# Splitting and composition methods for the time dependent Schrödinger equation

S. Blanes,

joint work with F. Casas and A. Murua

*Instituto de Matemática Multidisciplinar*
*Universidad Politécnica de Valencia, SPAIN*

***Geometric Numerical Integration***

***Oberwolfach, 21-25 March 2011***

# The Schrödinger equation

We consider different families of **Geometric Integrators** for solving the SE

$$i\frac{\partial}{\partial t}\psi(x,t) = \left(-\frac{1}{2\mu}\nabla^2 + V(x,t)\right)\psi(x,t)$$

The solution of the discretised **autonomous** equation is given by

$$i\,\frac{d}{dt}\mathbf{c}(t) = \mathbf{H}\mathbf{c}(t) \quad \Rightarrow \quad \mathbf{c}(t) = e^{-it\mathbf{H}}\mathbf{c}(0)$$

where $\mathbf{c} = (c_1, \ldots, c_N)^T \in \mathbb{C}^N$ and $\mathbf{H} = \mathbf{T} + \mathbf{V} \in \mathbb{R}^{N\times N}$ Hermitian matrix.
Fourier methods are frequently used

$$(\mathbf{V}\mathbf{c})_i = V(x_i)c_i \qquad N \text{ products}$$
$$\mathbf{T}\mathbf{c} = \mathcal{F}^{-1}\mathbf{D}_T\,\mathcal{F}\mathbf{c} \qquad \mathcal{O}(N\log N) \text{ operations}$$

$\mathcal{F}$ is the fast Fourier transform (FFT)

# The Methods Considered

I - If the solution is **not smooth**: methods which involve **products** $\mathbf{H}\,\mathbf{c}$

**Polynomial approximations** to the action of the exponential on a vector

$$\mathbf{c}^{n+1} = e^{-ih\mathbf{H}}\mathbf{c}^{n}$$

1 – **Taylor**

2 – **Chebyshev**

3 – **Splitting (real + imaginary parts)**

II - If the solution is **smooth**: **to split** $\mathbf{H=T+V}$

**High-order Splitting-decomposition methods (with possible complex coefficients)**

$$\mathrm{e}^{-ia_1hT}\,\mathrm{e}^{-ib_1hV}\,\ldots\,\mathrm{e}^{-ia_shT}\,\mathrm{e}^{-ib_shV}$$

# The Methods Considered

I - If the solution is **not smooth**: methods which involve **products** $\mathbf{H}\,\mathbf{c}$

**Polynomial approximations** to the action of the exponential on a vector

$$\mathbf{c}^{n+1} = e^{-ih\mathbf{H}}\mathbf{c}^{n}$$

1 – **Taylor**

2 – **Chebyshev**

3 – **Splitting (real + imaginary parts)**

# The Methods Considered

I - If the solution is **not smooth**: methods which involve **products** $\mathbf{H}\,\mathbf{c}$
**Polynomial approximations** to the action of the exponential on a vector

$$\mathbf{c}^{n+1} = e^{-ih\mathbf{H}}\mathbf{c}^{n}$$

1 – **Taylor**

2 – **Chebyshev**

3 – **Splitting (real + imaginary parts)**

If we consider $\mathbf{c} = \mathbf{q} + i\mathbf{p}$ the following problems are equivalent

$$i\frac{d}{dt}\mathbf{c}(t) = \mathbf{H}\mathbf{c}(t) \quad \Rightarrow \quad \mathbf{c}(t) = e^{-it\mathbf{H}}\mathbf{c}(0)$$

$$\frac{d}{dt}\begin{Bmatrix} \mathbf{q} \\ \mathbf{p} \end{Bmatrix} = \begin{pmatrix} 0 & \mathbf{H} \\ -\mathbf{H} & 0 \end{pmatrix}\begin{Bmatrix} \mathbf{q} \\ \mathbf{p} \end{Bmatrix} \qquad \mathbf{O}(t) = \begin{pmatrix} \cos(t\mathbf{H}) & \sin(t\mathbf{H}) \\ -\sin(t\mathbf{H}) & \cos(t\mathbf{H}) \end{pmatrix}$$

or, in short: $z' = M\,z$ with $z = (\mathbf{q},\mathbf{p})^{\mathrm{T}}$

# The Taylor Method

An $m$-stage Taylor method to approximate $z(t_n + \tau) = e^{\tau M} z(t_n)$

$$z_{n+1} = P_m^T(\tau M) z_n$$

where $P_m^T(\tau M)$ is the Taylor expansion of the exponential, which approximates the exact solution up to order $m$

$$P_m^T(\tau M) \equiv \sum_{j=0}^{m} \frac{1}{j!} (\tau M)^j = e^{\tau M} + \mathcal{O}(\tau^{m+1})$$

# The Taylor Method

An $m$-stage Taylor method to approximate $z(t_n + \tau) = e^{\tau M} z(t_n)$

$$z_{n+1} = P_m^T(\tau M) z_n$$

where $P_m^T(\tau M)$ is the Taylor expansion of the exponential, which approximates the exact solution up to order $m$

$$P_m^T(\tau M) \equiv \sum_{j=0}^{m} \frac{1}{j!} (\tau M)^j = e^{\tau M} + \mathcal{O}(\tau^{m+1})$$

We can advance each step by using the **Horner's algorithm**

$$\begin{aligned}
&y_0 = z_n \\
&\textbf{do} \quad i = 1, m \\
&\quad y_i = z_n + \frac{1}{m+1-i} \tau M y_{i-1} \\
&\textbf{enddo} \\
&z_{n+1} = y_m
\end{aligned}$$

This algorithm can be trivially rewritten in terms of the real vectors, and it only requires to store two extra complex vectors (**four real vectors**).

# The Taylor Method

The matrix $P_m^T(\tau M)$ that propagates the numerical solution can be written as

$$P_m^T(\tau H) = \begin{pmatrix} T_1(\tau H) & T_2(\tau H) \\ -T_2(\tau H) & T_1(\tau H) \end{pmatrix}$$

where the entries $T_1(y)$ and $T_2(y)$ are the Taylor series expansion of $cos(y)$ and $sin(y)$ up to order $m$, i.e.

$$P_1^T(x) = \begin{pmatrix} 1 & x \\ -x & 1 \end{pmatrix}, \qquad P_2^T(x) = \begin{pmatrix} 1 - \frac{x^2}{2} & x \\ -x & 1 - \frac{x^2}{2} \end{pmatrix}$$

Notice that $\det P_m^T(y) = T_1(y)^2 + T_2(y)^2 \neq 1$

The eigenvalues are given by

$$\lambda_{\pm}^T = T_1 \pm iT_2$$

The scheme is stable if $T_1^2(y) + T_2^2(y) \leq 1$

For practical purposes, we require however

$$T_1^2(y) + T_2^2(y) \leq 1 + tol$$

# The Chebyshev Method

The Chebyshev method approximates the action of the exponential on the initial conditions by a near-optimal polynomial given by:

$$u(t) = \mathrm{e}^{-itH}\, u_0 \approx P_m^C(tH)\, u_0$$

where

$$P_{m-1}^C(tH)u_0 = c_0 u_0 + 2\sum_{k=1}^{m} c_k T_k\left(\frac{H}{\rho(H)}\right) u_0$$

with $c_k = (-i)^k J_k(t\,\rho(H))$. Here, $T_k(x)$ is the $k$th Chebyshev polynomial generated from the recursion

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x), \qquad k \geq 1$$

and $T_0(x)$=1, $T_1(x)$=$x$. $J_k(w)$ are the Bessel functions of the first kind which provides a **superlinear convergence** for $m > t\,\rho(H)$ i.e.

$$\frac{t\rho(H)}{m} \leq 1$$

# The Chebyshev Method

The **Clenshaw algorithm** allows to compute the action of the polynomial by storing only three vectors

$$
\begin{aligned}
& d_{m+1} = 0, \qquad d_m = 0 \\
& \textbf{do} \quad i = m-1, 0 \\
& \quad d_i = c_i z_n + \frac{2}{\rho(H)} H d_{i+1} - d_{i+2} \\
& \textbf{enddo} \\
& z_{n+1} = d_0 - d_2
\end{aligned}
$$

which can also be easily rewritten in term of the real vectors and it only requires **to store six real vectors**. The scheme can be written as

$$
P_m^C(\tau H) = \begin{pmatrix} C_1(\tau H) & C_2(\tau H) \\ -C_2(\tau H) & C_1(\tau H) \end{pmatrix}
$$

As in the Taylor case: $\det P_m^C(y) = C_1(y)^2 + C_2(y)^2 \neq 1$

It is not a symplectic transformation.

**Taylor**

| order | tol=inf | tol=$10^{-8}$ | tol=$10^{-4}$ |
|---|---|---|---|
| 10. | 0. | 0. | 0. |
| 15. | 0.111249 | 0.111249 | 0.111249 |
| 20. | 0.164515 | 0.164515 | 0.164515 |
| 25. | 0. | 0.065246 | 0.065246 |
| 30. | 0. | 0.108088 | 0.108088 |
| 35. | 0.0461259 | 0.138361 | 0.322661 |
| 40. | 0.0804521 | 0.160884 | 0.321618 |
| 45. | 0. | 0.178294 | 0.320804 |
| 50. | 0. | 0.192154 | 0.32015 |

**Chebyshev**

| order | tol=inf | tol=$10^{-8}$ | tol=$10^{-4}$ |
|---|---|---|---|
| 20. | 0.00362818 | 0.321723 | 0.643217 |
| 25. | 0.00233368 | 0.384289 | 0.64029 |
| 30. | 0.00162599 | 0.425648 | 0.638324 |
| 35. | 0.00119743 | 0.45502 | 0.636912 |
| 40. | 0.000918402 | 0.476957 | 0.63585 |
| 45. | 0.000726646 | 0.635021 | 0.635021 |
| 50. | 0.000589228 | 0.634356 | 0.634356 |
| 55. | 0. | 0.633812 | 0.633812 |
| 60. | 0. | 0.633357 | 0.633357 |

# The Symplectic Splitting Method

Taylor and Chebyshev methods consider vector-matrix products to approximate

$$\mathbf{c}^{n+1} = e^{-ih\mathbf{H}}\mathbf{c}^n$$

A first order approximation

$$\mathbf{c}^{n+1} = \mathbf{c}^n - ih\mathbf{H}\mathbf{c}^n$$

can be written with $\mathbf{c} = \mathbf{q} + i\mathbf{p}$

$$\mathbf{q}^{n+1} + i\mathbf{p}^{n+1} = \mathbf{q}^n + i\mathbf{p}^n - ih\mathbf{H}(\mathbf{q}^n + i\mathbf{p}^n)$$

or

$$\begin{aligned}\mathbf{q}^{n+1} &= \mathbf{q}^n + h\mathbf{H}\mathbf{p}^n \\ \mathbf{p}^{n+1} &= \mathbf{p}^n - h\mathbf{H}\mathbf{q}^n\end{aligned}$$

# The Symplectic Splitting Method

The linear time dependent SE

$$i\frac{d}{dt}\mathbf{c}(t) = \mathbf{H}\mathbf{c}(t) \quad \Rightarrow \quad \mathbf{c}(t) = e^{-it\mathbf{H}}\mathbf{c}(0)$$

with $\mathbf{H}$ real and can be reformulated using real variables as the Hamiltonian system:

$$\mathcal{H} = \tfrac{1}{2}\mathbf{p}^T\mathbf{H}\mathbf{p} + \tfrac{1}{2}\mathbf{q}^T\mathbf{H}\mathbf{q}$$

$$\frac{d}{dt}\left\{ \begin{array}{c} \mathbf{q} \\ \mathbf{p} \end{array} \right\} = \left( \begin{array}{cc} 0 & \mathbf{H} \\ -\mathbf{H} & 0 \end{array} \right) \left\{ \begin{array}{c} \mathbf{q} \\ \mathbf{p} \end{array} \right\}$$

with formal solution: $\mathbf{O}(t) = \left( \begin{array}{cc} \cos(t\mathbf{H}) & \sin(t\mathbf{H}) \\ -\sin(t\mathbf{H}) & \cos(t\mathbf{H}) \end{array} \right)$

# The Symplectic Splitting Method

We have built splitting methods for the harmonic oscillator!!!

$$\frac{d}{dt}\left\{\begin{array}{c} q \\ p \end{array}\right\} = \Big[\underbrace{\left(\begin{array}{cc} 0 & 1 \\ 0 & 0 \end{array}\right)}_{A} + \underbrace{\left(\begin{array}{cc} 0 & 0 \\ -1 & 0 \end{array}\right)}_{B}\Big]\left\{\begin{array}{c} q \\ p \end{array}\right\}$$

Splitting method

$$K(h) \equiv \prod_{i=1}^{m} e^{h a_i A}\, e^{h b_i B}$$

We have

$$K(h) \equiv \prod_{i=1}^{m} \left(\begin{array}{cc} 1 - a_i b_i h^2 & a_i h \\ -b_i h & 1 \end{array}\right) = \left(\begin{array}{cc} K_1 & K_2 \\ K_3 & K_4 \end{array}\right)$$

$$K_1 = \sum_{i=0}^{m} k_{1,i} h^{2i} \qquad K_2 = \sum_{i=1}^{m} k_{2,i} h^{2i-1}$$

$$K_3 = \sum_{i=1}^{m} k_{3,i} h^{2i-1} \qquad K_4 = \sum_{i=0}^{m} k_{4,i} h^{2i}$$

which are polynomials of **degree twice higher** as in the previous cases for the same number of stages.

# The Symplectic Splitting Method

The algorithm: a **generalisation** of the **Horner's** algorithm or the **Clenshav** algorithm

$$
\begin{aligned}
\textbf{do} \quad & i = 1, m \\
& v = Hp \\
& q := q + a_i \tau v \\
& v = Hq \\
& p := p - b_i \tau v \\
\textbf{enddo} \quad &
\end{aligned}
$$

It only requires to store one additional real vector of dimension

The methods preserve symplecticity by construction: $\det K(h) = 1$

Stability: *M* is stable if $|\mathrm{Tr}\ K|<2$, i.e.

$$|K_1 + K_4| = \left| \sum_{i=0}^{m} \left(k_{1,i} + k_{4,i}\right) h^{2i} \right| < 2$$

# The Symplectic Splitting Method

**Theorem**: Any composition method is conjugate to an orthogonal method, and unitarity is preserved up to conjugacy.

There is a recursive procedure to get the coefficients of the splitting methods from the coefficients of the matrix $K$.

We can build different matrices with:

- **Large stability domain**

- **Accurate approximation** to the solution in the whole interval (like Chebyshev)

- Methods with **different orders of accuracy** and very large number of satges.

## Splitting Methods: valid for $\dfrac{t\rho(H)}{m} \leq \theta'$

| $m$ | $r$ | $\theta'$ | $y_*/m$ | $\sum_j(\lvert a_j\rvert + \lvert b_j\rvert)$ | $\mu_r(\theta' m)$ | $\nu_r(\theta' m)$ |
|---|---|---|---|---|---|---|
| 10 | 6 | 1 | 1.1586 | 3.895 | 0.001412 | 0.01245 |
| 20 | 16 | 1 | 1.0456 | 3.0553 | 0.000611028 | 0.0258433 |
| 30 | 24 | 1 | 1.0246 | 3.19658 | 0.0000841871 | 0.0373544 |
| 30 | 6 | 1.4 | 1.41876 | 3.0921 | 0.0000518519 | 0.0131295 |
| 30 | 0 | 1 | 1.1411 | 3.04948 | $2.91902 \cdot 10^{-13}$ | $2.28673 \cdot 10^{-9}$ |
| 30 | 0 | 0.75 | 1.027 | 3.44381 | $1.2545 \cdot 10^{-17}$ | $5.96706 \cdot 10^{-14}$ |
| 30 | 0 | 0.5 | 0.937874 | 3.84442 | $7.96031 \cdot 10^{-24}$ | $6.66693 \cdot 10^{-18}$ |
| 40 | 0 | 1 | 1.15953 | 3.21986 | $1.06301 \cdot 10^{-15}$ | $1.07587 \cdot 10^{-12}$ |

**Error bounds:** $$\|u_n - u(t)\| \leq \frac{t\,\mu_r(\theta)\,\|u_0\|_{r+1} + \nu_r(\theta)\|u_0\|_r}{\theta^r}\,\tau^r$$

$$\theta = m\theta' \qquad \|u\|_k := \|H^k u\|$$

## *Schrödinger equation with a Poschl-Teller potential*

$$i\frac{\partial}{\partial t}\psi(x,t) = \left(-\frac{1}{2\mu}\frac{\partial^2}{\partial x^2} + V(x)\right)\psi(x,t)$$

with

$$V(x) = -\frac{\alpha^2}{2\mu}\frac{\lambda(\lambda-1)}{\cosh^2(\alpha x)}$$

$$\mu = 1745, \quad \alpha = 2, \quad \lambda = 24{,}5,$$

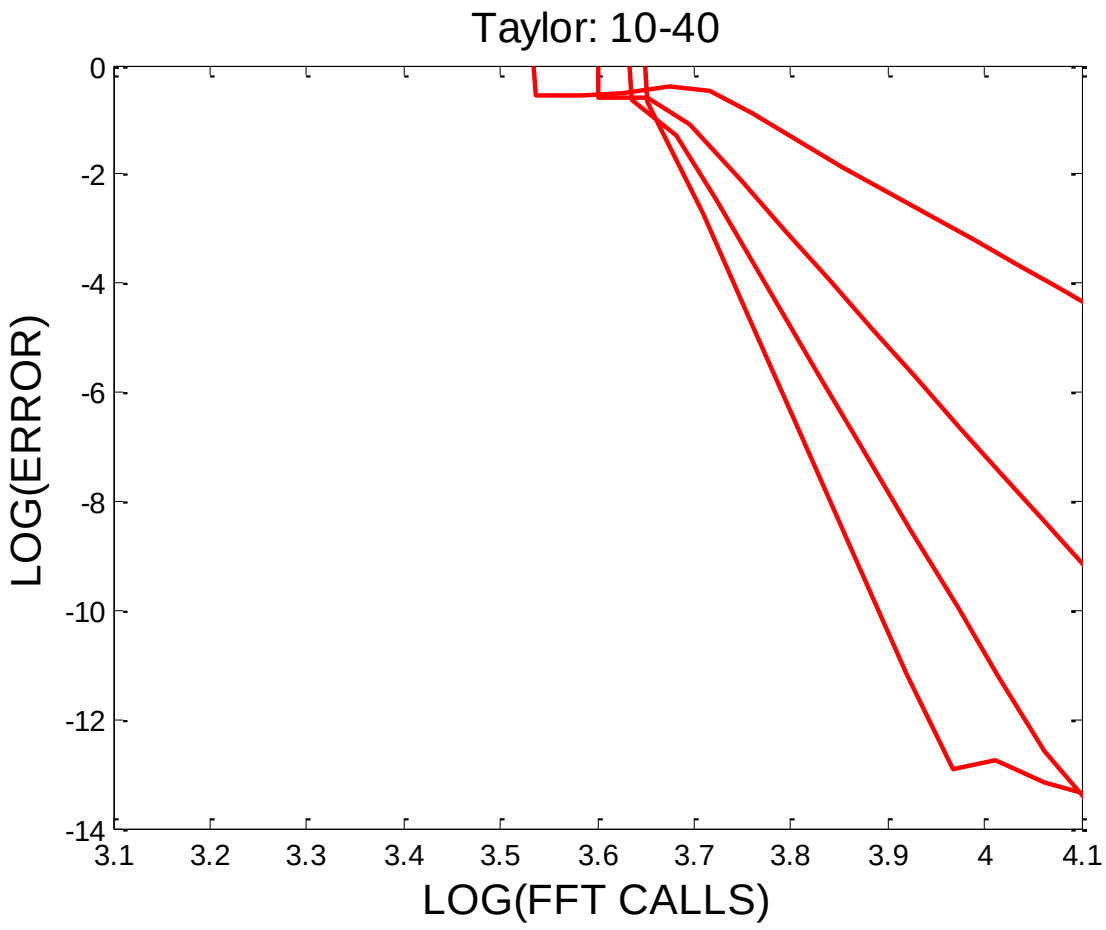
Initial conditions

$$\psi(x,0) = \rho e^{-9^2 x^2}$$

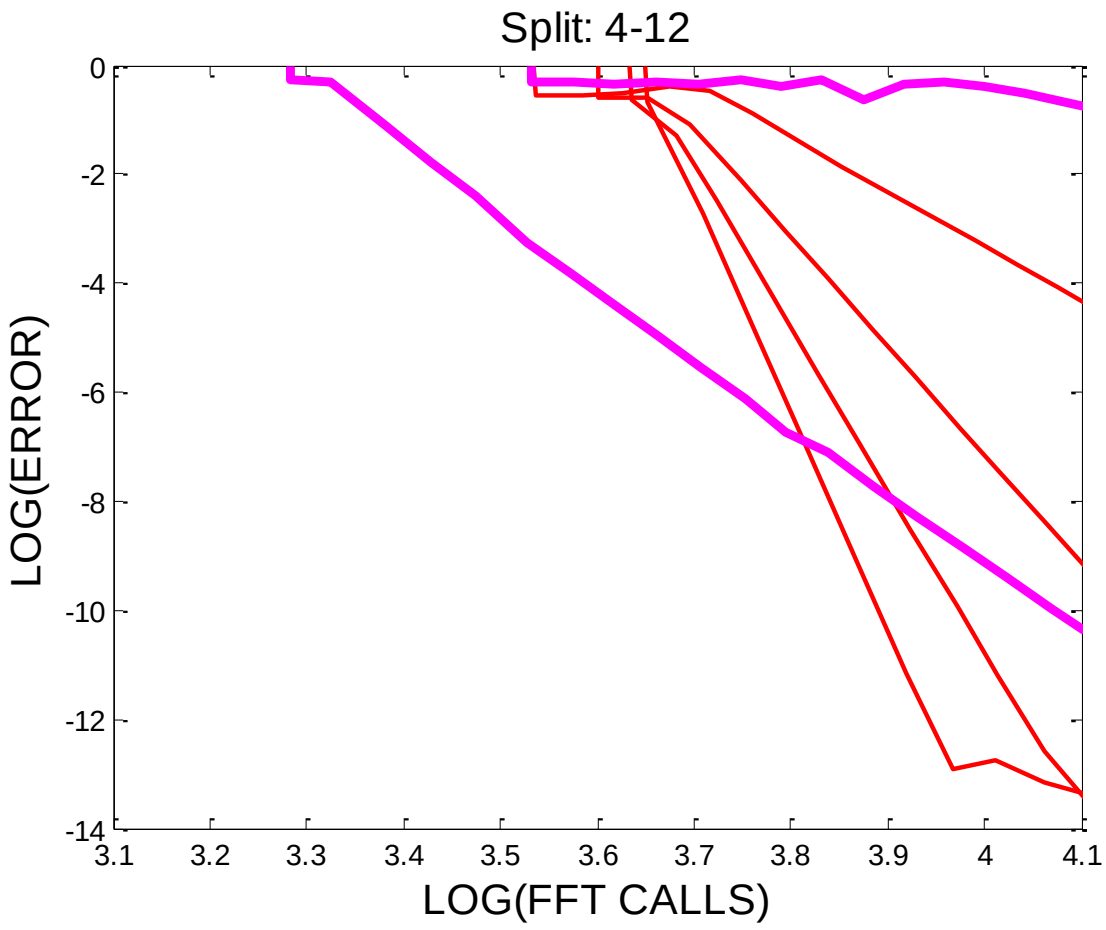
$$t \in [0, 2T] \quad x \in [-5, 5], \quad N = 128 \text{ parts}$$

$$\rho(H) \simeq \frac{1}{2\mu}\left(\frac{\pi}{\Delta x}\right)^2 + (V_{max} - V_{min}) = 1.12$$

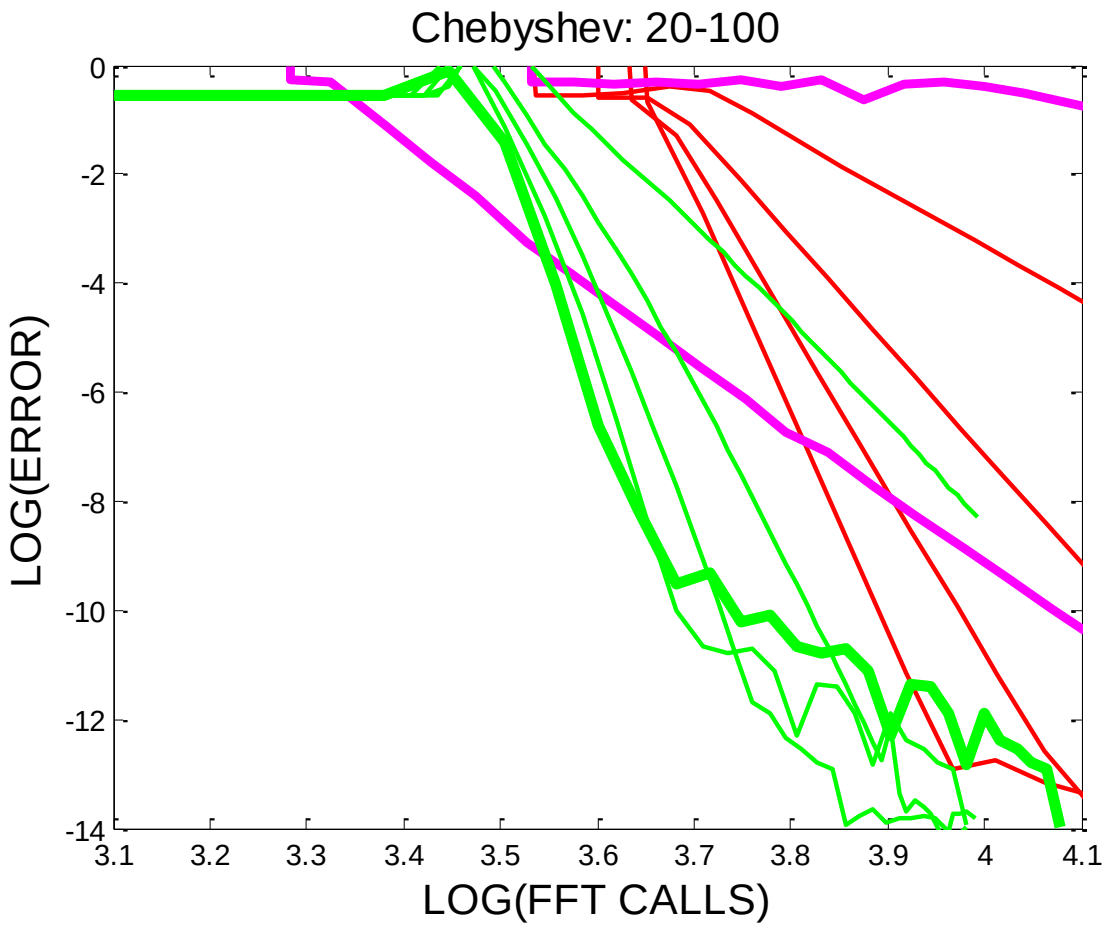
## *Poschl-Teller potential*



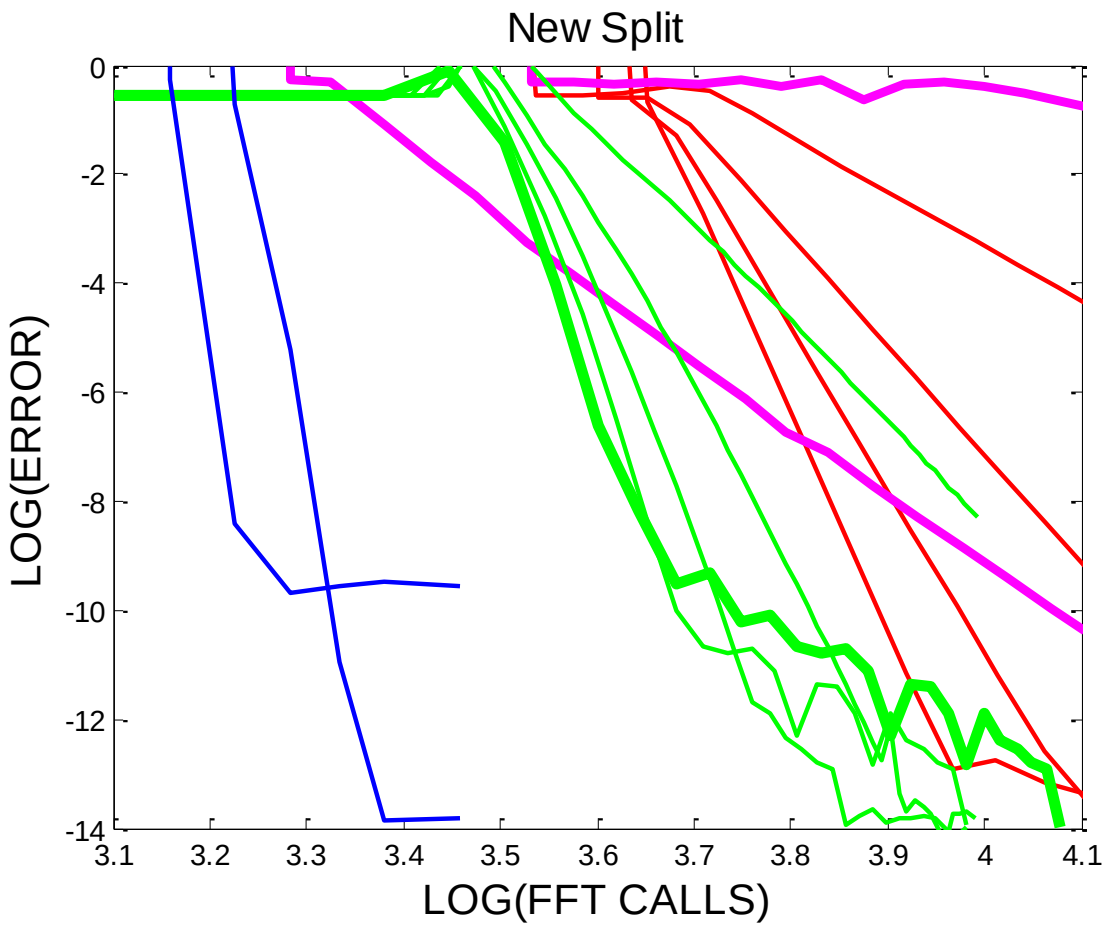
## *Poschl-Teller potential*



## *Poschl-Teller potential*



## *Poschl-Teller potential*



# References


SB, F. Casas, and A. Murua, **Work in Progress.**

Group webpage: **http://www.gicas.uji.es**

SB, F. Casas and A. Murua, **Symplectic splitting operator methods for the time-dependent Schrödinger equation**, J. Chem. Phys. 124 (2006) 234105.

SB, F. Casas and A. Murua, **On the linear stabylity of splitting methods**, Found. Comp. Math., 8 (2008), 357-393.

SB, F. Casas and A. Murua, **Error analysis of splitting methods for the time-dependent Schrödinger equation**, Submitted.

# The Schrödinger equation

$$i\frac{\partial}{\partial t}\psi(x,t) = \left(-\frac{1}{2\mu}\nabla^2 + V(x,t)\right)\psi(x,t)$$

II - If the solution is **smooth**: **to split** **H=T+V**

**High-order Splitting methods**

$$\mathrm{e}^{-ia_1 hT}\,\mathrm{e}^{-ib_1 hV}\,\ldots\,\mathrm{e}^{-ia_s hT}\,\mathrm{e}^{-ib_s hV}$$

but the computation using FFTs is:

$$\mathcal{F}^{-1}\mathrm{e}^{-ia_1 hD_T}\mathcal{F}\,\mathrm{e}^{-ib_1 hV}\,\ldots\,\mathcal{F}^{-1}\mathrm{e}^{-ia_s hD_T}\mathcal{F}\,\mathrm{e}^{-ib_s hV}$$

The computational cost is independent if the coeficients are real or complex but, the coefficnets $a_i$ must be real because $D_T$ is an unbounded operator.

## Splitting methods with complex coefficients

Given a **symmetric** method of order $2p$, $\mathcal{S}^{[2p]}(h)$, we can define a recursion by **symmetric** compositions

$$\mathcal{S}^{[2p+2]}(h) = \prod_{i=1}^{m_p} \mathcal{S}^{[2p]}(\alpha_{p,i} h)$$

$$\sum_{i=1}^{m_p} \alpha_{p,i} = 1 \qquad \text{and} \qquad \sum_{i=1}^{m_p} \alpha_{p,i}^{2p+1} = 0.$$

Starting from $\mathcal{S}^{[2]}(h)$, we have

$$\mathcal{S}^{[2(p+1)]}(h) = \prod_{i_p=1}^{m_p} \left( \prod_{i_{p-1}=1}^{m_{p-1}} \left( \cdots \left( \prod_{i_1=1}^{m_1} \mathcal{S}^{[2]}(\alpha_{p,i_p} \alpha_{p-1,i_{p-1}} \cdots \alpha_{1,i_1} h) \right) \cdots \right) \right)$$

Castella, Chartier, Descombes, & Vilmart, BIT 49 (2009), 487-508, and Hansen & Ostermann, BIT 49 (2009), 527-542, obtained methods up to order 14 with coefs. having positive real part.

$$\mathcal{S}_h^{[2]} \to \mathcal{S}_h^{[4]} \to \mathcal{S}_h^{[6]} \to \mathcal{S}_h^{[8]} \to \ldots \mathcal{S}_h^{[14]} \to \mathcal{S}_h^{[16]}$$

## Splitting methods with complex coefficients

$$\frac{\partial}{\partial t}\mathbf{u} = \delta \triangle \mathbf{u} + F(t, \mathbf{u})$$

(a) The linear heat equation with potential

$$\frac{\partial}{\partial t}\mathbf{u} = \triangle \mathbf{u} + V(x)\mathbf{u}$$

(b) The semi-linear complex Ginzburg–Landau equation:

$$\frac{\partial \mathbf{u}}{\partial t} = \alpha \, \Delta \mathbf{u} + \varepsilon \, \mathbf{u} - \beta \, |\mathbf{u}|^2 \mathbf{u},$$

with $\alpha = 1 + ic_1$, $\beta = 1 - ic_3$ and $\varepsilon, c_1, c_3 \in \mathbb{R}$ .

## Splitting methods with complex coefficients

$$\frac{\partial}{\partial t}\mathbf{u} = \delta \triangle \mathbf{u} + F(t, \mathbf{u})$$

(a) The linear heat equation with potential

$$\frac{\partial}{\partial t}\mathbf{u} = \triangle \mathbf{u} + V(x)\mathbf{u}$$

(b) The semi-linear complex Ginzburg–Landau equation:

$$\frac{\partial \mathbf{u}}{\partial t} = \alpha\, \Delta \mathbf{u} + \varepsilon\, \mathbf{u} - \beta\, |\mathbf{u}|^2 \mathbf{u},$$

with $\alpha = 1 + ic_1$, $\beta = 1 - ic_3$ and $\varepsilon, c_1, c_3 \in \mathbb{R}$ .

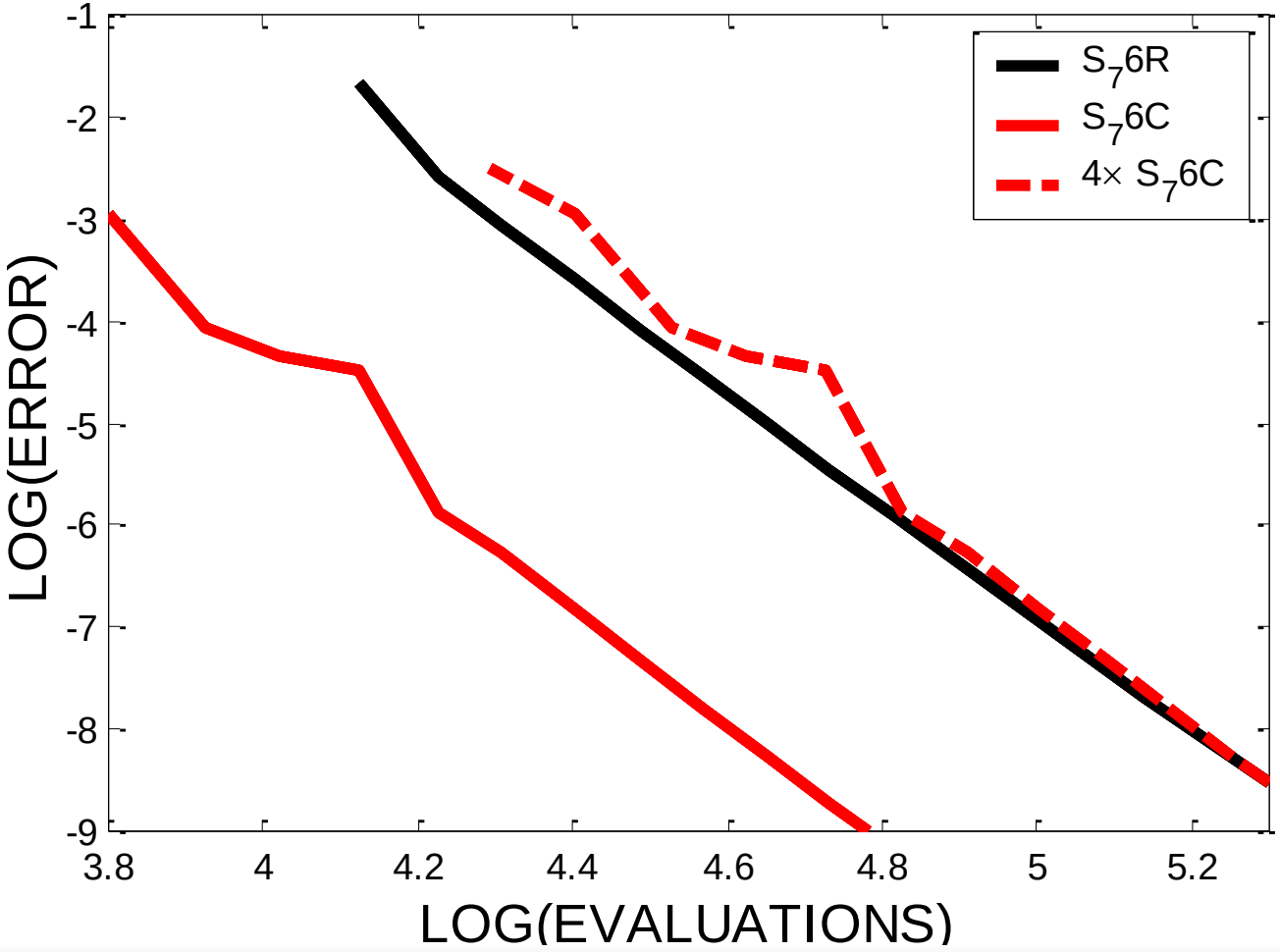
$$\begin{aligned} \mathcal{S}^{[6]}(h) &= \mathcal{S}^{[2]}(\gamma_1 h) \cdots \mathcal{S}^{[2]}(\gamma_7 h) \\ \mathcal{S}^{[8]}(h) &= \mathcal{S}^{[2]}(\gamma_1 h) \cdots \mathcal{S}^{[2]}(\gamma_{15} h) \end{aligned}$$

SB, Casas, Chartier, and Murua, **Splitting methods with complex coefficients for some classes of evolution equations.** Submitted. arXiv:1102.1622v1

## A simple example: the Lotka-Volterra problem

$$\dot{u} = u(v-2), \qquad \dot{v} = v(1-u)$$

$$I(u,v) = \ln(uv^2) - (u+v).$$



## Splitting for perturbed systems

$$i\frac{\partial}{\partial t}\psi(x,t) = \left(-\frac{1}{2\mu}\nabla^2 + V(x,t)\right)\psi(x,t)$$

Additional benefits of the unitary splitting: It allows different ways to splitt depending on the structure of the problem

$$H = \frac{1}{2}(p^2 + x^2) + \varepsilon V_I(x,t) \qquad p = -i\frac{\partial}{\partial x}$$

## Splitting for perturbed systems

$$i\frac{\partial}{\partial t}\psi(x,t) = \left(-\frac{1}{2\mu}\nabla^2 + V(x,t)\right)\psi(x,t)$$

Additional benefits of the unitary splitting: It allows different ways to splitt depending on the structure of the problem

$$H = \frac{1}{2}(p^2 + x^2) + \varepsilon V_I(x,t) \qquad p = -i\frac{\partial}{\partial x}$$

$$e^{-it\frac{1}{2}(p^2+x^2)} = e^{-if(t)\frac{1}{2}x^2}\ e^{-ig(t)\frac{1}{2}p^2}\ e^{-if(t)\frac{1}{2}x^2}$$

$$g(t) = \sin(t), \qquad f(t) = (1-\cos(t))/\sin(t) \qquad |t| < \pi:$$

## Splitting for perturbed systems

$$i\frac{\partial}{\partial t}\psi(x,t) = \left(-\frac{1}{2\mu}\nabla^2 + V(x,t)\right)\psi(x,t)$$

Additional benefits of the unitary splitting: It allows different ways to splitt depending on the structure of the problem

$$H = \frac{1}{2}(p^2 + x^2) + \varepsilon V_I(x,t) \qquad p = -i\frac{\partial}{\partial x}$$

$$e^{-it\frac{1}{2}(p^2+x^2)} = e^{-if(t)\frac{1}{2}x^2}\; e^{-ig(t)\frac{1}{2}p^2}\; e^{-if(t)\frac{1}{2}x^2}$$

$$g(t) = \sin(t), \qquad f(t) = (1-\cos(t))/\sin(t) \qquad |t| < \pi:$$

This result can be generalised to the explicitly time-dependent case:

$$H = \frac{1}{2}(p^2 + w(t)x^2) + \varepsilon V_I(x,t)$$

P. Bader and SB, **Fourier methods for the perturbed harmonic oscillator in linear and nonlinear Schrödinger equations.** **Phys Rev. E.** In press.

# Conclusions

- Splitting methods are powerful tools for numerically solving the Schrödinger equation

- Some a priori knowledge of the problem and its solution is essential for an efficient integration

- The performance strongly depends on how the system has been split as well as on the choice of the appropriate method for each problem

- Splitting methods with complex coefficients could lead to very efficient methods for many problems