Exponential integrators and functions of the matrix exponential

Paul Matthews, Stephen Cox, Hala Ashi and Linda Cummings
School of Mathematical Sciences,
University of Nottingham, UK

- Introduction to exponential time differencing methods
- Examples of application to highly oscillatory problems
- Comparison of methods for evaluating coefficients

Motivation – Semilinear PDEs

Many nonlinear PDEs of interest are *semilinear*: the term with the highest spatial derivative is linear. Examples: N-S, K-S, KdV, R-D, S-H, NLS...

Spatial discretisation (spectral or finite difference) $\rightarrow$ system of ODEs. For example with a Fourier spectral method,

$$\frac{\partial w}{\partial t} = \frac{\partial^m w}{\partial x^m} + N(w).$$

Take discrete Fourier transform, $w \rightarrow \hat{w}$:

$$\frac{d\widehat{w}_n}{dt} = (ik_n)^m \hat{w}_n + \hat{N}(w).$$

This set of ODEs is *stiff* and the stiffness arises from the *linear* term.

- Time step is limited by linear term in equation for highest Fourier mode.
- Explicit methods need small time step h. Fully implicit methods are slow.
- It is best to handle the stiff linear term implicitly or exactly.

Three classes of methods

Consider the system

$$\dot{U} = c\,U + F(U,t), \qquad (1)$$

where the matrix c has
large and negative eigenvalues (for dissipative PDEs),
large and imaginary eigenvalues (for dispersive PDEs).
F represents nonlinear or forcing terms.

Now consider three main classes of methods for (1).
All avoid the small-timestep problem of explicit methods.

1. Linearly implicit method

Also known as 'IMEX' (Implicit-Explicit) or semi-implicit.
See Ascher, Ruuth & Wetton (1995). Well known.

Use implicit method (e.g. backwards Euler) for linear term, explicit method (e.g. Euler) for nonlinear term:

$$U_{n+1} = (I - ch)^{-1}(U_n + hF_n). \tag{LI1}$$

The implicit part is stable for *any* h, (method is A-stable).
Works well on the slow manifold.

Disadvantages:

- Cannot extend beyond second order without losing A-stability (Dahlquist second stability barrier).
- Fails to capture rapid exponential decay / rapid oscillations.

Why solve the linear part implicitly when we can solve it exactly . . .

2. Integrating factor method (Lawson 1967)

Multiply (1) through by integrating factor e^{-ct}

$$d(e^{-ct}U)/dt = e^{-ct}F(U,t).$$

Now apply an explicit method (e.g. Euler):

$$U_{n+1} = e^{ch}(U_n + hF_n). \tag{IF1}$$

Advantages:

- Can extend to any order, using any multi-step or RK method.
- Fast exponential part is handled exactly.

Disadvantages:

- Inaccurate for problems where F is slowly varying (poor on slow manifold) because the fast time appears in the F term $\rightarrow$ large error constants.
- Fixed points wrong, with relative error $O(ch)$.

Often used with spectral methods (books by Canuto, Trefethen, Boyd).

3. Exponential time differencing (Certaine 1960)

Exact solution to

$$d(e^{-ct}U)/dt = e^{-ct}F(U,t)$$

is

$$U_{n+1} = e^{ch}U_n + \int_0^h e^{-c(\tau-h)}F(U(t_n+\tau), t_n+\tau)\, d\tau. \tag{2}$$

Approximate F as a polynomial in t and do the integral exactly.
Assuming $F =$ constant $= F_n$ we have

$$U_{n+1} = e^{ch}U_n + (e^{ch} - I)c^{-1}F_n. \tag{ETD1}$$

Advantages:

- Exponential behaviour correct.
- Smaller error constants than IF.
- Preserves fixed points.
- Can extend to any order by constructing multistep or RK methods.

Exponential time differencing – History

LI and IF methods appear frequently in the literature. The ETD method is less common. I have found it in only one textbook (Miranker 1981).

ETD has been re-invented very many times over the years. Each inventor gives the method a different name (ELP, EPI, QSS, PSS, slave-frog, method of patches, . . .)

ETD1 is used in computational electromagnetism
(papers by Holland, Schuster, book by Taflove, 1995).
The term 'Exponential time differencing' comes from this field.

• Certaine (1960) "The solution of ordinary differential equations with large time constants". Key ideas. How to develop multistep ETD methods of any order. Formulas for implicit forms of ETD2 and ETD3.

• Nørsett (1969). Formulas for multistep ETD methods of any order using generating function.

- Friedli (1978). (Auf Deutsch). Consistency conditions for ETDRK methods.

- Hochbruck Lubich & Selhofer (1998). Re-linearising. Krylov subspace method.

- Beylkin et al. (1998). Formulation, stability regions.

- Cox & Matthews (2002). Formulas for ETDRK methods of order up to 4. Explanation of superiority of ETD over LI and IF, illustrated with ODE and PDE tests.

- Kassam & Trefethen (2005). Contour integral method for evaluating coefficients. Tests 5 types of 4th order method on 4 PDEs – ETDRK4 is best. Matlab codes.

- Berland, Skaflestad & Wright (NTNU) Matlab package.

- Minchev & Wright – review paper – unpublished.

Definitions

Consider the system

$$\dot{U} = c\,U + F(U,t).$$

An *exponential integrator* is a numerical method that is exact when $F = 0$.

An *exponential time differencing method* (or *Certaine's method*) is a numerical method that is exact when $F =$ constant.

Oscillatory PDE Examples

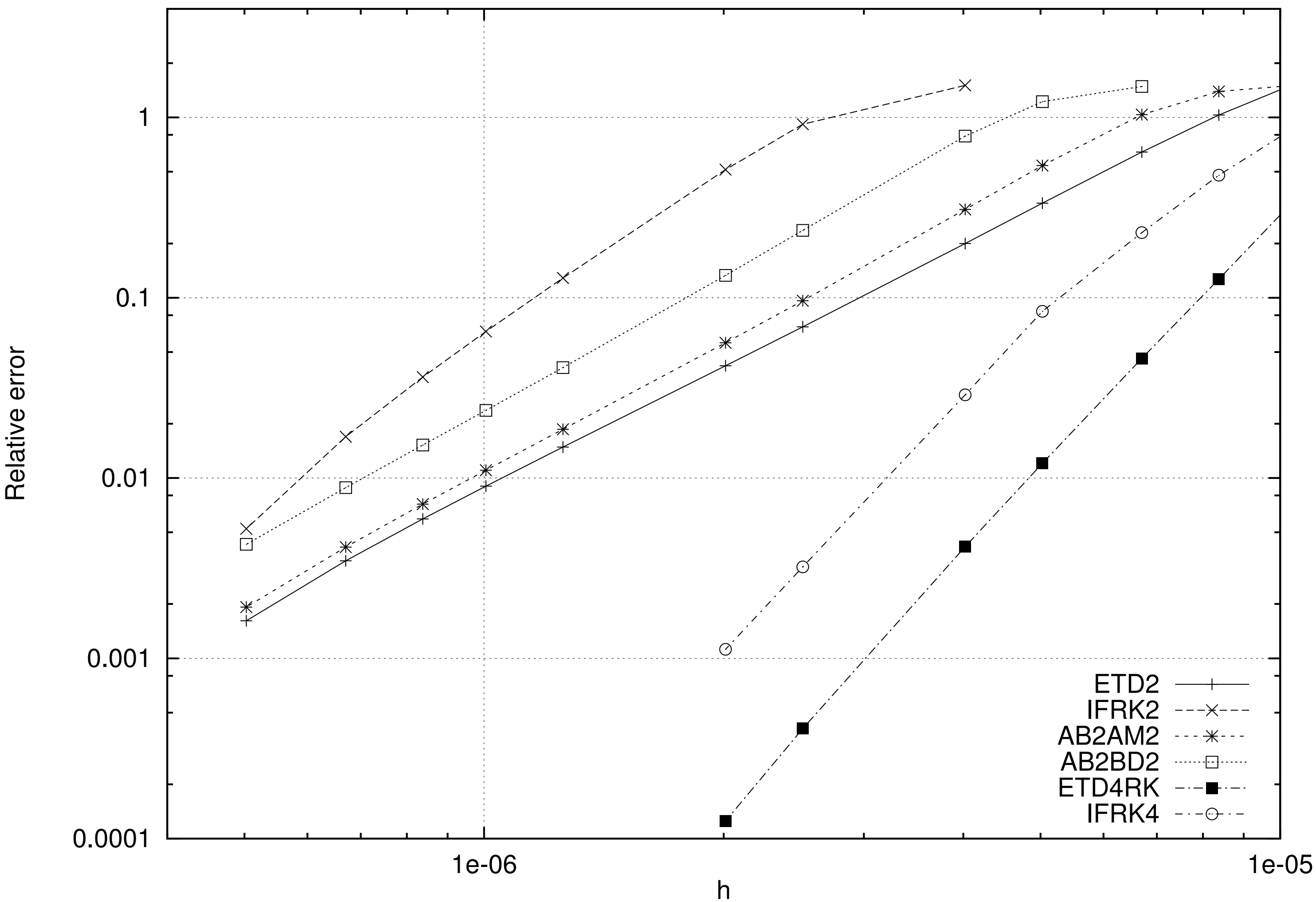
KdV equation

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{\partial^3 u}{\partial x^3} = 0,$$

with soliton solution $u = f(x - ct)$, where $f(x) = 3c\,\mathrm{sech}^2(c^{1/2}x/2)$, $c = 625$. Follow one period, i.e. up to $t = 2\pi/c$.

2nd order methods: ETD better than LI better than IF.

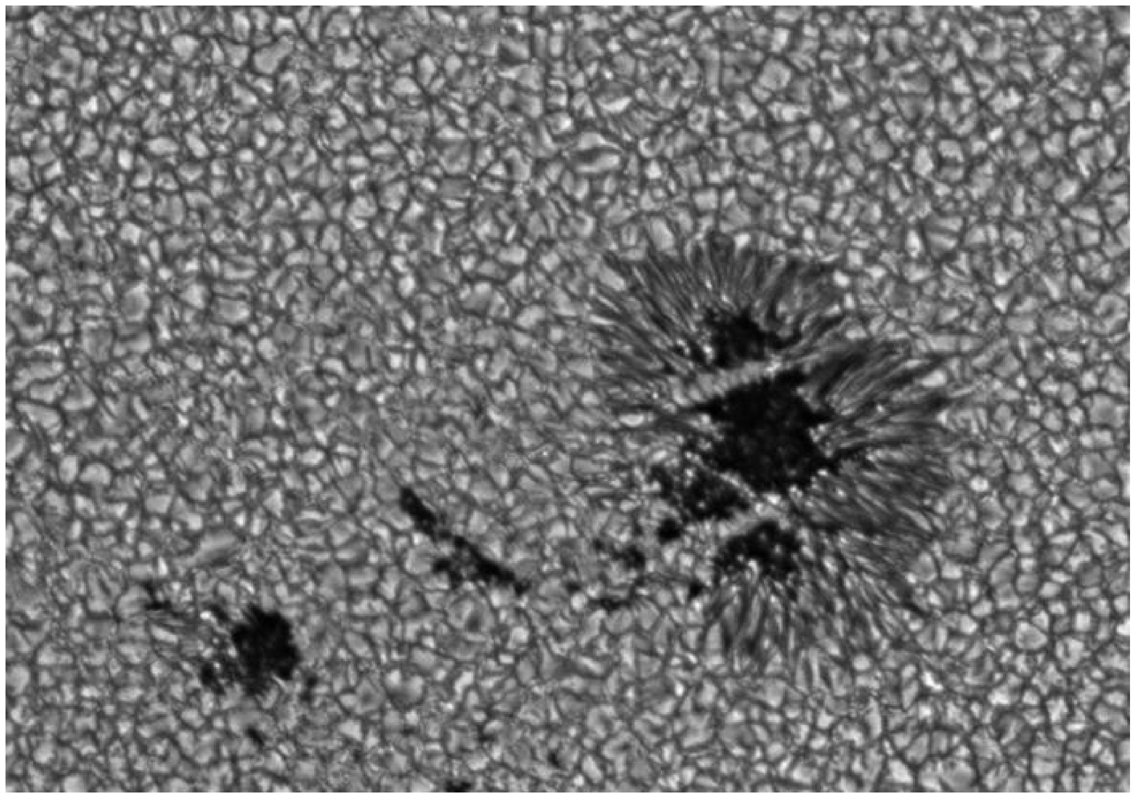
4th order methods: ETDRK better than IFRK (LI methods unstable).



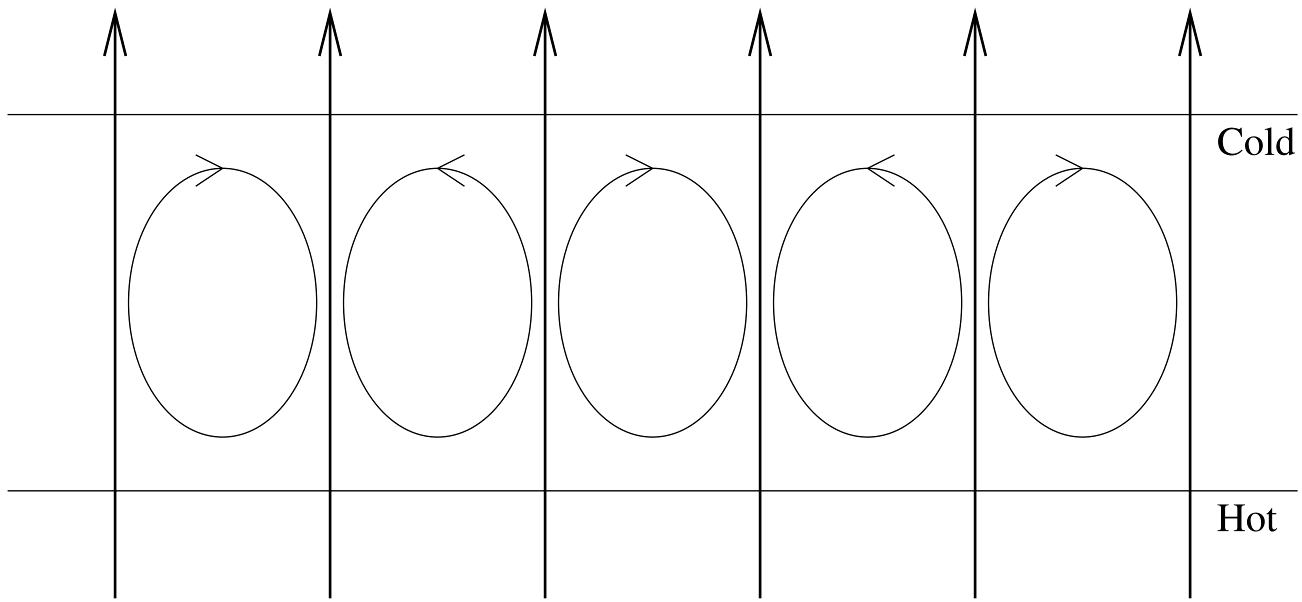
Relative error
1
0.1
0.01
0.001
0.0001
1e-06
1e-05
h
ETD2
IFRK2
AB2AM2
AB2BD2
ETD4RK
IFRK4

Magnetoconvection

Modelling convection in sunspots (Proctor, Weiss, . . .)

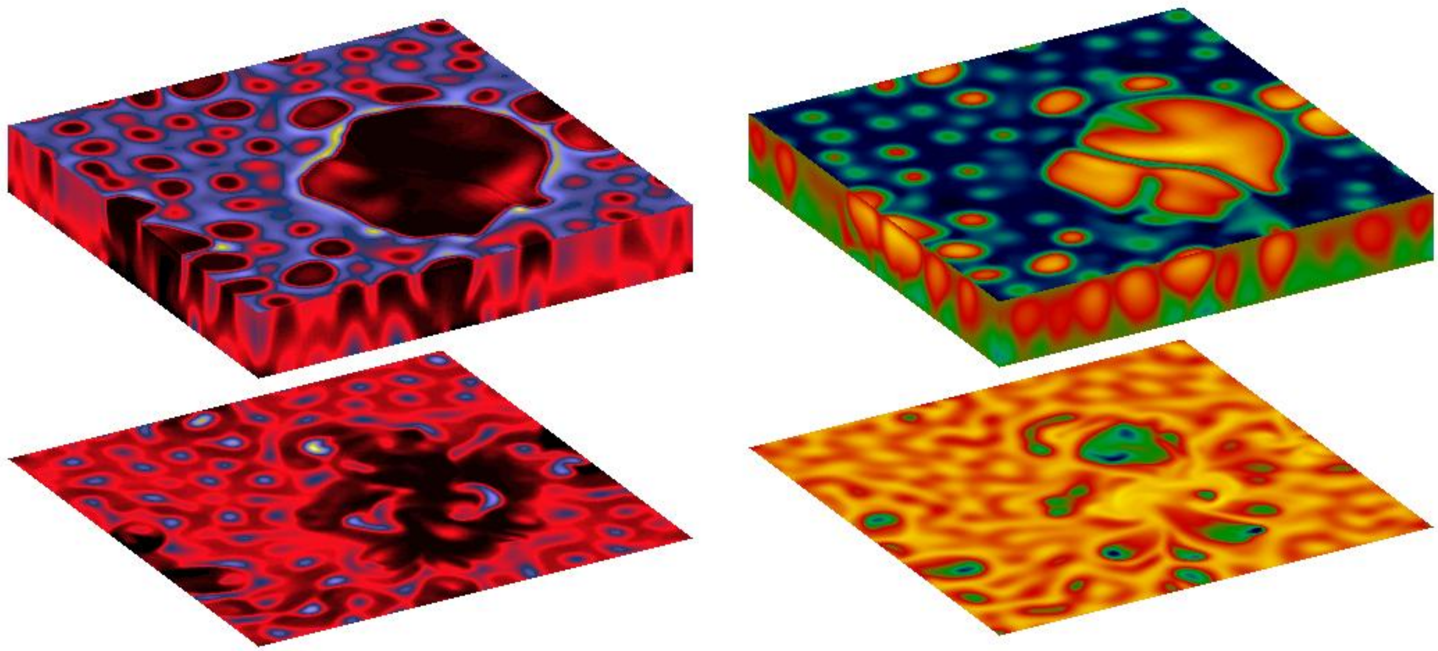


Thermal convection in an imposed vertical magnetic field

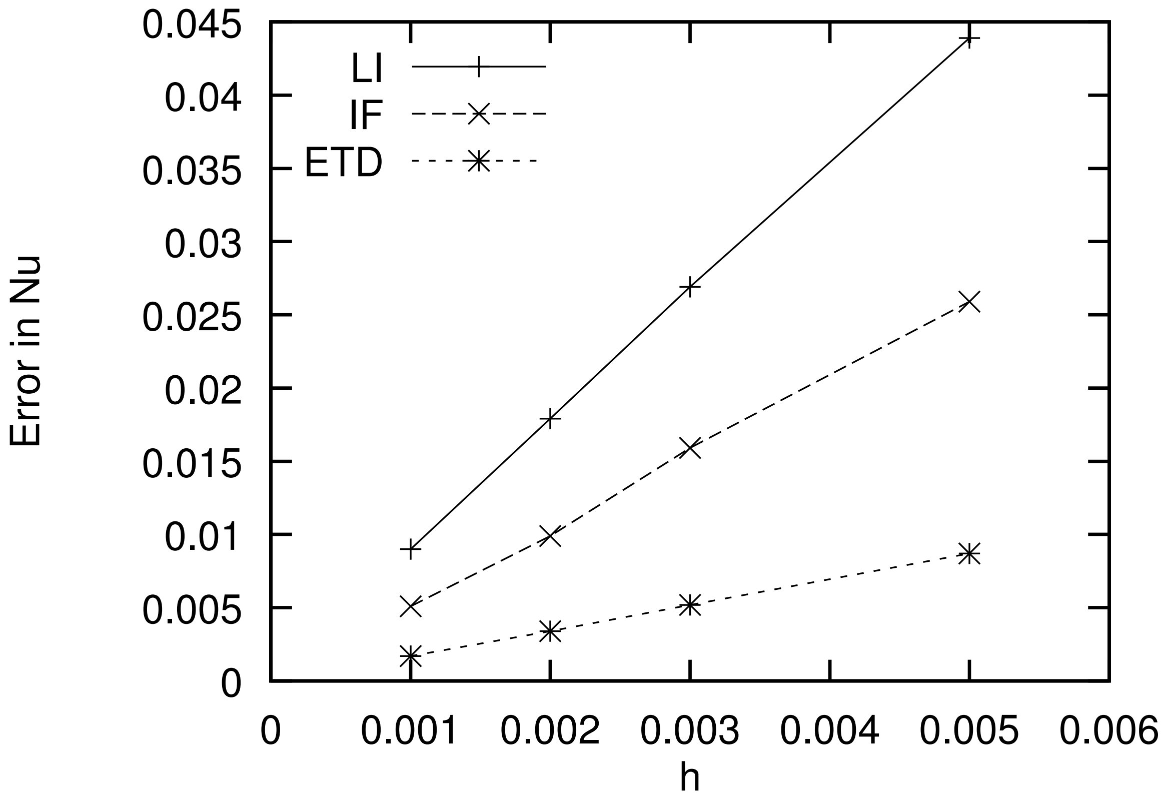


Solve coupled equations for velocity, magnetic field, heat.

Convection is highly oscillatory if field strength Q is large.



Standing wave solution with $Q = 1000$, $Ra = 3000$. ("Moderately nonlinear" regime). First order methods only:



Exponential time differencing coefficients

ETD methods of order n involve the first n 'ϕ functions'

$$\begin{aligned}
\phi_0(z) &= e^z, \\
\phi_1(z) &= (e^z - 1)/z, \\
\phi_2(z) &= (e^z - 1 - z)/z^2, \\
\phi_n(z) &= \frac{1}{z^n}\left(e^z - \sum_{k=0}^{n-1}\frac{z^k}{k!}\right) = \sum_{k=0}^{\infty}\frac{z^k}{(k+n)!}
\end{aligned}$$

These functions become increasingly difficult to evaluate as n increases, for small z, due to subtraction errors.

In the scalar case we can simply use the explicit formula for large z and truncated Taylor series for small z.

But in the matrix case, this does not work well, since typically z has large and small (or zero) eigenvalues.

Remarks:

- To solve $\dot{U} = cU + F(U,t)$ with fixed time step h and fixed linearisation, we only need to find the $\phi_n(ch)$ functions once at the start.
- For Fourier spectral methods c is diagonal so the problem reduces to the scalar case.
- For finite difference or finite element methods, c is sparse. In this case, e^{ch} and $\phi_n(ch)$ are 'fairly sparse' (Iserles 2000).
- If we are re-linearising each step, there is no need to find the ϕ_n functions (Saad, Aluffi-Pentini). But need a matrix exponential each step.

Tests:

Compare methods to find the functions $\phi_n(Mh)$ where M is the 40×40 finite difference matrix for the second derivative,
$M(j,j) = -2$, $M(j, j \pm 1) = 1$, for different values of h.
(Similar results for FD first derivative and Chebyshev matrix).

(i) Cauchy integral method (Kassam & Trefethen)

Use Cauchy Integral Formula (matrix form):

$$f(z) = \frac{1}{2\pi i} \int_{\Gamma} f(t)(tI - z)^{-1} \, dt$$

where the contour Γ encloses all the eigenvalues of z. Approximate using periodic trapezium rule, using a circle of N points $t_j = z_0 + re^{i\theta_j}$:

$$f(z) \approx \frac{1}{N} \sum_{j=1}^{N} re^{i\theta_j} f(t_j)(t_j I - z)^{-1}.$$

- Fairly slow – need N matrix inverses
- Need to know where eigenvalues are
- Inaccurate if contour too large: error $\sim r^N/N!$

(Recent improved method, Schmelzer & Trefethen 2007)

(ii) Scaling and squaring

Choose m so that $2^{-m}||z|| < \delta$, some threshold value ~ 1.
Use Taylor series or Pade to find $\phi_n(2^{-m}z)$.
Then apply squaring rules m times to find $\phi_n(z)$.

$$\begin{aligned}
\phi_0(2z) &= (\phi_0(z))^2, \\
\phi_1(2z) &= (\phi_0(z)\phi_1(z) + \phi_1(z))/2, \\
\phi_2(2z) &= (\phi_1(z)^2 + 2\phi_2(z))/4.
\end{aligned}$$

- Widely used for the matrix exponential (Matlab, Higham)
- Speed depends on number of squarings m required
- Accuracy decreases as m increases – relative error $\sim 2^m$
- Different forms of scaling rules give very similar accuracy

(iii) 'Compound matrix' method

Consider the matrix

$$M_4 = \begin{pmatrix} z & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Then it can easily be shown that

$$e^{M_4} = \begin{pmatrix} \phi_0(z) & \phi_1(z) & \phi_2(z) & \phi_3(z) \\ \dots & \dots & \dots & \dots \end{pmatrix}$$

(minor variation results of Saad, Sidje). So, if we have a subroutine for the matrix exponential (`expm`) we can find all the ϕ_n functions with one call.

- Very easy to code
- As accurate as the underlying matrix exponential subroutine
- Inefficient – slow if z is large (1.5 sec if z is 128x128)

(iv) Diagonalisation method

If V is the matrix of eigenvectors of z then $z = VDV^{-1}$
where D is the diagonal matrix of eigenvalues λ, so

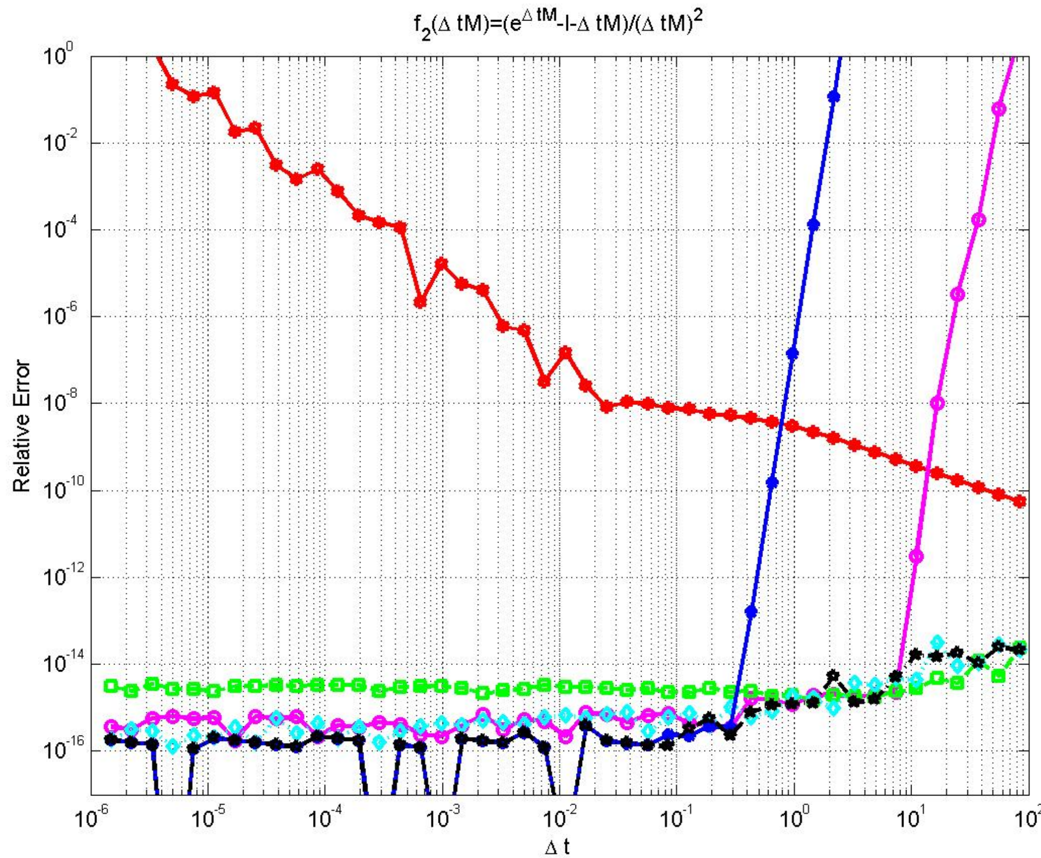
$$\phi_n(z) = V\phi_n(D)V^{-1}.$$

Find $\phi_n(D)$ as in scalar case using explicit formula for large λ, Taylor series for small λ.

- Fast and accurate for 'nice' matrices (eg 2nd derivative FD)

- Fails if z is not diagonalisable

Results for ϕ_2



Explicit formula, Taylor series, Cauchy (128 points), Scaling and squaring, Diagonalisation, Composite matrix

Summary

- Discretisation of semilinear PDE $\longrightarrow$ Linearly stiff system of ODEs.

- Can solve linear term exactly at almost no extra cost.
  Two types of exponential integrator: IF (Lawson), ETD (Certaine).
  ETD better than IF in most applications.

- Need to find ϕ_n functions at start of computation.
  Easy for Fourier spectral methods.
  For other space discretisations there are several suitable methods, including diagonalisation, scaling and squaring.

- More work needed on application to serious large-scale problems.