f^{(m_k-1)}(\lambda_k)E_k^{m_k-1}}{(m_k - 1)!}.$$
Interpolation

Definition (Sylvester, 1883; Buchheim, 1886)

Distinct e’vals $\lambda_1, \ldots, \lambda_s$, $n_i = \text{max size of Jordan blocks for } \lambda_i$. Then $f(A) = r(A)$, where $r$ is unique Hermite interpolating poly of degree $< \sum_{i=1}^{s} n_i$ satisfying

$$r^{(j)}(\lambda_i) = f^{(j)}(\lambda_i), \quad j = 0: n_i - 1, \quad i = 1: s.$$
Definition (Sylvester, 1883; Buchheim, 1886)

Distinct e’vals $\lambda_1, \ldots, \lambda_s$, $n_i = \text{max size of Jordan blocks for } \lambda_i$. Then $f(A) = r(A)$, where $r$ is unique Hermite interpolating poly of degree $< \sum_{i=1}^s n_i$ satisfying

$$r^{(j)}(\lambda_i) = f^{(j)}(\lambda_i), \quad j = 0: n_i - 1, \quad i = 1: s.$$ 

Example. Let $f(t) = t^{1/2}$, $A = \begin{bmatrix} 2 & 2 \\ 1 & 3 \end{bmatrix}$, $\lambda(A) = \{1, 4\}$.

Taking +ve square roots,

$$r(t) = f(1) \frac{t - 4}{1 - 4} + f(4) \frac{t - 1}{4 - 1} = \frac{1}{3}(t + 2).$$

$$\Rightarrow \quad A^{1/2} = r(A) = \frac{1}{3}(A + 2I) = \frac{1}{3} \begin{bmatrix} 4 & 2 \\ 1 & 5 \end{bmatrix}.$$
Cauchy Integral Theorem

**Definition**

\[
f(A) = \frac{1}{2\pi i} \int_{\Gamma} f(z)(zl - A)^{-1} \, dz,
\]

where \( f \) is analytic on and inside a closed contour \( \Gamma \) that encloses \( \lambda(A) \).
Theorem

The three definitions are equivalent, modulo analyticity assumption for Cauchy.

- Interpolation: for basic properties.
- JCF: for solving matrix equations (e.g., $X^2 = A$, $e^X = A$). For evaluation (normal $A$).
- Cauchy: various uses.
The three definitions are equivalent, modulo analyticity assumption for Cauchy.

- Interpolation: for basic properties.
- JCF: for solving matrix equations (e.g., $X^2 = A$, $e^X = A$). For evaluation (normal $A$).
- Cauchy: various uses.

For computation:

- Use the definitions (with care).
- Schur decomposition for general $f$.
- Methods specific to particular $f$. 
1. Definition of $f(A)$

2. Motivation and MATLAB

3. $e^A$ and its Frechét derivative

4. $A^{1/2}$: Modified Newton Methods
Want to have techniques for evaluating interesting \( f \) at matrix args as well as scalar args.

Example:

\[
\frac{d^2 y}{dt^2} + Ay = 0, \quad y(0) = y_0, \quad y'(0) = y'_0
\]

has solution

\[
y(t) = \cos(\sqrt{A} t)y_0 + (\sqrt{A})^{-1} \sin(\sqrt{A} t)y'_0,
\]

where \( \sqrt{A} \) is any square root of \( A \).

MATLAB has \texttt{expm, logm, sqrtm, funm}. 
HELP is available

<>
help
Type HELP followed by
INTRO (To get started)
NEWS (recent revisions)
ABS ANS ATAN BASE CHAR CHOL CHOP CLEA COND CONJ COS
DET DIAG DIAR DISP EDIT EIG ELSE END EPS EXEC EXIT
EXP EYE FILE FLOP FLPS FOR FUN HESS HILB IF IMAG
INV KRON LINE LOAD LOG LONG LU MACR MAGI NORM ONES
ORTH PINV PLOT POLY PRIN PROD QR RAND RANK RCON RAT
REAL RETU RREF ROOT ROUN SAVE SCHU SHOR SEMI SIN SIZE
SQRT STOP SUM SVD TRIL TRIU USER WHAT WHIL WHO WHY
< > ( ) = . , ; \ / ’ + - * :
help fun

FUN For matrix arguments X, the functions SIN, COS, ATAN, SQRT, LOG, EXP and X**p are computed using eigenvalues D and eigenvectors V. If <V,D> = EIG(X) then f(X) = V*f(D)/V. This method may give inaccurate results if V is badly conditioned. Some idea of the accuracy can be obtained by comparing X**1 with X.
For vector arguments, the function is applied to each component.

The availability of [FUN] in early versions of MATLAB quite possibly contributed to the system's technical and commercial success.

Outline

1. Definition of $f(A)$
2. Motivation and MATLAB
3. $e^A$ and its Frechét derivative
4. $A^{1/2}$: Modified Newton Methods
Matrix Exponential

- Large literature.

- Over 500 citations on ISI Citation Index.
Scaling and Squaring Method

- $B \leftarrow A/2^s$ so $\|B\|_\infty \approx 1$
- $r_m(B) = [m/m]$ Padé approximant to $e^B$
- $X = r_m(B)^{2^s} \approx e^A$

Used by `expm` in MATLAB.

- Originates with Lawson (1967).
- Moler & Van Loan (1978): give backward error analysis covering truncation error in Padé approximants, allowing choice of $s$ and $m$. 
Padé Approximations $r_m$ to $e^x$

$r_m(x) = p_m(x)/q_m(x)$ known explicitly:

$$p_m(x) = \sum_{j=0}^{m} \frac{(2m - j)!}{(2m)!} \frac{m!}{(m - j)!} \frac{x^j}{j!}$$

and $q_m(x) = p_m(-x)$. Error satisfies

$$e^x - r_m(x) = (-1)^m \frac{(m!)^2}{(2m)!(2m + 1)!} x^{2m+1} + O(x^{2m+2}).$$
Let
\[ e^{-A} r_m(A) = I + G = e^H \]
and assume \( \| G \| < 1 \). Then
\[
\| H \| = \| \log(I + G) \| \leq \sum_{j=1}^{\infty} \| G \|^j / j = -\log(1 - \| G \|).
\]
Hence
\[ r_m(A) = e^A e^H = e^{A+H}. \]
Rewrite as
\[ r_m(A/2^s)^{2^s} = e^{A+E}, \]
where \( E = 2^s H \) satisfies
\[
\| E \| \leq -2^s \log(1 - \| G \|).
\]
Let

$$e^{-2^{-s}A} r_m(2^{-s}A) = I + G,$$

where \( \|G\| < 1 \). Then the Padé approximant \( r_m \) satisfies

$$r_m(2^{-s}A)^{2s} = e^{A+E},$$

where

$$\frac{\|E\|}{\|A\|} \leq \frac{-\log(1 - \|G\|)}{\|2^{-s}A\|}.$$

- Need to bound \( \|G\| \) given \( m \) and \( \|2^{-s}A\| \).
- Then \( \|E\|/\|A\| \) bounded in terms of \( m, s \) and \( \|A\| \).
- Now select “best” \((s, m)\) pair for given \( \|A\| \).
\[ \rho(x) := e^{-x} \rho_m(x) - 1 = \sum_{i=2m+1}^{\infty} c_i x^i \]

converges absolutely for \(|x| < \min\{ |t| : q_m(t) = 0 \} =: \nu_m\). Hence, with \(\theta := \|2^{-s}A\| < \nu_m\),

\[ \| G \| = \| \rho(2^{-s}A) \| \leq \sum_{i=2m+1}^{\infty} |c_i| \theta^i =: f(\theta). \]  

(\ast)

Thus \(\|E\|/\|A\| \leq -\log(1 - f(\theta))/\theta\).

- If only \(\|A\|\) known, (\ast) is optimal bound on \(\|G\|\).
Solve “bound = unit roundoff” by summing 150 terms of series in 250 digit arithmetic.

Work out cost of evaluating $r_m(B)$ and of the squaring phase.

Minimize cost subject to retaining numerical stability in evaluation of $r_m(B)$.

<table>
<thead>
<tr>
<th>Precision</th>
<th>$m$</th>
<th>$\theta_m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE single</td>
<td>7</td>
<td>3.9</td>
</tr>
<tr>
<td>IEEE double</td>
<td>13</td>
<td>5.4</td>
</tr>
<tr>
<td>IEEE quad</td>
<td>17</td>
<td>3.3</td>
</tr>
</tbody>
</table>
Algorithm 1 (H, 2005; MATLAB 7.2, Mathematica 5.1)

1. for $m = [3 \ 5 \ 7 \ 9]$
2. \quad if $\|A\|_1 \leq \theta_m$
3. \quad \quad $X = r_m(A)$.
4. \quad quit
5. end
6. end

7. $A \leftarrow A/2^s$ with $s \geq 0$ minimal s.t. $\|A/2^s\|_1 \leq \theta_{13} = 5.4$
8. $A_2 = A^2$, $A_4 = A_2^2$, $A_6 = A_2A_4$
9. $U = A\left[ A_6(b_{13}A_6 + b_{11}A_4 + b_9A_2) + b_7A_6 + b_5A_4 + b_3A_2 + b_1I \right]$
10. $V = A_6(b_{12}A_6 + b_{10}A_4 + b_8A_2) + b_6A_6 + b_4A_4 + b_2A_2 + b_0I$
11. Solve $(-U + V)r_{13} = U + V$ for $r_{13}$.
12. $X = r_{13}2^s$ by repeated squaring.
### Comparison with Existing Algorithms

| Method              | \( m \) | \( \max |2^{-s}A| \) |
|---------------------|---------|----------------|
| Alg 1               | 13      | 5.4            |
| Ward (1977)         | 8       | 1.0            |
| Old MATLAB \texttt{expm} | 6       | 0.5            |
| Sidje (1998)        | 6       | 0.5            |

\[ \theta_8 = 1.5 \]

\[ \theta_6 = 0.54 \]

- \( \|A\|_1 > 1 \): Alg 1 requires 1–2 fewer mat mults than Ward, 2–3 fewer than \texttt{expm}.

- \( \|A\|_1 \in (2, 2.1) \):

<table>
<thead>
<tr>
<th></th>
<th>Alg 1</th>
<th>Ward</th>
<th>\texttt{expm}</th>
<th>Sidje</th>
</tr>
</thead>
<tbody>
<tr>
<td>mults</td>
<td>5</td>
<td>7</td>
<td>8</td>
<td>10</td>
</tr>
</tbody>
</table>
The bound

\[ \| A^2 - fl(A^2) \| \leq \gamma_n \| A \|^2, \quad \gamma_n = \frac{nu}{1 - nu}. \]

shows the dangers in matrix squaring.

**Open question**: are errors in squaring phase consistent with conditioning of the problem?

Our choice of parameters uses 1–5 fewer matrix squarings than previous algorithms. Hence has potential accuracy advantages.
Numerical Experiment

- 70 test matrices, dimension 2–10.
- Evaluated 1-norm relative error $\|X - \hat{X}\|_1/\|X\|_1$.

Notation:

- `expm`: Alg 1 (MATLAB 7.2).
- `old_expm`: MATLAB 7.1.
- `funm`: MATLAB 7.
- `padm`: Sidje.

$$\text{cond}(A) = \lim_{\epsilon \to 0} \max_{\|E\|_2 \leq \epsilon \|A\|_2} \frac{\|e^{A+E} - e^A\|_2}{\epsilon \|e^A\|_2}.$$
Different S&S Codes and **funm**

For the given set of solvers and test problems, plot

\( x\text{-axis: } \alpha \)

\( y\text{-axis: } \text{probability that solver has error within factor } \alpha \text{ of smallest error over all solvers on the test set.} \)
Performance Profile

- expm (Alg 1)
- padm
- funm
- old_expm

Functions of a Matrix
Nick Higham
Fréchet derivative of $f : \mathbb{C}^{n \times n} \to \mathbb{C}^{n \times n}$ at $X \in \mathbb{C}^{n \times n}$

A linear mapping $L : \mathbb{C}^{n \times n} \to \mathbb{C}^{n \times n}$ s.t. for all $E \in \mathbb{C}^{n \times n}$

$$f(X + E) - f(X) - L(X, E) = o(\|E\|).$$

**Example** For $f(X) = X^2$ we have 

$$f(X + E) - f(X) =XE + EX + E^2,$$

so $L(X, E) = XE + EX$. 
Frechét Derivative of $e^A$

\[ L(A, E) = \int_0^1 e^{A(1-s)} E e^{As} \, ds. \]

\[ \|L(A)\| := \max \|L(A, E)\| / \|E\|. \]

Condition no. of the exponential: \( \kappa_{\exp}(A) = \frac{\|L(A)\| \|A\|}{\|e^A\|}. \)

\[ \|L(A)\| \geq \|L(A, I)\| = \|e^A\| \Rightarrow \kappa_{\exp}(A) \geq \|A\|. \]

**Theorem**

- If \( A \in \mathbb{C}^{n \times n} \) is normal then in the 2-norm, \( \kappa_{\exp}(A) = \|A\|_2. \)
- If \( A \in \mathbb{R}^{n \times n} \) is a nonnegative scalar multiple of a stochastic matrix then in the \( \infty \)-norm, \( \kappa_{\exp}(A) = \|A\|_\infty. \)
Repeated trap rule:
\[ \int_0^1 f(t) \, dt \approx \frac{1}{m} \left( \frac{1}{2} f_0 + f_1 + f_2 + \cdots + f_{m-1} + \frac{1}{2} f_m \right), \quad f_i := f(i/m) \]
gives
\[ L(A, E) \approx \frac{1}{m} \left( \frac{1}{2} e^A E + \sum_{i=1}^{m-1} e^{A(1-i/m)} E e^{A i/m} + \frac{1}{2} E e^A \right). \]
Requires \( e^{A/m}, e^{2A/m}, \ldots, e^{(m-1)A/m}, e^A \).

**Lemma**

Consider \( R_1(A, E) = \sum_{i=1}^{p} w_i e^{A(1-t_i)} E e^{A t_i} \) and denote its \( m \)-times repeated form by \( R_m(A, E) \). If

\[ Q_s = R_1(2^{-s} A, 2^{-s} E), \]
\[ Q_{i-1} = e^{2^{-i} A} Q_i + Q_i e^{2^{-i} A}, \quad i = s : -1 : 1, \]

then \( Q_0 = R_{2s}(A, E) \).
Recall \( L_{x^2}(A, E) = AE + EA \).

Applying chain rule to \( e^A = (e^{A/2})^2 \) gives

\[
L_{\text{exp}}(A, E) = L_{x^2}(e^{A/2}, L_{\text{exp}}(A/2, E/2))
\]

\[
= e^{A/2} L_{\text{exp}}(A/2, E/2) + L_{\text{exp}}(A/2, E/2) e^{A/2}.
\]

Recurrence for \( L_0 = L_{\text{exp}}(A, E) \):

\[
L_s = L_{\text{exp}}(2^{-s}A, 2^{-s}E),
\]

\[
L_{i-1} = e^{2^{-i}A} L_i + L_i e^{2^{-i}A}, \quad i = s : -1 : 1.
\]
Quadrature Algorithm

Algorithm (Kenney & Laub, 1998; Mathias, 1993)

Approximates $L = L_{\text{exp}}(A, E)$ via repeated Simpson rule; order of magnitude estimate.

1. $B = A/2^s$ with $s \geq 0$ minimal s.t. $\|A/2^s\|_1 \leq 1/2$
2. $X = e^B$
3. $\tilde{X} = e^{B/2}$
4. $Q_s = 2^{-s}(XE + 4\tilde{X}E\tilde{X} + EX)/6$
5. for $i = s: -1: 1$
6. \quad if $i < s$, $X = e^{2^{-i}A}$, end
7. \quad $Q_{i-1} = XQ_i + Q_iX$
8. end
9. $L = Q_0$
Kronecker Formula

\[ L(A, E) = \int_0^1 e^{A(1-s)} E e^{A s} \, ds. \]
Kronecker Formula

\[ L(A, E) = \int_0^1 e^{A(1-s)} E e^{As} \, ds. \]

- \( A \otimes B = (a_{ij}B) \in \mathbb{C}^{n^2 \times n^2}. \)
- \( A \oplus B = A \otimes I_n + I_m \otimes B. \)
- \( e^{A \oplus B} = e^A \otimes e^B. \)
- \( \text{vec}(AXB) = (B^T \otimes A)\text{vec}(X). \)
Kronecker Formula

\[ L(A, E) = \int_{0}^{1} e^{A(1-s)} E e^{As} \, ds. \]

- \( A \otimes B = (a_{ij}B) \in \mathbb{C}^{n^2 \times n^2}. \)
- \( A \oplus B = A \otimes I_n + I_m \otimes B. \)
- \( e^{A \oplus B} = e^A \otimes e^B. \)
- \( \text{vec}(AXB) = (B^T \otimes A)\text{vec}(X). \)

**Theorem**

\[ \text{vec}(L(A, E)) = K(A)\text{vec}(E), \text{ where} \]

\[ K(A) = \frac{1}{2}(e^{A^T} \oplus e^A) \tau\left(\frac{1}{2}[A^T \oplus (-A)]\right) \in \mathbb{C}^{n^2 \times n^2}, \]

\[ \tau(x) = \tanh(x)/x \text{ and } \frac{1}{2}\|A^T \oplus (-A)\| < \pi/2 \text{ assumed.} \]
Approximating the Frechét Derivative

\[
\begin{align*}
[A^T \oplus (-A) - \theta I] \text{vec}(E) &= [A^T \otimes I - I \otimes A - \theta I] \text{vec}(E) \\
&= \text{vec}(EA - AE - \theta I) \\
&= \text{vec}(E(A - \theta/2 \cdot I) - (A + \theta/2 \cdot I)E).
\end{align*}
\]

Obtain Padé approximant

\[
\tau(x) = \frac{\tanh(x)}{x} \approx r_m(x) = \prod_{i=1}^{m} \left( \frac{x}{\beta_i} - 1 \right)^{-1} \left( \frac{x}{\alpha_i} - 1 \right)
\]

by truncating

\[
\tau(x) = 1 + \frac{1}{1 + \frac{x^2/(1 \cdot 3)}{1 + \frac{x^2/(3 \cdot 5)}{1 + \cdots \frac{x^2/((2k - 1) \cdot (2k + 1))}{1 + \cdots}}}}.
\]
Algorithm (Fréchet derivative; Kenney & Laub, 1998)

Evaluates $L = L_{\exp}(A, E)$ using scaling and squaring and [8/8] Padé approximantant to $\tau$.

1. $B = A/2^s$ with $s \geq 0$ minimal s.t. $\|A/2^s\|_1 \leq 1$
2. $G_0 = 2^{-s}E$
3. for $i = 1:8$
4. Solve $(I + B/\beta_i)G_i + G_i(I - B/\beta_i) = (I + B/\alpha_i)G_{i-1} + G_{i-1}(I - B/\alpha_i)$.
5. end
6. $X = e^B$
7. $L_s = (G_mX + XG_m)/2$
8. for $i = s:-1:1$
9. if $i < s$, $X = e^{2^{-i}A}$, end
10. $L_{i-1} = XL_i + L_iX$
11. end
12. $L = L_0$
1. Definition of $f(A)$

2. Motivation and MATLAB

3. $e^A$ and its Fréchet derivative

4. $A^{1/2}$: Modified Newton Methods
Matrix Square Root

- $X$ is a square root of $A \in \mathbb{C}^{n \times n}$ iff $X^2 = A$.
- Number of square roots may be zero, finite or infinite.

**Definition**

For $A$ with no eigenvalues on $\mathbb{R}^- = \{ x \in \mathbb{R} : x \leq 0 \}$ the
principal square root $A^{1/2}$ is unique square root $X$ with spectrum in open right half-plane.
Newton’s Method for Square Root

Apply Newton to $F(X) = X^2 - A = 0$: $X_0$ given,

\[
\begin{align*}
\text{Solve} \quad X_k E_k + E_k X_k &= A - X_k^2 \\
X_{k+1} &= X_k + E_k
\end{align*}
\]

$k = 0, 1, 2, \ldots$

**Modified Newton iteration**: freeze Fréchet derivative at $X_0$:

\[
\begin{align*}
\text{Solve} \quad X_0 E_k + E_k X_0 &= A - X_k^2 \\
X_{k+1} &= X_k + E_k
\end{align*}
\]

$k = 0, 1, 2, \ldots$

$X_0$ diagonal $\Rightarrow$ cheap to solve for $E_k$. 
Set $X_0 = (2\alpha)^{-1}I$ in modified Newton:

Visser iteration (1937)

$$X_{k+1} = X_k + \alpha(A - X_k^2), \quad X_0 = (2\alpha)^{-1}I.$$ 

- Stationary iteration.
- Richardson iteration.
- Linear convergence.
- Choice of $\alpha$?
Visser History

\[ X_{k+1} = X_k + \alpha (A - X_k^2), \quad X_0 = (2\alpha)^{-1} I. \]

- Used with \( \alpha = 1/2 \) by Visser (1937) to show positive operator on Hilbert space has a positive square root.
- Likewise in functional analysis texts, e.g. Riesz & Sz.-Nagy (1956).
- Enables proof of existence of \( A^{1/2} \) without using spectral theorem.
- Elsner proves cgce for \( A \in \mathbb{C}^{n \times n} \) with real, positive ei’vals if \( 0 < \alpha \leq \rho(A)^{-1/2} \).
Let $X_k = \theta Y_k$, $\beta = \theta \alpha$ and $\tilde{A} = \theta^{-2} A$.

$$Y_{k+1} = Y_k + \beta (\tilde{A} - Y_k^2), \quad Y_0 = \frac{1}{2\beta} I.$$  

Set $\beta = 1/2$:

$$Y_{k+1} = Y_k + \frac{1}{2} (\tilde{A} - Y_k^2), \quad Y_0 = I.$$  

With $\tilde{A} \equiv I - C$ and $Y_k = I - P_k$:

$$P_{k+1} = \frac{1}{2} (C + P_k^2), \quad P_0 = 0.$$  

$Q_k = P_k/2$:

$$Q_{k+1} = Q_k^2 + \frac{C}{4}, \quad Q_0 = 0.$$
Visser Convergence

\[ X_{k+1} = X_k + \alpha (A - X_k^2), \quad X_0 = (2\alpha)^{-1} I. \]

**Theorem (H, 2006)**

*Let \( A \in \mathbb{C}^{n \times n} \) and \( \alpha > 0 \). If \( \Lambda(I - 4\alpha^2 A) \) lies in the cardioid

\[ \mathcal{D} = \{ 2z - z^2 : z \in \mathbb{C}, \ |z| < 1 \} \]

then \( A^{1/2} \) exists and \( X_k \rightarrow A^{1/2} \) linearly.*
Example

$A \in \mathbb{R}^{16 \times 16}$ spd with $a_{ii} = i^2$, $a_{ij} = 0.1$, $i \neq j$.
Aim for rel residual $< nu$ in IEEE DP arithmetic.

**Pulay iteration** $D = \text{diag}(A)$: $\theta = 0.191$, **9 iters**.

**Visser iteration** $\alpha = 0.058$ (hand optimized), **245 iters**.
In Conclusion

- Many applications of $f(A)$, e.g. control theory, computer graphics, theoretical physics.
- Need better understanding of conditioning of $f(A)$.
- Can we exploit structure, e.g. $A \in$ matrix automorphism group or Jordan or Lie algebra?
- Krylov methods needed for large, sparse $A$.
- How to use Cauchy integral computationally (H & Trefethen)?
D. A. Bini, N. J. Higham, and B. Meini.
Algorithms for the matrix $p$th root.

C.-H. Guo and N. J. Higham.
A Schur–Newton method for the matrix $p$th root and its inverse.

N. J. Higham.
*Functions of a Matrix: Theory and Computation.*
Book in preparation.
N. J. Higham.  
The scaling and squaring method for the matrix exponential revisited.  

C. S. Kenney and A. J. Laub.  
A Schur–Fréchet algorithm for computing the logarithm and exponential of a matrix.  

R. Mathias.  
Evaluating the Frechet derivative of the matrix exponential.  