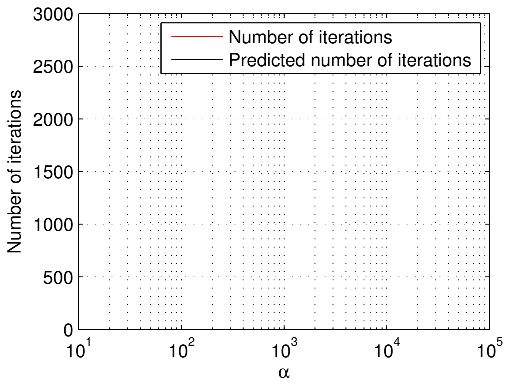
Homework 3

Deadline (for bonus points): 2014-12-12

1. **Exercise about basic QR-method.** Implement the basic QR-method. Apply it to `alpha_example.m` from the course web page. Measure the error with the maximum value below the diagonal `errfun=@(A) max(max(abs(tril(A,-1))))`.

   (a) Plot the number of iterations required to achieve error 10^{-10}, as a function of α. More precisely, generate the following plot.



For the theoretical reasoning in (b) and (c) you may use the function `eig`

   (b) Suppose the eigenvalues are ordered by magnitude $|\lambda_1| < \cdots < |\lambda_m|$. From the lecture notes we know that the elements below the diagonal will asymptotically after n iterations be proportional to $|\lambda_i/\lambda_j|^n$ with $i < j$. For large α the error will be dominated by one particular choice of i and j. Which ones?

   (c) Use (b) to establish an estimated number of iterations required to reach a specified tolerance, for different choices of α. Add a plot of the predicted number of iterations in the plot generated in (a), for tolerance 10^{-10}, and discuss the result.

Hint for (c): Show that if the error behaves as $e_k = |\beta|^k$, then $e_N = \texttt{TOL}$ if $N = \ln(\texttt{TOL})/\ln(|\beta|)$.

2. **Exercises about Hessenberg reduction.**

Hint for (a): First derive a formula first for the case $\|y\| = 1$.

   (a) Generalize Lemma 2.2.3 in the lecture notes as follows. Given a vector $x \in \mathbb{R}^n$ and a vector $y \in \mathbb{R}^n$ with $y \neq 0$ and $x \neq 0$, derive a formula for a Householder reflector (represented by a normal direction $u \in \mathbb{R}^n$) such that $Px = \alpha y$ for some value α.

(b) Implement a naive (inefficient) Hessenberg reduction by completing the program `naive_hessenberg_red.m` on the course web page.

(c) Implement Algorithm 2 in the lecture notes and compare the computation time with the algorithm in (b). Carry out the comparison by computing a Hessenberg reduction of `A=alpha_example(1,m)`, which generates an $m \times m$-matrix. Complete the following table.

	CPU-time Algorithm 2	CPU-time of algorithm in (b)
m=10		
m=100		
m=200		
m=300		
m=400		

3. **Exercise about matrix exponential.** The matrix

$$A := \begin{pmatrix} 1 & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ \alpha & & & 1 \end{pmatrix} \in \mathbb{R}^{m \times m}$$

has only simple eigenvalues when $\alpha > 0$ for $m = 10$. Implement the following methods for the matrix exponential e^A:

[GVL] are specific references to the Book Golub and Van Loan, Matrix computations, 4th edition (2013)

- The eigenvector approach [GVL Corollary 9.1.3]. You may use `[V,D]=eig(A)`.
- Taylor approximation method. Use N terms where $\|A^N\|/N! < TOL = 10^{-16}$.
- S&S = Scaling-and-squaring [GVL Algorithm 9.3.1]. Use $\delta = 10^{-16}$

For S&S you may use `find_q.m` on the course web page.

(a) Carry out simulations for the methods and for different α. Complete the folling table with the norm of the error. Use `expm(A)` as an exact solution.

	Taylor	Eigenvector approach	S&S
$\alpha = 1$			
$\alpha = 10^{-8}$			
$\alpha = 10^{-12}$			

Additional optional reading: The classical paper *Nineteen Dubious Ways to Compute the Exponential of a Matrix* `http://www.cs.cornell.edu/cv/researchpdf/19ways+.pdf` describes many other methods to compute the matrix exponential.

(b) How many matrix-matrix products are needed for Taylor and S&S?

(c) Discuss the results above. Based on the numerical experiments, which of the methods above is best in terms of robustness and efficiency?

4. **Exercise about matrix square root.** We will call the iteration [GVL (9.4.7)] Newton-SQRT and [GVL (9.4.8)] Denman-Beavers iteration.

   (a) Let A be symmetric positive definite. Show that Newton-SQRT has local quadratic convergence.

   (b) Prove that the iterates X_k of Newton-SQRT are equal to the iterates X_k in Denman-Beavers iteration.

   (c) Implement Newton-SQRT and Denman-Beavers iteration and apply them to the problem

   ```
   A=gallery('wathen',10,10);
   ```

   As a reference you may use `B=sqrtm(full(A))`. Let k be the number of iterations. Complete the following table with the norm of the error $\|X_k - B\|_2$.

k	Newton-SQRT	Denman-Beavers
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
15		
20		

   (d) What are the disadvantages / advantages of Newton-SQRT and Denman-Beavers? For the matrix in (c): Which method is best/fastest if an error of 10^{-4} is sufficient? Which method is best/fastest if an error of 10^{-13} is sufficient? "Fastest" is here meant in terms of CPU-time.

Hints for (a): We know that Newton's method has local quadratic convergence when applied to the scalar problem $f(x) = x^2 - \alpha = 0$. Show that the eigenvalues of X_k are iterates of Newton's method applied to scalar problems where α are the eigenvalues of A. If $A = V\Lambda V^{-1}$, what is $V^{-1}X_kV$?

Note that two matrix inversions are necessary for one step of Denman-Beavers.

PhD students: see next page.

Only for PhD students taking the course *Numerical linear algebra*:

5. **Exercise about exploitation of structure in specific application.** The purpose of this exercise is to learn some techniques to derive more efficient methods by taking problem-specific structure into account. (The new method you will derive is not necessarily the best for this problem-type.)

   (a) Prove that

   $$\frac{d}{dt}\exp(tA) = A\exp(tA) = \exp(tA)A$$

   (b) Let $G(t) := \exp(-tA)B\exp(tA)$ and let $[\cdot,\cdot]$ denote a commutator, i.e., $[A,B] := AB - BA$. Show that

   $$G(t) = B + t[B,A] + \frac{t^2}{2!}[[B,A],A] + \frac{t^3}{3!}[[[B,A],A],A] + \cdots \quad (*)$$

   (c) Suppose A is anti-symmetric $A^T = -A$. Let

   $$P := \int_0^\tau \exp(tA^T)B\exp(tA)\,dt$$

   Derive an expression for P involving commutators of A and B.

   (d) Let $C_k = [C_{k-1}, A]$ with $C_0 = B$. Show that $\|C_k\| \le 2^k\|A\|^k\|B\|$.

   (e) Suppose $\|A\| < \frac{1}{2}$ and $t \le 1$. Let G_N be the truncation of G,

   $$G_N(t) := B + t[B,A] + \cdots + \frac{t^N}{N!}[\cdots[[B,A],A]\cdots,A].$$

   Derive a bound for $\|G_N(t) - G(t)\|$, which converges to zero as $N \to \infty$ for any $t \le 1$.

   (f) Combine (c)-(e) and derive a numerical method to compute P where A is anti-symmetric and $\|A\| < 1/2$. Construct the algorithm such that the user can specify a tolerance.

   (g) Compare your numerical method with the naive numerical integration approach:

   ```
   P=integral(@(t) expm(t*A')*B*expm(t*A),0,T,'arrayvalued',true);
   ```

   Use $\tau = 1$ and the matrices generated by:

   ```
   A=gallery('neumann',20^2); A=A-A'; A=A/(2*norm(A,1));
   B=sprandn(length(A),length(A),0.05);
   ```

   How much better is the new method?

Connection with current research: In the field of quantum chemistry, the relation (*) for $t = 1$ is commonly called the Baker-Campbell-Hausdorff expansion. It is fundamental in one of the leading numerical methods in that field - the so-called coupled cluster approach.

The quantity P is called a Gramian, and it is often used in system and control in order to study controllability, observability and to derive optimal control as well as carrying out "model order reduction".

Not a part of the exercise: Can you derive a similar algorithm which does not require the matrix to be anti-symmetric?