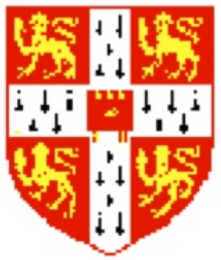


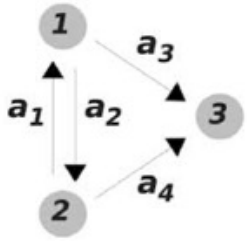
UNIVERSITY OF CAMBRIDGE

CENTRE FOR MATHEMATICAL SCIENCES

DEPARTMENT OF APPLIED MATHEMATICS
& THEORETICAL PHYSICS

A tale of two matrices: Master Equations and Lie-group methods

Arieh Iserles & Shev MacNamara



mass-action
kinetics

Laplacian matrix of G

$$\frac{d}{dt}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} -(a_2+a_3) & a_1 & 0 \\ a_2 & -(a_1+a_4) & 0 \\ a_3 & a_4 & 0 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

Oberwolfach **March 2016**

I. A tale of two matrices

Consider the linear ODE system*

$$\boldsymbol{y}' = [A^{[0]} + f(t)A^{[1]}]\boldsymbol{y}, \qquad \boldsymbol{y}(0) = \boldsymbol{y}_0 \succ \mathbf{0} \in \mathbb{R}^{N+1}, \;\; \mathbf{1}^\top \boldsymbol{y}_0 = 1,$$

where $|f(t)| \leq 1$,

$$A^{[0]} = \begin{bmatrix} -N & 1 & 0 & \cdots & \cdots & 0 \\ N & -N & 2 & 0 & \cdots & 0 \\ 0 & N-1 & -N & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 2 & -N & N \\ 0 & \cdots & \cdots & 0 & 1 & -N \end{bmatrix},$$

$$A^{[1]} = \begin{bmatrix} N & 1 & 0 & \cdots & \cdots & 0 \\ -N & N-2 & 2 & 0 & \cdots & 0 \\ 0 & -N+1 & N-4 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -2 & -N+2 & N \\ 0 & \cdots & \cdots & 0 & -1 & -N \end{bmatrix}.$$

*A model of isomerization in a monomolecular framework.

I.1 What are the eigenvalues?

Proposition *Let* $B_N = A^{[1]}$ *and set*

$$(R_N)_{k,\ell} = \begin{cases} 1, & k \geq \ell, \\ 0, & k > \ell, \end{cases} \qquad k, \ell = 0, \ldots, N.$$

Then

$$R_N B_N = \begin{bmatrix} B_{N-1} & N e_N \\ \mathbf{0}^\top & 0 \end{bmatrix} R_N,$$

where $e_N \in \mathbb{R}^N$ *is the* N*th unit vector.*

Lemma *The matrix* $A^{[1]}$ *is nilpotent.*

Proof. By induction. Set

$$\tilde{B}_N = \begin{bmatrix} B_{N-1} & N e_N \\ \mathbf{0}^\top & 0 \end{bmatrix}, \qquad \text{hence} \qquad \tilde{B}_N = R_N B_N R_N^{-1}$$

is similar to B_N.

For any $N \times N$ matrix S, $\boldsymbol{t} \in \mathbb{R}^N$ and $m \geq 0$

$$\begin{bmatrix} S & \boldsymbol{t} \\ \boldsymbol{0}^\top & 0 \end{bmatrix}^m = \begin{bmatrix} S^m & S^{m-1}\boldsymbol{t} \\ \boldsymbol{0}^\top & 0 \end{bmatrix}.$$

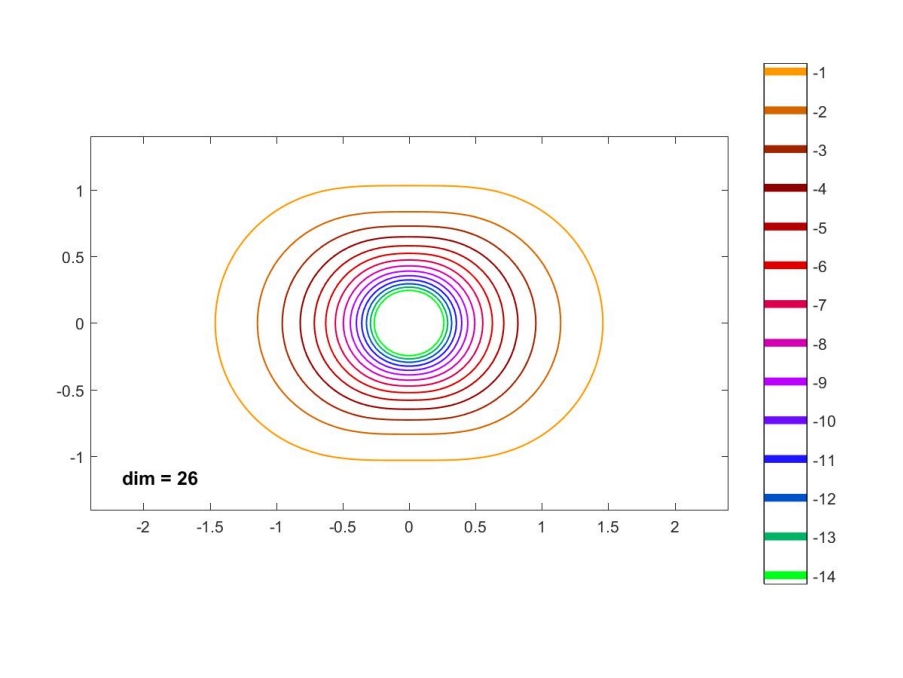
In our case

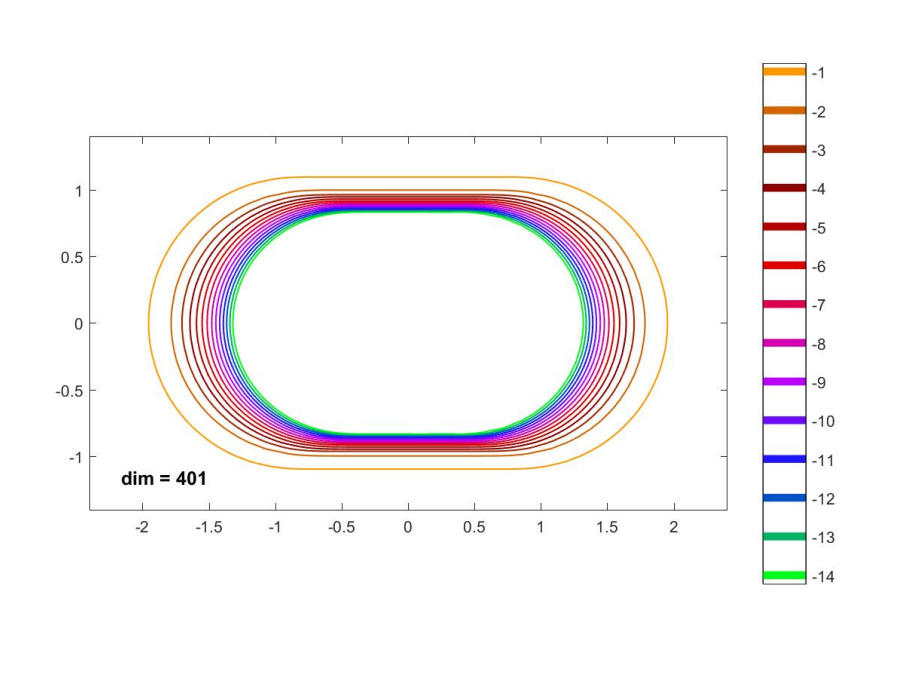
$$\tilde{B}_N^{N+1} = \begin{bmatrix} B_{N-1}^{N+1} & NB_{N-1}^N \boldsymbol{e}_N \\ \boldsymbol{0}^\top & 0 \end{bmatrix} = O$$

(because, by induction, $B_{N-1}^N = O$) and $A^{[1]} = B_N$ is nilpotent. □

You will try in vain to use MATLAB to confirm the nilpotency of $A^{[1]}$ – essentially, for $N \geq 10$ MATLAB produces rubbish. The reason is that this is an example of a twisted Toeplitz matrix* with truly humongous pseudospectrum and great sensitivity to perturbations.

*J. Chapman & L.N. Trefethen, "Wave packet pseudomodes of twisted Toeplitz matrices", *Comm. Pure & Appld Maths* **57** (2004), 1233–1264.





Pseudospectra of $A^{[1]}$ for $N = 25$ and $N = 400$. *(Courtesy of Nick Trefethen.)*

Unlike MATLAB, MAPLE gives zero eigenvalues because it uses integer arithmetic. But it is useless for large N...

Lemma *The spectrum of* $A^{[0]}$ *is* $\{-2r : r = 0, 1, \ldots, N\}$ *and the eigenvector* v_r *corresponding to* $-2r$ *is*

$$v_{r,m} = \begin{cases} (-1)^m \binom{r}{m} \, {}_2F_1\left[\begin{matrix} -N+r, m; \\ r-m+1; \end{matrix} \; -1 \right], & m = 0, \ldots, r \\ (-1)^r \binom{N-r}{m-r} \, {}_2F_1\left[\begin{matrix} -N+m, -r; \\ m-r+1; \end{matrix} \; -1 \right], & m = r, \ldots, N. \end{cases}$$

The proof follows by showing that

$$\mathcal{V}_r(t) := \sum_{m=0}^{N} v_{r,m} t^m = (1+t)^{N+\lambda/2} (1-t)^{-\lambda/2}.$$

$\mathcal{V}_r$ is a polynomial and this is true only if $\lambda = -2r$ – the exact form of v_r follows from expansion of the term on the right.

Let V be the matrix of the above eigenvectors and denote by U the matrix of left eigenvectors s.t. $U = V^{-1}$. Then

$$U_{r,m} = \frac{1}{2^N}\frac{\binom{N}{r}}{\binom{N}{m}} V_{m,r}, \qquad m, r = 0, \ldots, N,$$

therefore $V^{-1} = 2^{-N} V$. More importantly, $\mathrm{e}^{tA^{[0]}} = 2^{-N} V \mathrm{e}^{t\Lambda} V$, where $\Lambda = \mathrm{diag}\,(0, -2, \ldots, -2N)$. Likewise, $\mathrm{e}^{tA^{[1]}} = Z C^{-1} \mathrm{e}^{tE} C \tilde{Z}^{-1}$, where

$$E = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & 0 & 1 \\ 0 & \cdots & \cdots & \cdots & 0 \end{bmatrix}, \qquad C = \mathrm{diag}\,[0!,\ 1!,\ \ldots,\ N!],$$

$$Z_{m,n} = \begin{cases} 0, & m \le n-1, \\ (-1)^{m-n}\binom{N-n}{m-n}, & m \ge n, \end{cases} \qquad \tilde{Z}_{m,n} = \begin{cases} 0, & m \le n-1, \\ \binom{N-n}{m-n}, & m \ge n. \end{cases}$$

I.2 Solvability et. al

The motivation to this work was the observation that the Magnus expansion for $y' = [A^{[0]} + f(t)A^{[1]}]y$ converges with *much* greater radius of convergence than predicted by theory.*

Our ODE system originates in monomolecular reactions and such systems are known to be solvable explicitly. Herewith a proof, valid for more general systems. Recall that a Lie algebra $\mathfrak{g}$ is solvable if there exists $M \geq 0$ s.t. $\mathfrak{g}^{[M]} = \{0\}$, where $\mathfrak{g}^{[0]} = \mathfrak{g}$ and $\mathfrak{g}^{[k+1]} = [\mathfrak{g}^{[k]}, \mathfrak{g}^{[k]}]$.

The solution of our linear ODE lives in the free Lie algebra $\mathcal{F}$ generated by $A^{[0]}, A^{[1]}$. And here cometh magic:

$$[A^{[0]}, A^{[1]}] = -2A^{[1]}.$$

Therefore $\dim \mathfrak{g}^{[1]} = 1$, hence it is a commutative algebra and $\mathfrak{g}^{[2]} = \{0\}$. The algebra is solvable!

*K. Kormann & S. MacNamara, “Error control for exponential integration of the Master Equation”, Technical Report (2015).

With greater generality, consider $\boldsymbol{y}' = [\alpha_0(t)B_0 + \alpha_1(t)B_1]\boldsymbol{y}$ such that $[B_0, B_1] = a_0B_0 + a_1B_1$: we have a 2-dimensional free Lie algebra $\mathcal{F}$. We define the natural embedding $\eta : \mathcal{F} \to \mathbb{R}^2$ s.t.

$$\eta(\beta_0B_0 + \beta_1B_1) = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}.$$

Therefore $\eta([\gamma_0B_0 + \gamma_1B_1, \beta_0B_0 + \beta_1B_1]) = C_{\boldsymbol{\gamma}}\boldsymbol{\beta}$, where

$$C_{\boldsymbol{\gamma}} = \begin{bmatrix} -\gamma_1a_0 & \gamma_0a_0 \\ \gamma_1a_1 & -\gamma_0a_1 \end{bmatrix}.$$

It can be proved easily that

$$C_{\boldsymbol{\gamma}}^n = (-\rho)^nC_{\boldsymbol{\gamma}} \qquad \text{where} \qquad \rho = -(\gamma_1a_0 + \gamma_1a_1).$$

We will prove that the solution of the ODE is $\boldsymbol{y}(t) = \mathrm{e}^{\sigma_0(t)B_0}\mathrm{e}^{\sigma_1(t)B_1}\boldsymbol{y}_0$ for some functions σ_0, σ_1 which obey a nonlinear ODE.

Differentiating,

$$\boldsymbol{y}' = (\sigma_0' B_0 \mathrm{e}^{\sigma_0 B_0} \mathrm{e}^{\sigma_1 B_1} + \sigma_1' \mathrm{e}^{\sigma_0 B_0} B_1 \mathrm{e}^{\sigma_1 B_1}) \boldsymbol{y} = (\sigma_0' B_0 + \sigma_1' \mathrm{Ad}_{\sigma_0 B_0} B_1) \boldsymbol{y},$$

where $\mathrm{Ad}_P Q = \mathrm{e}^P Q \mathrm{e}^{-P}$. Since $\mathrm{Ad}_P Q = \sum_{m=0}^{\infty} (1/m!) \mathrm{ad}_P^m Q$, using the natural embedding

$$\eta\left(\sigma_0' B_0 + \sigma_1' \mathrm{Ad}_{\sigma_0 B_0} B_1\right) = \sigma_0' \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \sigma_1' \sum_{m=0}^{\infty} \frac{\sigma_0^m}{m!} C_{[1,0]}^m \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$= \sigma_0' \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \sigma_1' \mathrm{e}^{\sigma_0 C_{[1,0]}} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$= \sigma_0' \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \frac{\sigma_1'}{a_0\gamma_1 + a_1\gamma_0} \begin{bmatrix} a_0\gamma_0 \left[1 - \mathrm{e}^{-\sigma_0(a_0\gamma_1 + a_1\gamma_0)}\right] \\ a_0\gamma_1 + a_1\gamma_0 \mathrm{e}^{-\sigma_0(a_0\gamma_1 + a_1\gamma_0)} \end{bmatrix} = \begin{bmatrix} \alpha_0 \\ \alpha_1 \end{bmatrix}.$$

The outcome is a set of two coupled nonlinear ODEs,

$$\sigma_0' + \sigma_1' \frac{a_0\gamma_0 \left[1 - \mathrm{e}^{-\sigma_0(a_0\gamma_1 + a_1\gamma_0)}\right]}{a_0\gamma_1 + a_1\gamma_0} = \alpha_0,$$

$$\sigma_1' \frac{a_0\gamma_1 + a_1\gamma_0 \mathrm{e}^{-\sigma_0(a_0\gamma_1 + a_1\gamma_0)}}{a_0\gamma_1 + a_1\gamma_0} = \alpha_1.$$

II. Master equations

Master equations in chemistry and physics describe the time evolution of a system that can be in exactly one of possible states at any time $t \geq 0$ and where the variable is the probability of being in a given state. Such equations are of the form

$$\boldsymbol{y}' = A(t)\boldsymbol{y}, \quad t \geq 0, \qquad \boldsymbol{y}(0) = \boldsymbol{y}_0 \succ \mathbf{0} \in \mathbb{R}^{N+1}, \quad \mathbf{1}^\top \boldsymbol{y}_0 = 1,$$

where $\mathbf{1}^\top A(t) = \mathbf{0}^\top$.* It is easy to prove that the solution $\boldsymbol{y}(t)$ is a probability distribution for every $t \geq 0$: $y_k(t) \geq 0$ and $\mathbf{1}^\top \boldsymbol{y}(t) \equiv 1$.

Master equations are the differential form of the Chapman–Kolmogorov equation from stochastic analysis: if $f_1, \ldots, f_n$ are stochastic processes with joint probability $p_{i_1,\ldots,i_n}(f_1, \ldots, f_n)$ then

$$p_{i_1,\ldots,i_{n-1}}(f_1, \ldots, f_{n-1}) = \int_{-\infty}^{\infty} p_{i_1,\ldots,i_n}(f_1, \ldots, f_n)\, \mathrm{d}f_n.$$

*T. Jahnke & W. Huisinga, "Solving the chemical master equation for monomolecular reaction systems analytically", *J. Math. Biol.* **54** (2007), 1–26.

Many chemists believe that for $N \gg 1$ the only sensible method is Monte Carlo. It is our business to prove them wrong.

Another council of despair is approximating a continuous-time Markov chain, is by a discrete-time Markov chain.* In the simplest case, let

$$\rho(t) = \min\{a_{0,0}(t), \ldots, a_{N,N}(t)\}$$

and set $P(t) = I + \frac{1}{\rho(t)} A(t)$. Given $h_n = \rho^{-1}(t_n)$ and $t_n = t_{n-1} + h_n$, we set

$$\boldsymbol{y}_n = P(t_{n-1})\boldsymbol{y}_{n-1},$$

a 1st-order method. Note that $\|P\|_1 = 1$ and $0 \le P_{k,\ell} \le 1$, hence positivity.

*R.B. Sidje, K. Burrage & S. MacNamara, "Inexact uniformization method for computing transient distributions of Markov chains", *SIAM J. Sci. Comput.* **29** (2007), 2562–2580.

Of particular interest are matrices $A(t)$ which are graph Laplacians: $A_{k,k} \leq 0$ and $A_{k,\ell} \geq 0$ for $k \neq \ell$.

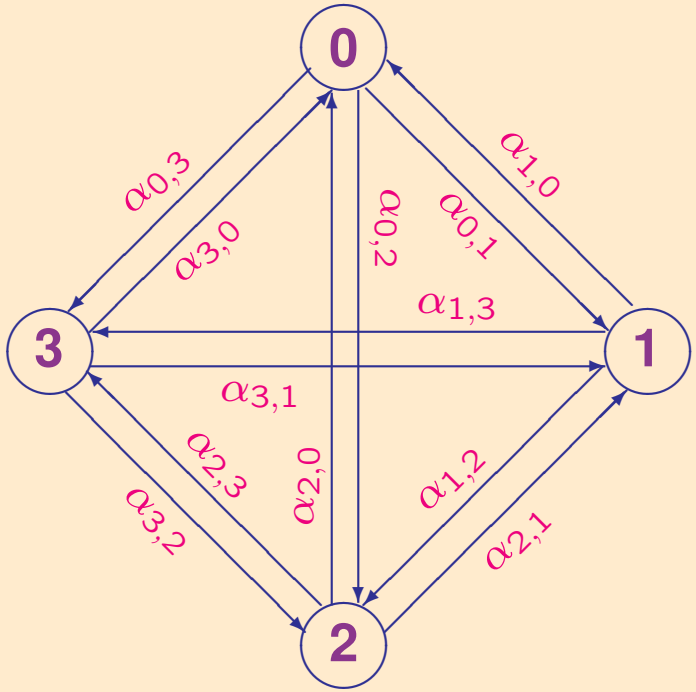
Let $\tilde{\mathfrak{m}}_N$ be the set of such $(N+1) \times (N+1)$ matrices. $|f(t)| \leq 1$ implies that $A^{[0]} + f(t)A^{[1]} \in \tilde{\mathfrak{m}}_N$ so, unknowingly, we have been talking of graph Laplacians.

Graph Laplacians, their eigenvalues and eigenvectors, have huge importance in graph and network theory.

II.1 Why graph Laplacians?

A huge number of processes in biochemical systems (enzyme kinetics, allosteric enzymes, ion channels, G-protein coupled receptors, gene regulation – enough?) can be depicted as a network: Next slide: an example of a labelled graph and the corresponding graph Laplacian

A network:



Its graph Laplacian:

$$A = \begin{bmatrix} -(\alpha_{0,1}+\alpha_{0,2}+\alpha_{0,3}) & \alpha_{1,0} & \alpha_{2,0} & \alpha_{3,0} \\ \alpha_{0,1} & -(\alpha_{1,0}+\alpha_{1,2}+\alpha_{1,3}) & \alpha_{2,1} & \alpha_{3,1} \\ \alpha_{0,2} & \alpha_{1,2} & -(\alpha_{2,0}+\alpha_{2,1}+\alpha_{2,3}) & \alpha_{3,2} \\ \alpha_{0,3} & \alpha_{1,3} & \alpha_{2,3} & -(\alpha_{3,0}+\alpha_{3,1}+\alpha_{3,2}) \end{bmatrix}.$$

Graph Laplacians (which are also related to the Kirkchhoff Theorem and electrical circuits*) have many extra features, e.g. their spectrum lives in $\mathbb{C}^-$. According to Tutte's Matrix-Tree Theorem, the stationary state of $\boldsymbol{y}' = A(t)\boldsymbol{y}$, where $A(t)$ is a graph Laplacian and $\boldsymbol{y}_0 \succ \mathbf{0}$, is positive.†

OUR GOAL: Given the linear system

$$\boldsymbol{y}' = A(t)\boldsymbol{y}, \quad t \geq 0, \qquad \boldsymbol{y}(0) = \boldsymbol{y}_0 \succ \mathbf{0} \in \mathbb{R}^{N+1},$$

where A is a graph Laplacian (or, with greater generality, $\mathbf{1}^\top A(t) \equiv \mathbf{0}^\top$) and $\mathbf{1}^\top \boldsymbol{y}_0 = 1$, find a numerical solution $\boldsymbol{y}_n \approx \boldsymbol{y}(t_n)$, $0 = t_0 < t_1 < \cdots$, such that

(a) $\mathbf{1}^\top \boldsymbol{y}_n \equiv 1$, $n \in \mathbb{N}$;

(b) $\boldsymbol{y}_n \succ \mathbf{0}$, $n \in \mathbb{N}$.

*J. Gunawardena, "A linear framework for time-scale separation in nonlinear biochemical systems", *PLoS ONE* **7** (2012), e36321; I. Mirzaev & J. Gunawardena, "Laplacian dynamics on general graphs", *Bull. Math. Bio.* **75** (2013), 2118–49.

†$A(t)$ being singlular, this is a statement about the eigenvector space corresponding to zero eigenvalues.

II.2 The devil's dilemma: (a) or (b)?

Conserving positivity is very difficult with methods of order ≥ 2* while conserving linear invariants is trivial. Thus, any reasonable multistep, Runge–Kutta or Lie-group method conserves (a) and fails on (b). What to do?

We can conserve (b) by a simple change of variables, $y_k(t) = \mathrm{e}^{\theta_k(t)}$, $k = 0, \ldots, N$. This results in the nonlinear ODE

$$\theta_k' = \sum_{\ell=0}^{N} A_{k,\ell}(t) \mathrm{e}^{\theta_\ell - \theta_k}, \qquad k = 0, \ldots, N$$

and positivity is built-in – except that (a) becomes a nasty nonlinear invariant,

$$\sum_{k=0}^{N} \mathrm{e}^{\theta_k(t)} \equiv 1,$$

which is difficult to enforce and of questionable stability.

*C. Bolley & M. Crouzeix, "Conservation de la positivité lors de la discrétisation des problèmes d'évolution paraboliques", *RAIRO Anal. Numér.* **12** (1978), 237–245.

II.3 Lie-group methods

Back to square number one: conserve (a) automatically and do 'something' about (b). Given that the system is linear, we use a Lie-group method. Note that $\mathfrak{m}_N$, the set of $(N+1)\times(N+1)$ matrices A s.t. $\mathbf{1}^\top A = \mathbf{0}^\top$, is a Lie algebra.* The corresponding Lie group $\mathfrak{M}_N$ is the set of matrices B s.t. $\mathbf{1}^\top B = \mathbf{1}^\top$.

We have three options:

- Magnus,
- Fer,
- Canonical coordinates of the second kind (Marthinsen & Owren).

They all map $\mathfrak{m}_N$ to $\mathfrak{M}_N$. But... what about positivity?

*The set $\tilde{\mathfrak{m}}_N$ of graph Laplacians is a cone in $\mathfrak{m}_N$.

III. Flags of methods

No specific Lie-group is method can be assured of producing $\boldsymbol{y}_n \succ \mathbf{0}$. However, all such methods converge, hence for sufficiently large order, once the error is sufficiently small, if $\boldsymbol{y}_{n-1} \succcurlyeq \mathbf{0}$, they are assured to produce $\boldsymbol{y}_n \succ \mathbf{0}$.

Similarly to flag manifolds, we say that a set of numerical methods $\{\mathcal{M}_{r,h}\}_{r=1}^{r^\star}$ is a flag if

1. Each $\mathcal{M}_{r,h}$ is of order p_r, i.e.

$$\mathcal{M}_{r,h}[t_{n-1}, \boldsymbol{y}(t_{n-1})] - \boldsymbol{y}(t_n) = \mathcal{O}(h^{p_k+1}),$$

   where $t_n = t_{n-1} + h$;

2. $p_1 < p_2 < \cdots$, hence $\lim_{r\to\infty} p_r = \infty$;

3. Each method $\mathcal{M}_{r+1,h}[t_{n-1}, \boldsymbol{y}_{n-1}]$ can be obtained with marginal cost utilising the computations that have led to $\mathcal{M}_{r,h}[t_{n-1}, \boldsymbol{y}_{n-1}]$.

III.1 Flag Fer?

We have tried to construct flag methods for Fer and for CCSK – the problem is in recycling past values. Fer is particularly enticing: with exact integration

$$B_0(t) = A(t),$$
$$B_m(t) = \sum_{k=1}^{\infty} \frac{(-1)^k k}{(k+1)!} \mathrm{ad}^k_{\int_{t_{n-1}}^{t_{n-1}+h} B_{m-1}(t)}, \qquad m \geq 1,$$
$$\mathcal{M}_{r,h}[t_{n-1}, \boldsymbol{y}_{n-1}] = \mathrm{e}^{\int_{t_{n-1}}^{t_{n-1}+h} B_0(\xi)\,\mathrm{d}\xi} \cdots \mathrm{e}^{\int_{t_{n-1}}^{t_{n-1}+h} B_{r-1}(\xi)\,\mathrm{d}\xi} \boldsymbol{y}_{n-1}.$$

The order is $p_r = 2^{r+1} - 2$ and we can recycle values,

$$\mathcal{F}_{r,h} = \mathrm{e}^{\int_{t_{n-1}}^{t_{n-1}+h} B_0(\xi)\,\mathrm{d}\xi} \cdots \mathrm{e}^{\int_{t_{n-1}}^{t_{n-1}+h} B_{r-1}(\xi)\,\mathrm{d}\xi}$$
$$\Rightarrow \quad \mathcal{F}_{r,h} = \mathcal{F}_{r-1,h} \mathrm{e}^{\int_{t_{n-1}}^{t_{n-1}+h} B_{r-1}(\xi)\,\mathrm{d}\xi}, \qquad \mathcal{M}_{r,h} = \mathcal{F}_{r,h} \boldsymbol{y}_{n-1}.$$

The problem is discretising integrals: they must be all computed to order $p_{r^\star}$ and this, plus all the exponentials, becomes prohibitively expensive.

III.2 Flag Magnus

Given that Magnus is implemented in a time-symmetric manner, $p_k = 2k$. The idea is to discretise integrals from the outset, using a Gauss–Legendre basis of order $2r^\star$. For example, for $r^\star = 3$, we compute

$$A_1, A_3 = hA(t_{n-1} + (\tfrac{1}{2} \mp \tfrac{\sqrt{15}}{10})h), \qquad A_2 = hA(t_{n-1} + \tfrac{1}{2}h);$$

$$B_1 = A_2, \qquad B_2 = \tfrac{\sqrt{15}}{3}(A_3 - A_1), \qquad B_3 = \tfrac{10}{3}(A_3 - 2A_2 + A_1);$$

$$\Omega_1 = B_1 \qquad \text{(order 2)},$$

$$\Omega_2 = \Omega_1 + \tfrac{1}{12}B_3 + \tfrac{1}{12}[B_1, B_2], \qquad \text{(order 4)},$$

$$\Omega_3 = \Omega_2 - \tfrac{1}{240}[B_2, B_3] - \tfrac{1}{240}[B_2, [B_1, B_2]] + \tfrac{1}{360}[B_1, [B_1, B_3]],$$

$$\text{(order 6)};$$

$$\mathcal{M}_{r,h} = \mathrm{e}^{\Omega_r} \boldsymbol{y}_{n-1}.$$

Let $\Theta_1 = \Omega_1$, hence $\mathcal{M}_{1,h} = \mathrm{e}^{\Theta_1} \boldsymbol{y}_{n-1}$ and set

$$\begin{aligned}\Theta_2 &= \mathrm{BCH}(\Omega_2, -\Theta_1) = \mathrm{BCH}\Big(B_1 + \tfrac{1}{12}B_3 + \tfrac{1}{12}[B_1, B_2], -B_1\Big) \\ &= \frac{1}{12}B_3 + \tfrac{1}{12}[B_1, B_2] + \tfrac{1}{12}[B_1, B_3] + \tfrac{1}{12}[B_1, [B_1, B_2]] = \mathcal{O}\big(h^3\big)\,.\end{aligned}$$

Unlike Ω_2, Θ_2 is not time symmetric. Hence $\mathcal{M}_{2,h} = \mathrm{e}^{\Theta_2}\mathcal{M}_{1,h}$.

Likewise, $\Theta_3 = \mathrm{BCH}(\Omega_3, \mathrm{BCH}(-\Theta_1, -\Theta_2))$ and, after long computation,

$$\begin{aligned}\Theta_3 = &-\tfrac{1}{24}[B_1,B_3] - \tfrac{1}{24}[B_1,[B_1,B_2]] - \tfrac{1}{240}[B_2,[B_1,B_2]] + \tfrac{1}{60}[B_1,[B_1,B_3]] \\ &+ \tfrac{1}{72}[B_1,[B_1,[B_1,B_2]]] + \tfrac{1}{480}[B_1,[B_2,B_3]] + \tfrac{1}{480}[B_1,[B_2,[B_1,B_2]]] \\ &+ \tfrac{7}{48}[B_1,[B_1,[B_1,B_3]]] + \tfrac{1}{288}[B_1,[B_1,[B_1,[B_1,B_2]]]] + \mathcal{O}(h^7).\end{aligned}$$

We have $\mathcal{M}_{3,h} = \mathrm{e}^{\Theta_3}\mathcal{M}_{2,h}$ and $\Theta_3 = \mathcal{O}\big(h^4\big)$.

IV. Computing exponentials

IV.1 Padé approximations

Padé approximations are fully consistent with taking $\mathfrak{m}_N$ to $\mathfrak{M}_N$. Note that, in a flag method, we need to compute everything to the 'top order' $p_{r^\star}$, otherwise computations cannot be recycled. Hence (and adding A-stability to the mix: stiffness can be an issue!)

$$\begin{aligned}
\Theta_1 = \mathcal{O}(h) \quad &\Rightarrow \quad \mathrm{e}^{\Theta_1} = r_{3/3}(\Theta_1) + \mathcal{O}\left(h^7\right), \\
\Theta_2 = \mathcal{O}\left(h^3\right) \quad &\Rightarrow \quad \mathrm{e}^{\Theta_2} = r_{1/1}(\Theta_1) + \mathcal{O}\left(h^9\right), \\
\Theta_3 = \mathcal{O}\left(h^4\right) \quad &\Rightarrow \quad \mathrm{e}^{\Theta_3} = r_{1/1}(\Theta_1) + \mathcal{O}\left(h^{12}\right).
\end{aligned}$$

The downside is that computing $r_{3/3}(\Theta_1)v$ is expensive and either we need to cube Θ_1 – danger of large numbers and poor conditioning – or factorize it at the price of complex arithmetic.

IV.2 Generalised polar decomposition

Recall the basis facts:* $\mathfrak{g}$ is a Lie algebra, $\mathcal{G}$ the corresponding Lie group, and $\sigma : \mathfrak{g} \to \mathfrak{g}$ an involutory inner automorphism: $\sigma(X) = \Sigma X \Sigma^{-1}$, where $\Sigma \in \mathcal{G}$, $\Sigma^2 = I$. Let

$$\mathfrak{k} = \{X \in \mathfrak{g} : \sigma(X) = X\}, \qquad \mathfrak{p} = \{X \in \mathfrak{g} : \sigma(X) = -X\}$$

be the sets of fixed points and anti-fix points of σ. Then

1. $\mathfrak{k}$ is a Lie algebra, while $\mathfrak{p}$ is a Lie triple space;

2. $\mathfrak{g} = \mathfrak{k} \oplus \mathfrak{p}$. Specifically, let $X \in \mathfrak{g}$. Then $K = \frac{1}{2}[X + \sigma(X)] \in \mathfrak{k}$, $P = \frac{1}{2}[X - \sigma(X)] \in \mathfrak{p}$ and $X = K + P$;

3. It is possible to construct $\tilde{K} : \mathbb{R}_+ \to \mathfrak{k}$, $\tilde{P} : \mathbb{R}_+ \to \mathfrak{p}$, both in the free Lie algebra generated by K and P, such that $\mathrm{e}^X = \mathrm{e}^{\tilde{P}} \mathrm{e}^{\tilde{K}}$.

*H.Z. Munthe-Kaas, G.R.W. Quispel & A. Zanna, "Generalized polar decompositions on Lie groups with involutive automorphisms', *Found. Comput. Math.* **3** (2001), 297–324.

Choose $\Sigma = I - 2e_1e_1^\top$, then P is nonzero in just its first row and column, hence $\operatorname{rank} P \leq 2$ and $\mathrm{e}^{\tilde{P}}$ is trivial to compute. Then continue likewise 'slicing' the Lie algebra K with $\Sigma = I - 2e_2e_2^\top$ and so on – finally, we have $\mathrm{e}^X = \mathrm{e}^{\tilde{P}_1}\mathrm{e}^{\tilde{P}_2}\cdots\mathrm{e}^{\tilde{P}_r}$ and each exponential can be evaluated trivially.

An improvement:* First bring X to an upper Hessenberg form using Householder reflections, $X = HYH$, hence $\mathrm{e}^X = H\mathrm{e}^Y H$. Apply GPD to Y but slicing from the bottom. Each iteration 'contaminates' the upper Hessenberg form with few entries at the bottom right of the matrix, which can be dealt with using Givens rotations.

However... $X \in \mathfrak{m}_N \not\to Y \in \mathfrak{m}_N$ and Householder reflections destroy the structure of $\mathfrak{m}_N$. What to do?

*A. Iserles & A. Zanna, "Efficient computation of the matrix exponential by generalized polar decompositions', *SIAM J. Numer. Anal.* **42** (2005), 2218–2256.

IV.3 Upper Hessenberg in $\mathfrak{m}_N$

Firstly, we need to choose $\Sigma \in \mathfrak{M}_N$, $\Sigma^2 = I$ – and here is one:

$$\Sigma = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & 0 \\ 0 & \cdots & 0 & 1 & 2 \\ 0 & \cdots & 0 & 0 & -1 \end{bmatrix}.$$

In place of Householder, we adapt a Gaussian elimination-like algorithm from Golub & Van Loan to $\mathfrak{m}_N$. Let $\boldsymbol{\theta}$ be the leading column of $X \in \mathfrak{m}_N$: of course, $1^\top \boldsymbol{\theta} = 0$. We seek a nonsingular matrix S_1 s.t. the leading column of $X_1 = S_1 X S_1^{-1}$ is consistent with upper Hessenberg and $X_1 \in \mathfrak{m}_N$.

Unless $\boldsymbol{\theta}$ has at least two nonzero elements, we chose S_1 as a permutation matrix. Otherwise use permutation to bring the largest two terms to θ_0 and θ_1 and note that they are both nonzero.

Let

$$\boldsymbol{a} = -\frac{1}{\theta_1}\begin{bmatrix} 0 \\ \theta_0 \\ \theta_2 \\ \vdots \\ \theta_N \end{bmatrix}, \qquad \boldsymbol{b} = \frac{1}{\theta_0}\begin{bmatrix} 0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_N \end{bmatrix}$$

and set

$$S_1 = [\boldsymbol{e}_1,\ \boldsymbol{a},\ \boldsymbol{e}_2,\ \cdots,\ \boldsymbol{e}_N], \qquad S_1^{-1} = [\boldsymbol{e}_1,\ \boldsymbol{b},\ \boldsymbol{e}_2,\ \cdots,\ \boldsymbol{e}_N].$$

In order to continue and generate $S_2, S_3, \ldots$, gradually, 'building' an upper Hessenberg matrix, we need some subtle combinatorial considerations which we omit.

IV.4 Upper Hessenberg commutators

Once $\tilde{X}$ is upper Hessenberg, the cost of linear algebra collapses:

$$K = \begin{bmatrix} \times & \times & \cdots & \cdots & \times \\ \times & \times & \cdots & \cdots & \times \\ 0 & \times & \ddots & & \times \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \times & \times \end{bmatrix}, \qquad P = \begin{bmatrix} 0 & \cdots & 0 & 0 & \times \\ 0 & \cdots & 0 & 0 & \times \\ \vdots & & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & \times \\ 0 & \cdots & 0 & \times & \times \\ 0 & \cdots & 0 & \times & \times \end{bmatrix}$$

$$\Rightarrow \qquad [K, P] = \begin{bmatrix} 0 & \cdots & 0 & 0 & 0 & \times \\ 0 & \cdots & 0 & 0 & 0 & \times \\ \vdots & & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & \times \\ 0 & \cdots & 0 & \times & \times & \times \\ 0 & \cdots & 0 & \times & \times & \times \end{bmatrix}$$

and so on. Few rogue elements under the first subdiagonal can be 'cleaned' by further low-dimensional transformations. The outcome is an algorithm that computes $\mathrm{e}^{\Theta}\boldsymbol{v}$ in $\mathcal{O}\left(N^3\right)$ flops to very high accuracy.

V. Further thoughts

1. In GNI we are typically interested in conserving invariants, i.e. equalities. What about conserving inequalities? Whether you call it GNI or not, it is important. And difficult.

2. Master Equations are fascinating and they have received very little expert numerical attention. Ditto for the Kossakowski–Linblad Equation, its quantum-mechanics counterpart, and, for the more ambitious, the Kardar–Parisi–Zhang System, its scaling limit.

3. Letting $N \to \infty$, chemists obtain the differential Volterra equation

$$\boldsymbol{y}'(t) = \int_0^t A(t-\xi)\boldsymbol{y}(\xi)\,\mathrm{d}\xi.$$

   Numerical theory?


THE
E:N:D
THE
E:N:D
THE
E:N:D