Practical and fail-safe way to compute matrix exponentials

§1. Introduction and notation

When it comes to computing a matrix exponential $\exp(A)$, most textbooks tells you that you can write it down *after* you have found an invertible matrix C whose columns are generalized eigenvectors of A such that $C^{-1}\cdot A\cdot C$ is the Jordan form J of A. The exponential $\exp tJ$ of J is given by an easy formula, and then $\exp(tA) = C\cdot\exp(tJ)\cdot C^{-1}$.

But then you need to find/compute a suitable invertible matrix C which does the job. Let's assume that you have already found all eigenvalues of $A\in \mathrm{M}_n(\mathbb{C})$, i.e. you have factored the characteristic polynomial $f(x) := \det(x\cdot \mathrm{I}_n - A)$ of A:

$$f(x) = \prod_{i=1}^{r}(x-\lambda_i)^{d_i} = (x-\lambda_1)^{d_1}\cdots(x-\lambda_r)^{d_r},$$

where $\lambda_1,\dots,\lambda_r$ are mutually distinct complex numbers, and $d_1,\dots,d_r\in\mathbb{N}_{\geq 1}$ are positive integers. As we have said in class, this part is in many ways the hardest part. Roots of a polynomial with rational coefficients are often irrational. But let's say you are in luck; you have found all eigenvalues of A, expressed in simple formulas.

Reading the textbooks, you get the general impression that computing a suitable matrix C is "in principle not difficult", but they usually spell out the actual procedure. There is a reason for this "standard omission": it takes a while to explain the procedure/algorithm in a way that can be turned into computer codes. One such procedure is given in a 4-page note, in the form of a challenge problem.

Once you know the factorization of the determinant of $A\in\mathrm{M}_n(\mathbb{C})$, it is possible to write down a "formula" $E(t,x)$ with the eigenvalues of A as parameters, which is a polynomial in x of degree $n-1$, and the coefficients of various powers of x are polynomials in $e^{\lambda_1 t}, e^{\lambda_r t}$ and t. Substitute the variable x by A in this polynomial, you get $\exp(tA)$: $\exp(tA) = E(t,A)$. The general formula is given in another note, also in the form of a challenge problem.

OK, by now you have two ways to compute $\exp(tA)$. Which one is better? (I will let you decide for yourselves.) In the rest of this note I will explain a variant of the method using Jordan forms. This method is simpler than both (I think).

Basic idea. (i) Our method is based on the following observation: suppose that the characteristic polynomial of $B\in\mathrm{M}_m(\mathbb{C})$ has only one root μ, i.e. $\det(x\cdot\mathrm{I}_m - B) = (x-\mu)^m$, then $(B-\mu\cdot\mathrm{I}_m)^m = 0$ according to the Cayley–Hamilton theorem, so

$$\begin{aligned}\exp(tB) = \exp(\mu t)\cdot\exp(t(B-\mathrm{I}_m)) &= \sum_{k=0}^{m-1}\frac{t^k e^{\mu t}}{k!}(B-\mu\mathrm{I}_m)^k \\ &= \mathrm{I}_m + te^{\mu t}(B-\mu\mathrm{I}_m)+\cdots+\frac{t^{m-1}e^{\mu t}}{(m-1)!}(B-\mu\mathrm{I}_m)^{m-1}\end{aligned}$$

(ii) For each $i=1,\dots,r$, the generalized λ_i-eigenspace $V(\lambda_i) := \mathrm{Ker}((A-\lambda_i)^{d_i})$ has dimension d_i, and is closed/stable under (left multiplication by) A. Moreover the restriction to $V(\lambda_i)$ of A is a linear operator on $V(\lambda_i)$ whose characteristic polynomial is $(x-\lambda_i)^{d_i}$, therefore its exponential can be computed as in (i) above.

(1.1) The algorithm

Step 1. For each $i = 1, \ldots, r$, compute $(A - \lambda_i \mathrm{I}_n)^{d_i}$, and use Gaussian elimination to find a basis $v_{i,1}, \ldots, v_{i,d_i}$ of the d_i-dimensional vector subspace $\mathrm{Ker}((A - \lambda_i \mathrm{I}_n)^{d_i})$ of $\mathbb{C}^n_{\mathrm{col}}$.

Let C be the invertible $n \times n$ matrix whose columns are $v_{1,1}, v_{1,d_1}; \ldots, v_{r,1}, \ldots, v_{r,d_r}$.

Step 2. For each $i = 1, \ldots, r$, compute $A \cdot v_{i,k}$ for $k = 1, \ldots, d_i$, and find the d_i^2 complex numbers $b_{i,j,k}$, $1 \le j,k \le d_i$ uniquely determined by

$$A \cdot v_{i,k} = \sum_{j=1}^{d_i} b_{i,j,k} v_{i,j} = b_{i,1,k} v_{i,1} + \cdots + b_{i,d_i,k} v_{i,d_i} \qquad \text{for } j = 1, \ldots, d_i.$$

(Think of the above displayed equality as a system of n linear equations, one equation for each of the coordinates of $\mathbb{C}^n_{\mathrm{col}}$, in d_i unknowns $b_{i,1,k}, \ldots, b_{i,d_i,k}$.)

EFFECTS OF STEPS 1 AND 2 SO FAR.

Let B_i be the $d_i \times d_i$ matrix whose (j,k)-entry is $b_{i,j,k}$. Let B be the $n \times n$ matrix written in block form, with $B_1, \ldots, B_r$ sitting on the diagonal blocks; all other entries of B not in any of the diagonal blocks are 0. Then

- $A = C \cdot C^{-1}$ and $\exp(tA) = C \cdot \exp(tB) \cdot C^{-1}$.
- $\det(x \cdot \mathrm{I}_{d_i} - B_i) = (x - \lambda_i)^{d_i}$ for each $i = 1, \ldots, r$.

Step 3. Compute the matrices $(B_i - \lambda_i \mathrm{I}_{d_i})^h \in \mathrm{M}_{d_i}(\mathbb{C})$ for $i = 1, \ldots, r$, $h = 1, \ldots, d_i - 1$.

EFFECTS OF STEP 3: you have computed $\exp(tB)$.

- $\exp(tB_i) = \mathrm{I}_{d_i} + te^{\lambda_i t}(B_i - \lambda_i \mathrm{I}_{d_i}) + \cdots + \dfrac{t^{d_i - 1} e^{\lambda_i t}}{(d_i - 1)!}(B_i - \lambda_i \mathrm{I}_{d_i})^{d_i - 1}$ for $i = 1, \ldots, r$.
- $\exp(tB)$ is the $n \times n$ matrix in block form similar to B, with all off-diagonal blocks equal to 0, and the diagonal blocks are the $\exp(tB_1), \ldots, \exp(tB_r)$ given by the formula above.

Step 4. Compute C^{-1} (by Gaussian elimination) and

$$\exp(tA) = C \cdot \exp(tB) \cdot C^{-1}.$$

Remark. If all you need is to find a basis of the solutions of the first order ODE

$$\frac{d}{dt}\vec{x}(t) - A \cdot \vec{x}(t) = 0,$$

you don't need to compute C^{-1}: just compute the product $C \cdot \exp(tB)$. The n-columns of $C \cdot \exp(tB)$ form a basis of the $\mathbb{C}$-vector space of all solutions of the above linear ODE.