Applied Numerical Mathematics 178 (2022) 329–336

Contents lists available at ScienceDirect

Applied Numerical Mathematics

www.elsevier.com/locate/apnum

Optimized integrating factor technique for Schrödinger-like equations

M. Lovisetto *, D. Clamond, B. Marcos

Université Côte d'Azur, CNRS UMR 7351, Laboratoire J. A. Dieudonné, Parc Valrose, 06108 Nice cedex 2, France

ARTICLE INFO

Article history:
Received 18 January 2022
Received in revised form 31 March 2022
Accepted 12 April 2022
Available online 15 April 2022

Keywords:
Gauge optimization
Time-stepping
Integrating factor
Schrödinger-like equations
Lawson methods

ABSTRACT

The integrating factor technique is widely used to solve numerically (in particular) the Schrödinger equation in the context of spectral methods. Here, we present an improvement of this method exploiting the freedom provided by the gauge condition of the potential. Optimal gauge conditions are derived considering the equation and the temporal numerical resolution with an adaptive embedded scheme of arbitrary order. We illustrate this approach with the nonlinear Schrödinger (NLS) and with the Schrödinger–Newton (SN) equations. We show that this optimization increases significantly the overall computational speed, sometimes by a factor five or more. This gain is crucial for long time simulations, as, with larger time steps, less computations are performed and the overall accumulation of round-off errors is reduced.

1. Introduction

The Schrödinger equation is used in many fields of Physics. In dimensionless units, it takes the form

$$\mathrm{i}\,\partial_t\,\psi \;+\; \tfrac{1}{2}\,\nabla^2\psi \;-\; V\,\psi \;=\; 0, \tag{1}$$

where ψ is a function of the spatial coordinates $\boldsymbol{r}$ and of the time t, ∇^2 is the Laplace operator, and the potential V is generally a function of space and time and, possibly, a functional of ψ. In this paper, we focus more specifically on the nonlinear Schrödinger equation (V proportional to $|\psi|^2$) and on the Schrödinger–Newton (or Schrödinger–Poisson) equation in which the Laplacian of the potential is proportional to $|\psi|^2$. These special cases were chosen for clarity and because they are of practical interest, but the method presented here can be extended to more general potentials (and equations).

With the nonlinear Schrödinger equation (NLS) considered here, the (local nonlinear) potential is

$$V \;=\; g\,|\psi|^2, \tag{2}$$

where g is a coupling constant. For $g > 0$ the interaction is repulsive, while, for $g < 0$ it is attractive. The NLS describes various physical phenomena, such as Bose–Einstein condensates [10], laser beams in some nonlinear media [15], water wave packets [18], etc.

With the Schrödinger–Poisson equation, the potential is given by the Poisson equation

$$\nabla^2\,V \;=\; g\,|\psi|^2, \tag{3}$$

[*] Corresponding author.
E-mail address: Martino.LOVISETTO@univ-cotedazur.fr (M. Lovisetto).

https://doi.org/10.1016/j.apnum.2022.04.011
0168-9274/© 2022 IMACS. Published by Elsevier B.V. All rights reserved.

where g is another coupling constant, the interaction being attractive if $g > 0$ and repulsive if $g < 0$. It is therefore *nonlinear* and *non-local*, giving rise to collective phenomena [5], appearing for instance in optics [9,24,25], Bose–Einstein condensates [14], cosmology [17,21,22] and theories describing the quantum collapse of the wave function [11,23]. It is also used as a model to perform cosmological simulations in the semi-classical limit [29].

The above equations cannot be solved analytically (except for very special cases) and numerical methods must be employed. In this paper, we focus on spectral methods for the spatial resolution, i.e., methods that are based on fast Fourier transform (FFT) techniques, that are specially efficient and accurate [8]. For the temporal resolution, two families of methods are commonly employed to solve Schrödinger-like equations: integrating factors [20] and split-step integrators [6]. The latter methods have been used to integrate both the SN and NLS equations, but the former is used essentially to solve the NLS, with very performing results [2–4]. In this note, we focus on the former technique, which consists in integrating analytically the linear part of the equation and integrating numerically the remaining nonlinear part with a classical method [20]. The principle of the method is described as follows.

Writing the Schrödinger equation in the generic form

$$\mathrm{i}\,\partial_t\,\psi\;=\;F(\boldsymbol{r},t,\psi)\,,\qquad \psi\;=\;\psi(\boldsymbol{r},t)\,, \tag{4}$$

the right-hand side is split into linear and nonlinear parts

$$\mathrm{i}\,\partial_t\,\psi\;+\;\mathcal{L}\,\psi\;=\;\mathcal{N}(\boldsymbol{r},t,\psi)\,, \tag{5}$$

where $\mathcal{L}$ is an easily computable autonomous linear operator and $\mathcal{N}\stackrel{\text{def}}{=}F+\mathcal{L}\psi$ is the remaining (usually) nonlinear part. At the n-th time-step, with $t\in[t_n,t_{n+1}]$, considering the change of dependent variable

$$\phi\;\stackrel{\text{def}}{=}\;\exp[\,(t-t_n)\,\mathcal{L}\,]\,\psi\qquad\Longrightarrow\qquad \mathrm{i}\,\partial_t\,\psi\;=\;\exp[\,(t_n-t)\,\mathcal{L}\,]\,(\,\mathrm{i}\,\partial_t\,\phi\;-\;\mathcal{L}\,\phi\,)\,, \tag{6}$$

so $\phi=\psi$ at $t=t_n$, the equation (5) is rewritten

$$\mathrm{i}\,\partial_t\,\phi\;=\;\exp[\,(t-t_n)\,\mathcal{L}\,]\,\mathcal{N}. \tag{7}$$

The operator $\mathcal{L}$ being well chosen, the stiffness of (5) is considerably reduced and the equation (7) is (hopefully) well approximated by algebraic polynomials for $t\in[t_n;t_{n+1}]$. Thus, standard time-stepping methods, such as an adaptive Runge–Kutta method [1,7], can be used to efficiently solve (7). To do so, the solution is evaluated at two different orders and a local error is estimated as the difference between those quantities. Popular integrators can be found in [1,12].

Notice that, for finite domains, the integrating factor must be implemented such that boundary conditions and the domain geometry are properly taken into account. However, here, we focus on periodic domains, therefore there are no conditioning issues with the method, as explained in [16].

It is straightforward to apply this strategy to the Schrödinger equation (1) since $\frac{1}{2}\nabla^2\psi$ and the potential V are, respectively, linear and nonlinear operators of ψ. By switching to Fourier space in position, the equation becomes

$$\mathrm{i}\,\partial_t\,\widehat{\psi}\;-\;\tfrac{1}{2}\,k^2\,\widehat{\psi}\;-\;\widehat{V\,\psi}\;=\;0, \tag{8}$$

where "hats" denote the Fourier transform of the underneath quantity and $k\stackrel{\text{def}}{=}|\boldsymbol{k}|$ ($\boldsymbol{k}$ the wave vector). The equation is now in a form where the application of the integrating factor (IF) method is straightforward, i.e., (8) becomes

$$\mathrm{i}\,\partial_t\,\phi\;=\;-\mathrm{i}\,\exp[\tfrac{\mathrm{i}}{2}k^2(t-t_n)]\,\widehat{V\,\psi}, \tag{9}$$

where $\phi(\boldsymbol{k},t)\stackrel{\text{def}}{=}\widehat{\psi}(\boldsymbol{k},t)\exp[\frac{\mathrm{i}}{2}k^2(t-t_n)]$. If the nonlinear part of the equation is zero, then $\mathrm{i}\,\partial_t\,\phi=0$ and any (reasonable) temporal scheme will produce the exact solution $\phi(t)=\phi(t_n)$. In other words, the integrating factor technique is exact for linear equations. This indicates that the numerical errors depend on the magnitude of the nonlinear part. Therefore, in order to minimise these errors, a strategy consists in minimizing the magnitude of $\mathcal{N}$ at each time-step. To do so, we exploit the gauge invariance of the Schrödinger equation: if ψ is a solution of (1) at a given time t, then $\Psi=\psi(\mathbf{r},t)\mathrm{e}^{-\mathrm{i}\int_{t_0}^{t}\mathrm{d}s\,\mathcal{C}(s)}$, with $\mathcal{C}(t_0)=0$, is a solution of

$$\mathrm{i}\,\partial_t\,\Psi\;+\;\tfrac{1}{2}\,\nabla^2\Psi\;-\;(V+\mathcal{C}(t))\;\Psi\;=\;0, \tag{10}$$

as one can easily verify. Thus, at each time-step, adding a constant $\mathcal{C}_n$ to V in (1), modifies the solution as

$$\psi(t_n)\;\to\;\psi(t_n)\,\mathrm{e}^{-\mathrm{i}\varphi},\qquad \varphi\;\stackrel{\text{def}}{=}\;\sum_{j=0}^{n}\mathcal{C}_j\,h_j, \tag{11}$$

where $h_j\stackrel{\text{def}}{=}t_{j+1}-t_j$ is the j-th time-step. Of course, at the end of the computations, the operation (11) can be easily reverted if the original phase is relevant. Using this procedure, we observed up to a five-fold speed increase (the overall computational time is divided by about five) compared to taking $\mathcal{C}_n = 0$. Of course, the speed-up varies depending on the initial condition, of the (spatial and temporal) numerical schemes and on the choice of gauge corresponding to $\mathcal{C}_n = 0$.

In this paper, we derive some analytic formulas giving an optimal $\mathcal{C}_n$ in order to maximise the time-step, i.e., to minimize the overall computational time of the numerical resolution. We emphasize that the thus obtained optimal values of $\mathcal{C}_n$ do not affect the accuracy of the numerical solution, leaving it unchanged with respect to the $\mathcal{C}_n = 0$ case. Two strategies are presented. In section 2, a first 'natural' approach to derive a suitable $\mathcal{C}_n$ is based on the analytical structure of the equation and it is independent of the numerical algorithm employed for its resolution. More precisely, $\mathcal{C}_n$ is obtained minimizing a norm of the right-hand side of the equation (5). This provides an easy manner to obtain a formula that is moreover computationally 'cheap'. This expression is however only near optimal, so a 'better' expression is subsequently derived. Considering both the equations and the numerical algorithms, a second optimal expression for $\mathcal{C}_n$ is derived in section 3. This approach consists in minimizing exactly the numerical error and thus explicitly dependents on the numerical scheme. This provides a more accurate, but computationally expensive, solution. The advances of these special choices are illustrated numerically in section 4. Finally, a summary and perspectives are drawn in section 5.

2. Near optimal $\mathcal{C}_n$

As mentioned above, if properly chosen, the integrating factor is able to reduce the stiffness of the equation, making the numerical integration more efficient. In addition, the magnitude of the nonlinear part of (7) also contributes to the efficiency of the numerical integration. Specifically, if $\mathcal{N}$ is zero, $\partial_t \phi = 0$ and the integrating factor technique is exact. Thus, the efficiency of the algorithm is expected to increase as the magnitude of $\mathcal{N}$ gets smaller, and subsequently the overall computational time should be reduced. Here, we show how to choose the arbitrary constant $\mathcal{C}_n$ in order to reduce the magnitude of the nonlinear part $\mathcal{N}$. In the case of the Schrödinger equation, we have

$$\mathcal{N}(\boldsymbol{k}, t; \phi; \mathcal{C}_n) \; = \; -\mathrm{i} \exp[\tfrac{\mathrm{i}}{2} k^2 (t - t_n)] \, \mathcal{F}\{(V + \mathcal{C}_n) \, \psi\}, \tag{12}$$

where $\mathcal{F}$ denotes the Fourier transform, and $\psi(\boldsymbol{x}, t) = \mathcal{F}^{-1}\{\exp[-\tfrac{\mathrm{i}}{2} k^2 (t - t_n)] \, \phi(\boldsymbol{k}, t)\}$.

A natural strategy is to minimise the L_2-norm, namely

$$\mathcal{G}_n(\mathcal{C}_n) \overset{\text{def}}{=} \frac{1}{M} \sum_{m=-[M/2]}^{[M/2]-1} |\, \mathcal{N}(\boldsymbol{k}_m, t_n; \phi; \mathcal{C}_n) \,|^2, \tag{13}$$

where M is the number of spatial modes, square brackets denote the integer part and $\boldsymbol{k}_m$ is the m-th Fourier mode. Indeed, in the latter expression, we considered discrete Fourier transforms, given that we are in the context of numerical simulations. The explicit expression of $\mathcal{G}_n$ can be found exploiting the definition of the discrete Fourier transform. For simplicity, we do the calculations in one dimension (1D) without loss of generality, since the final result is independent of the spatial dimension d. From Parseval theorem, one obtains

$$\mathcal{G}_n(\mathcal{C}_n) \; = \sum_{\ell=-[M/2]}^{[M/2]-1} (V_\ell + \mathcal{C}_n)^2 \, |\psi_\ell|^2, \tag{14}$$

where $\psi_\ell \overset{\text{def}}{=} \psi(\boldsymbol{x}_\ell)$ and $V_\ell \overset{\text{def}}{=} V(\boldsymbol{x}_\ell)$ at time t_n. Since the function $\mathcal{G}_n(\mathcal{C}_n)$ is a second-order polynomial in $\mathcal{C}_n$, it admits an unique minimum, which is obtained from the equation $\mathrm{d}G_n(\mathcal{C}_n)/\mathrm{d}\mathcal{C}_n = 0$, yielding

$$\mathcal{C}_n \; = -\left(\sum_{\ell=-[M/2]}^{[M/2]-1} V_\ell \, |\psi_\ell|^2 \right) \Bigg/ \left(\sum_{\ell=-[M/2]}^{[M/2]-1} |\psi_\ell|^2 \right) \overset{\text{def}}{=} \tilde{\mathcal{C}}_n. \tag{15}$$

Therefore, at each time step n, $\tilde{\mathcal{C}}_n$, which is the value of $\mathcal{C}_n$ minimizing the L_2-norm of $\mathcal{N}$, is obtained from (15). We show below that even though this approach is not unique (i.e., different norms could be considered), the provided solution is quite advantageous compared to others, being computationally cheap and independent on the order of the numerical scheme.

3. Optimal $\mathcal{C}_n$

We show here another way to choose the arbitrary constant $\mathcal{C}_n$ in order to improve the algorithm efficiency and reduce the overall computational time. This approach is based on the principles of the adaptive time-step procedure, where at each time step n, an error Δ_n between two approximated solutions of different orders is estimated. Since the smaller this quantity the larger the time-step, minimizing Δ_n allows to choose a larger time-step, speeding-up the numerical integration and keeping roughly the same numerical error. More specifically, the error Δ_n depends on the arbitrary constant $\mathcal{C}_n$, hence the minimization can be performed (see below) choosing an appropriate $\mathcal{C}_n$. In this section, we first recall the method for determining the size of the time step used in the Runge–Kutta procedures; interested readers should refer to [1] for further details. Although the determination of $\mathcal{C}_n$ can be formally presented for any embedded Runge–Kutta schemes, this results in very cumbersome calculations with little insights. Thus, for brevity and clarity, we illustrate the method with the Heun method (that is a second-order Runge–Kutta method with an embedded first-order explicit Euler scheme for the time stepping [27]). We then sketch-out how this procedure can be implemented for generic embedded Runge–Kutta methods.

3.1. Principle of the adaptive time-step procedure

For the time stepping, embedded Runge–Kutta methods estimate the quadrature error comparing the results of two orders of the time integrator [1]. For a solver of order N with an embedded $(N-1)$-order scheme (hereafter schemes of orders $\{N, N-1\}$), at the n-th time step, the error Δ_n is [28]

$$\Delta_n \stackrel{\text{def}}{=} \sqrt{\frac{1}{M}\sum_{m=-[M/2]}^{[M/2]-1}\left(\frac{\left|\,\phi(\boldsymbol{k}_m,t_n) - \widetilde{\phi}(\boldsymbol{k}_m,t_n)\,\right|}{\textsc{Tol} + \max\Big(|\phi(\boldsymbol{k}_m,t_n)|\,,\,\left|\tilde{\phi}(\boldsymbol{k}_m,t_n)\right|\Big)\times\textsc{Tol}}\right)^2}\,, \tag{16}$$

where M is the number of spatial modes, square brackets denote the integer part, $\phi(\boldsymbol{k}_m, t_n)$ is the N-th order solution at the m-th Fourier mode, the "tilde" notation indicating the solution at order $N-1$, and $\textsc{Tol}$ is the tolerance (parameter defining the desired precision of the time-integration). The time step h_n is accepted if the error Δ_n is smaller than the tolerance $\textsc{Tol}$, otherwise h_n is reduced and this step is recomputed. h_n being accepted, the next time step h_{n+1} is obtained assuming the largest error equal to the tolerance. In order to avoid an excess of rejected time steps, we use the Proportional Integral (PI) Step Control [28], which chooses the optimal time step h_{n+1} as

$$h_{n+1} \;=\; h_n\,\Delta_n^{-b}\,\Delta_{n-1}^{c}, \tag{17}$$

where $b = 0.7/p$, $c = 0.4/p$, p being the order of the chosen integrator [13]. Interested readers should refer to [1] for details on this classical procedure.

3.2. Optimum time step

Since the constant $\mathcal{C}_n$ can be chosen freely, we seek for the value of $\mathcal{C}_n$ providing the largest h_{n+1}, namely, to maximise the right-hand side of (17). Since h_n and Δ_{n-1} are determined at the previous time-step, only Δ_n in (17) depends on $\mathcal{C}_n$. Thus, in order to maximize h_{n+1}, Δ_n must be minimized, i.e., one must solve $\mathrm{d}\Delta_n/\mathrm{d}\mathcal{C}_n = 0$. This derivation being characterized by cumbersome algebra for general embedded Runge–Kutta schemes, we illustrate the case of the Heun algorithm (that is a second-order Runge–Kutta method with an embedded first-order explicit Euler scheme for the time stepping [27]), the principle being the same for higher order integrators. Also for simplicity, we give the calculations in one dimension (1D) without loss of generality, since the final result is independent of the spatial dimension.

3.2.1. Optimum $\mathcal{C}_n$ for Heun's method

Heun's method consists, here, in solving the initial value problem (for $t \geqslant t_n$)

$$\mathrm{i}\,\partial_t\,\phi \;=\; f(\boldsymbol{k}, t; \phi; \mathcal{C}_n) \stackrel{\text{def}}{=} -\mathrm{i}\,\exp\!\Big[\,\tfrac{\mathrm{i}}{2}\,k^2\,(t-t_n)\,\Big]\,\mathcal{F}\{\,(V+\mathcal{C}_n)\,\psi\,\}, \tag{18}$$

and

$$\phi(\boldsymbol{k}, t) \stackrel{\text{def}}{=} \exp\!\Big[\,\tfrac{\mathrm{i}}{2}\,k^2\,(t-t_n)\,\Big]\,\mathcal{F}\{\psi(\boldsymbol{x}, t)\}. \tag{19}$$

Hereafter, for brevity, we denote

$$\phi_n \;=\; \phi_n(\boldsymbol{k}) \;\stackrel{\text{def}}{=}\; \phi(\boldsymbol{k}, t_n), \qquad \psi_n \;=\; \psi_n(\boldsymbol{x}) \stackrel{\text{def}}{=} \psi(\boldsymbol{x}, t_n), \qquad V_n \;=\; V_n(\boldsymbol{x}) \stackrel{\text{def}}{=} V(\boldsymbol{x}, t_n). \tag{20}$$

At time $t = t_{n+1}$, the first- and second-order (in h_n) approximations of ϕ, respectively $\widetilde{\phi}_{n+1}$ and ϕ_{n+1}, are

$$\widetilde{\phi}_{n+1} \;=\; \phi_n \;+\; h_n\,f(\boldsymbol{k}, t_n; \phi_n; \mathcal{C}_n), \tag{21}$$

$$\phi_{n+1} \;=\; \phi_n \;+\; \tfrac{1}{2}\,h_n\,[\,f(\boldsymbol{k}, t_n; \phi_n; \mathcal{C}_n) \;+\; f(\boldsymbol{k}, t_n + h_n; \phi_n + h_n\,f(\boldsymbol{k}, t_n; \phi_n; \mathcal{C}_n); \mathcal{C}_n)\,]\,. \tag{22}$$

The next time-step h_{n+1} is chosen using equation (17). For our equation, the difference between the first- and second-order approximations $\Delta\phi_{n+1} \stackrel{\text{def}}{=} \left|\phi_{n+1} - \widetilde{\phi}_{n+1}\right|$ is such that

$$\begin{aligned}(\Delta\phi_{n+1})^2 \;&=\; \tfrac{1}{4}\,h_n^2\,|\,f(\boldsymbol{k}, t_n; \phi_n; \mathcal{C}_n) \;-\; f(\boldsymbol{k}, t_n + h_n; \phi_n + h_n\,f(\boldsymbol{k}, t_n; \phi_n; \mathcal{C}_n); \mathcal{C}_n)\,|^2 \\ &=\; \tfrac{1}{4}\,h_n^2\,\Big|\,f(\boldsymbol{k}, t_n; \phi_n; \mathcal{C}_n) \;+\; \mathrm{i}\,\mathrm{e}^{\mathrm{i}k^2 h_n/2}\times \\ &\qquad\qquad \mathcal{F}\left\{(V_{n+1}+\mathcal{C}_n)\mathcal{F}^{-1}\left\{\mathrm{e}^{-\mathrm{i}k^2 h_n/2}\,(\phi_n + h_n\,f(\boldsymbol{k}, t_n; \phi_n; \mathcal{C}_n))\right\}\right\}\Big|^2,\end{aligned} \tag{23}$$

where $V_{n+1} = V(\boldsymbol{x}, t_n + h_n)$. We note that the absolute value in (23) is of first-order in h_n, as one can easily check with a Taylor expansion around $h_n = 0$, so $(\Delta\phi_{n+1})^2 = O\left(h_n^4\right)$. More precisely, after some elementary algebra, one finds

$$\begin{aligned}(\Delta\phi_{n+1})^2 \,&= \tfrac{1}{4}\, h_n^4\, \left|\mathcal{F}\left\{(V_n + \mathcal{C}_n)^2\, \psi_n\right\} \,+\, \mathrm{i}\,\mathcal{F}\{\partial_t V_n\, \psi_n\} \,+\, \mathcal{F}\{\partial_x V_n\, \partial_x \psi_n\}\right. \\ &\quad \left. +\, \tfrac{1}{2}\,\mathcal{F}\{\partial_{xx} V_n\, \psi_n\}\,\right|^2 \,+\, O\left(h_n^5\right),\end{aligned} \tag{24}$$

which, defining

$$\alpha(\boldsymbol{x}, t; \mathcal{C}_n) \stackrel{\text{def}}{=} (V_n + \mathcal{C}_n)^2\, \psi_n, \qquad \beta(\boldsymbol{x}, t) \stackrel{\text{def}}{=} \mathrm{i}\, \partial_t V_n\, \psi_n \,+\, \partial_x V_n\, \partial_x \psi_n + \tfrac{1}{2}\, \partial_{xx} V_n\, \psi_n, \tag{25}$$

can be rewritten as

$$(\Delta\phi_{n+1})^2 \,= \tfrac{1}{4}\, h_n^4\, \left|\mathcal{F}\{\alpha(\mathcal{C}_n) + \beta\}\right|^2 \,+\, O\left(h_n^5\right). \tag{26}$$

Introducing the mean quadratic error

$$E_n(\mathcal{C}_n) \stackrel{\text{def}}{=} \frac{1}{M} \sum_{m=-[M/2]}^{[M/2]-1} \Delta\phi_{n+1}^2(\boldsymbol{k}_m, t_n; \mathcal{C}_n), \tag{27}$$

substituting (26) into (27) and exploiting the definition of the discrete Fourier transform, one obtains (using Parseval theorem)

$$\begin{aligned}E_n(\mathcal{C}_n) \,&= \frac{1}{M} \sum_{m=-[M/2]}^{[M/2]-1} \left| \sum_{\ell=-[M/2]}^{[M/2]-1} \mathrm{e}^{-2\mathrm{i}\pi m\ell/M} (\alpha_\ell(\mathcal{C}_n) + \beta_\ell) \right|^2 \tfrac{1}{4}\, h_n^4 \,+\, O\left(h_n^5\right) \\ &= \sum_{\ell=-[M/2]}^{[M/2]-1} |\alpha_\ell(\mathcal{C}_n) + \beta_\ell|^2\, \tfrac{1}{4}\, h_n^4 \,+\, O\left(h_n^5\right).\end{aligned} \tag{28}$$

The minimum of $E_n(\mathcal{C}_n)$, obtained from the equation $\mathrm{d}E_n(\mathcal{C}_n)/\mathrm{d}\mathcal{C}_n = 0$, is such that

$$\sum_{\ell=-[M/2]}^{[M/2]-1} \frac{\mathrm{d}\,|\alpha_\ell(\mathcal{C}_n)|^2}{\mathrm{d}\mathcal{C}_n} \,+\, 2\,\mathrm{Re}\left(\frac{\mathrm{d}\alpha_\ell(\mathcal{C}_n)}{\mathrm{d}\mathcal{C}_n}\, \beta_\ell^*\right) \,=\, 0. \tag{29}$$

Therefore, the optimum $\hat{\mathcal{C}}_n$ providing the largest h_{n+1}, in the case of Heun's method, is a solution of (29).

3.2.2. *Optimum $\mathcal{C}_n$ for generic embedded Runge–Kutta schemes*

The optimum $\mathcal{C}_n$ for general embedded Runge–Kutta schemes can be obtained following the same principles illustrated above with the Heun algorithm. However, the algebraic calculations get rapidly very cumbersome, leading to expensive computations that, in most cases, exceeds the time gained with a larger step. Here, we sketch-out the procedure for generic embedded Runge–Kutta methods, considering solvers of order N with an embedded $(N-1)$-order scheme (for other embedded or extrapolation methods, the procedure is completely analogue). For a s-stage method, the error $\Delta\phi_{n+1}$ can be written as [28]

$$(\Delta\phi_{n+1})^2 \,= \left|\sum_{\ell=1}^{s} d_\ell\, w_\ell\right|^2, \quad d_\ell \stackrel{\text{def}}{=} a_{s,\ell} - b_\ell, \quad w_\ell \stackrel{\text{def}}{=} h_n\, f\left(\boldsymbol{k}, t_n + c_\ell\, h_n; \phi_n + \sum_{r=1}^{\ell-1} a_{\ell,r}\, w_r; \mathcal{C}_n\right), \tag{30}$$

where $a_{\ell,r}$, b_ℓ and c_ℓ are the coefficients of the Butcher tableau which characterizes the integrator [28]. Using Taylor expansions and un-nesting the scheme, it is possible to prove that a result with a similar structure compared with (26) is obtained. In this case, the number of stages s appears as exponent in the function α, which takes the form $\alpha(\boldsymbol{x}, t; \mathcal{C}_n) = (V_n + \mathcal{C}_n)^{2s}\, \psi_n$. The function β, on the other hand, becomes explicitly dependent on $\mathcal{C}_n$, involving a number of terms growing exponentially with s. For this reason, even though the exact result can always be achieved, the computational time needed to minimize the error (16) is often larger than the time gained with a larger step, especially for higher order schemes ($s > 3$). In the next section, we show how, for practical applications, an exact solution is not necessary to improve the algorithm, and (15) represents a fast and accurate method.

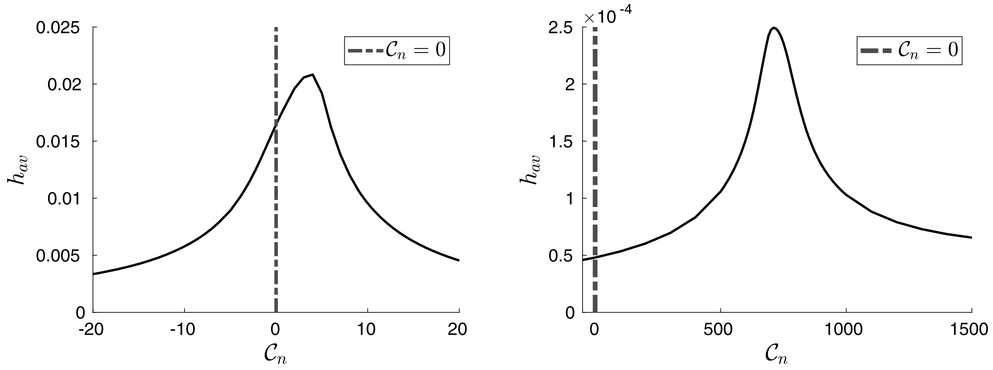

Fig. 1. Average time-step $h_{av} = \frac{1}{N_h}\sum_{n=1}^{N_h} h_n$ with a constant $\mathcal{C}_n$ for the IF method applied to the one dimensional NLS (left) and SN (right) equations.

Table 1
Comparisons for the SN and the NLS equations, in one and two spatial dimensions, between different values of $\mathcal{C}_n$.

Eq.	SN$_{1D}$	SN$_{1D}$	SN$_{2D}$	SN$_{2D}$	NLS$_{1D}$	NLS$_{1D}$	NLS$_{2D}$	NLS$_{2D}$
$\mathcal{C}$	0	$\tilde{\mathcal{C}}_n$	0	$\tilde{\mathcal{C}}_n$	0	$\tilde{\mathcal{C}}_n$	0	$\tilde{\mathcal{C}}_n$
$N_{\Delta t}$	20819	3871	8382	2682	6047	4781	754	690
$T(s)$	66.7	12.1	12856	4736	18.9	14.5	23769	22843

4. Numerical examples

Here, we consider numerical examples where we apply this method, focusing on both the SN and NLS equations solved with the Dormand and Prince 5(4) integrator [12] in one and two spatial dimensions. In all cases, we set open boundary conditions for the potential, while the initial conditions and the value of the physical parameters are chosen to be very close to regimes of physical interest, as described in [10,15,18]. Notice that, in the most general case, one should take into account the geometry of the domain and the nature of boundary conditions to calculate the potential V in the SN case. This operation is however straightforward in the considered examples, as we focused on rectangular domains using pseudo-spectral methods. Details on the numerical simulations can be found in Appendix A. The gain factor provided by the method depends on the optimal value of $\mathcal{C}_n$ compared to the $\mathcal{C}_n = 0$ case, which changes from case to case as a function of the boundary conditions for the potential and of the profile of the solution. Specifically, since the gain factor is evaluated with respect to the $\mathcal{C}_n = 0$ case, the more the optimal value of $\mathcal{C}_n$ is far from zero, the larger the gain factor gets.

For the one-dimensional NLS, some analytical stationary solutions are known. We then use one of these solutions (see Appendix A) as initial condition. For all other cases (SN and NLS 2D), no such stationary solutions are known, so we use gaussian initial conditions.

In Fig. 1, we show the average time-step $h_{av} = \sum_{n=1}^{N_h} h_n / N_h$, for an entire simulation with N_h time steps, as a function of $\mathcal{C}_n$ for the one-dimensional SN and NLS equations. These plots are generated taking $\mathcal{C}_n$ constant for the entire simulations, in order to better appreciate the strong dependence of the time-step on the choice of the gauge for the potential. In Fig. 2, we report the result of simulations performed choosing the near optimal $\mathcal{C}_n = \tilde{\mathcal{C}}_n$ at each time-step. Note that, for the one-dimensional NLS, the solution being stationary, h_n and the optimum $\mathcal{C}_n$ do not change in time, that is not the case in 2D. We show the time-step h_n as a function of time for the one-dimensional SN and NLS equations, comparing the $\mathcal{C}_n = 0$ case with $\mathcal{C}_n = \tilde{\mathcal{C}}_n$. In both cases, the time-step chosen by the algorithm with the optimisation of the gauge constant proves to be larger, compared to the $\mathcal{C}_n = 0$ case. In Table 1, we show the number of time-loops $N_{\Delta t}$ required to run each simulation and the time T needed to run the simulation (in seconds) for the cases $\mathcal{C}_n = 0$ and $\mathcal{C} = \tilde{\mathcal{C}}_n$. For NLS, in the one dimensional case we achieve roughly a 30% improvement in terms of speed gain between the $\mathcal{C} = 0$ and $\mathcal{C} = \tilde{\mathcal{C}}_n$ cases, while in two dimensions the speed gain is only approximately 10% since, here, the value of $\tilde{\mathcal{C}}_n$ is very close to zero. As long as the SN equation is concerned, (15) proved to reduce remarkably both the number of time loops and the effective time for the total simulation, providing up to a factor 5 of improvement with respect to the $\mathcal{C}_n = 0$ case in 1D and up to a factor 3 in 2D.

5. Conclusion

Exploiting a gauge condition on the potential, we optimized the integrating factor technique applied to the nonlinear Schrödinger and Schrödinger–Newton equations. Although the exact values of the piecewise constant $\mathcal{C}_n$ minimizing the error (16) (therefore maximizing the time-step) is in principle always possible to compute (e.g., with a computer algebra system), its expression depends on the particular numerical scheme chosen and it becomes complicated as the order of the method increases, resulting in a high computational cost. However, the near-optimal value obtained from the first approach

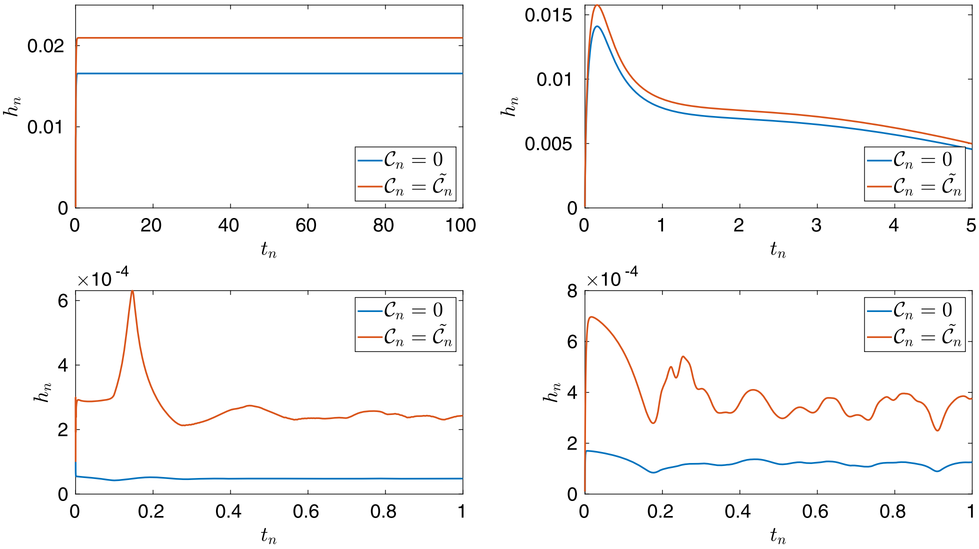

Fig. 2. Comparison between $\mathcal{C}_n = \tilde{\mathcal{C}_n}$ and $\mathcal{C}_n = 0$ for time-step h_n as a function of time, for the IF method applied to the NLS$_{\mathrm{1D}}$ (top left), NLS$_{\mathrm{2D}}$ (top right), SN$_{\mathrm{1D}}$ (bottom left) and SN$_{\mathrm{2D}}$ (bottom down).

we described, based on the minimization of the L_2-norm of the nonlinear part of the equation, proved to be an accurate and efficient solution in the tested cases. Thus, being computationally extremely cheap and independent of the particular numerical scheme employed, this is the approach one should choose for most simulations, at least when the computation of $\mathcal{N}$ is not very expensive. For Schrödinger-like equations with hard to compute potentials, most of the computational time is spent in the calculation of $\mathcal{N}$. For these very demanding equations, the extra cost needed to compute the optimum $\hat{\mathcal{C}_n}$ (instead of the near optimum $\tilde{\mathcal{C}_n}$) is negligible in comparison, so $\hat{\mathcal{C}_n}$ could be preferable.

For the cases tested here, we found a speed-up in the computation time up to a factor 5, the speed-up depending on the equation and on the physical regime. These examples show that this approach provides significant speed improvements, that with minor modifications of the original algorithm. Though we focused on the nonlinear Schrödinger and Schrödinger–Newton equations, the method principle is independent on the particular potential considered, so this approach can be extended to other Schrödinger-like equations. More generally, the idea behind the method presented in this note can be, at least in principle, generalized and extended to other equations with similar gauge conditions.

Appendix A. Numerical simulations

For the one-dimensional NLS, we considered the case $g = -1$ (see eqs. (1) and (3)) and we used $\psi(x, t = 0) = \sqrt{2}\,\mathrm{sech}\left(\sqrt{2}\,x\right)$ as initial condition. We discretized the space with $N = 2048$ points, in a computational box of length $L = 80$. The two-dimensional NLS, which is often employed in optics to model self-focusing beams in a medium with a cubic non-linearity [19,30], presents a finite time (blow-up) singularity [26]. More specifically, whenever the initial condition ψ_0 satisfies $E_g = \int \mathrm{d}\boldsymbol{r}\,\psi_0 \left(-\frac{1}{2}\nabla^2 + \frac{g}{2}\,|\psi_0|^2\right)\psi_0^* < 0$, the norm of the solution, or of one of its derivatives, becomes unbounded in finite time. For this reason, we stop the simulation at $t_{\mathrm{fin}} = 5$, i.e., before the singularity occurs. We set $\psi(\boldsymbol{r}, t = 0) = \mathrm{e}^{-r^2/2}/\sqrt{\pi}$ as initial condition and we consider the $g = -6$ case, for which the corresponding initial energy is $E(g = -6) \approx 0.02$, hence quite close to the singular regime; for the spatial discretization we used $L = 120$ and $N = 4096^2$ (squared box with side $L = 120$ discretized with 4096×4096 nodes). For both the one and two dimensional SN equations, we set $g = 500$ and considered a Gaussian initial condition, $\psi(\boldsymbol{x}, t = 0) = \mathcal{N}\mathrm{e}^{-|\boldsymbol{x}|^2/2}$ where $\mathcal{N}$ is the normalisation factor, fixed such that $\int \mathrm{d}\boldsymbol{x}\,|\psi(\boldsymbol{x}, t = 0)|^2 = 1$. The parameters of the spatial discretization are $L = 20$ and $N = 2048$ in 1D, while for the 2D case we set $L = 20$ and $N = 1024^2$. To solve the SN and the NLS equation we used the Dormand and Prince 5(4) integrator [12].

References

[1] R. Alexander, Solving ordinary differential equations i: nonstiff problems (e. hairer, sp norsett, and g. wanner), SIAM Rev. 32 (3) (1990) 485.

[2] W. Auzinger, I. Březinová, A. Grosz, H. Hofstätter, O. Koch, T. Sato, Efficient adaptive exponential time integrators for nonlinear Schrödinger equations with nonlocal potential, J. Comput. Math. Data Sci. 1 (2021) 100014.

[3] P. Bader, S. Blanes, F. Casas, M. Thalhammer, Efficient time integration methods for Gross–Pitaevskii equations with rotation term, arXiv preprint arXiv:1910.12097.

[4] C. Besse, G. Dujardin, I. Lacroix-Violet, High order exponential integrators for nonlinear Schrödinger equations with application to rotating Bose–Einstein condensates, SIAM J. Numer. Anal. 55 (3) (2017) 1387–1411.

[5] J. Binney, S. Tremaine, Galactic Dynamics, second edition, 2008.
[6] S. Blanes, F. Casas, A. Murua, Splitting and composition methods in the numerical integration of differential equations, arXiv preprint arXiv:0812.0377.
[7] J.C. Butcher, Numerical Methods for Ordinary Differential Equations, John Wiley & Sons, 2016.
[8] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods. Fundamentals in Single Domains, 2nd edition, Scientific Computation, Springer, 2008.
[9] F. Dabby, J. Whinnery, Thermal self-focusing of laser beams in lead glasses, Appl. Phys. Lett. 13 (8) (1968) 284–286.
[10] F. Dalfovo, S. Giorgini, L.P. Pitaevskii, S. Stringari, Theory of Bose–Einstein condensation in trapped gases, Rev. Mod. Phys. 71 (3) (1999) 463.
[11] L. Diosi, Gravitation and quantumm-echanical localization of macroobjects, arXiv preprint arXiv:1412.0201.
[12] J.R. Dormand, P.J. Prince, A family of embedded Runge–Kutta formulae, J. Comput. Appl. Math. 6 (1) (1980) 19–26.
[13] K. Gustafsson, Control theoretic techniques for stepsize selection in explicit Runge–Kutta methods, ACM Trans. Math. Softw. 17 (4) (1991) 533–554.
[14] F.S. Guzman, L.A. Urena-Lopez, Gravitational cooling of self-gravitating Bose condensates, Astrophys. J. 645 (2) (2006) 814.
[15] A. Hasegawa, Optical solitons in fibers, in: Optical Solitons in Fibers, Springer, 1989, pp. 1–74.
[16] M. Hochbruck, J. Leibold, A. Ostermann, On the convergence of Lawson methods for semilinear stiff problems, Numer. Math. 145 (3) (2020) 553–580.
[17] W. Hu, R. Barkana, A. Gruzinov, Fuzzy cold dark matter: the wave properties of ultralight particles, Phys. Rev. Lett. 85 (6) (2000) 1158.
[18] C. Kharif, E. Pelinovsky, A. Slunyaev, Rogue Waves in the Ocean, Springer Science & Business Media, 2008.
[19] K. Konno, H. Suzuki, Self-focussing of laser beam in nonlinear media, Phys. Scr. 20 (3–4) (1979) 382.
[20] J.D. Lawson, Generalized Runge–Kutta processes for stable systems with large Lipschitz constants, SIAM J. Numer. Anal. 4 (3) (1967) 372–380.
[21] D.J. Marsh, J.C. Niemeyer, Strong constraints on fuzzy dark matter from ultrafaint dwarf galaxy Eridanus ii, Phys. Rev. Lett. 123 (5) (2019) 051103.
[22] A. Paredes, H. Michinel, Interference of dark matter solitons and galactic offsets, Phys. Dark Universe 12 (2016) 50–55.
[23] R. Penrose, On gravity's role in quantum state reduction, Gen. Relativ. Gravit. 28 (5) (1996) 581–600.
[24] C. Rotschild, O. Cohen, O. Manela, M. Segev, T. Carmon, Solitons in nonlinear media with an infinite range of nonlocality: first observation of coherent elliptic solitons and of vortex-ring solitons, Phys. Rev. Lett. 95 (21) (2005) 213904.
[25] C. Rotschild, B. Alfassi, O. Cohen, M. Segev, Long-range interactions between optical solitons, Nat. Phys. 2 (11) (2006) 769–774.
[26] C. Sulem, P.-L. Sulem, The Nonlinear Schrödinger Equation: Self-Focusing and Wave Collapse, Vol. 139, Springer Science & Business Media, 2007.
[27] E. Süli, D.F. Mayers, An Introduction to Numerical Analysis, Cambridge University Press, 2003.
[28] G. Wanner, E. Hairer, Solving Ordinary Differential Equations II, Springer, Berlin Heidelberg, 1996.
[29] L.M. Widrow, N. Kaiser, Using the Schrödinger equation to simulate collisionless matter, Astrophys. J. 416 (1993) L71.
[30] V. Zakharov, V. Synakh, The nature of the self-focusing singularity, Zh. Èksp. Teor. Fiz. 68 (1975) 940–947.