**SCALAR: an AMR code to simulate axion-like dark matter models**

Mattia Mina,1,∗ David F. Mota,1 and Hans A. Winther1,2

1Institute of Theoretical Astrophysics, University of Oslo, 0315 Oslo, Norway
2Institute of Cosmology & Gravitation, University of Portsmouth, Portsmouth, Hampshire, PO1 3FX, UK

We present a new code SCALAR, based on the high-resolution hydrodynamics and N-body code RAMSES, to solve Schrödinger equation on adaptive refined meshes. The code is intended to be used to simulate axion or Fuzzy Dark Matter models where the evolution of the dark matter component is determined by a coupled Schrödinger-Poisson equation, but can also be used as a stand-alone solver for both linear and non-linear Schrödinger equations with any given external potential. This paper describes the numerical implementation of our solver and present tests to demonstrate how accurately it operates.

I. INTRODUCTION

The true nature of dark matter is not known. Weakly Interacting Massive Particles (WIMP) are still considered one of the most likely candidates for Cold Dark Matter (CDM), and several experiments are ongoing trying to detect such particles. These are however closing in on the neutrino floor where any signal would be drowned in the solar neutrino background [1].

A promising alternative to WIMPs are Ultra-Light Axions, Fuzzy Dark Matter [2–12] and Superfluid Dark Matter [13–16]. These models have distinct and observable signatures on the small scales of structure formation, as well as are able to solve some of the discrepancies observed in Cold Dark Matter simulations like the missing satellites problem [17, 18], the cusp-core problem [19] and the too-big-to-fail problem [20]. These disparities could, however, have a solution within baryonic processes, usually not included in standard CDM simulations as shown in [21–29]. To understand this better one should ideally perform simulations including both of these components.

In order to quantify the effects of axion-like dark matter models one needs to either solve a Schrödinger-Poisson system, or use the Mandelung formulation - which is the traditional hydrodynamical equations with an additional pressure term, which can be solved using methods like Smoothed Particle Hydrodynamics (SPH) as proposed in [30].

The Schrödinger-Poisson system has several applications in cosmology. For instance, the six dimensional Vlasov equation describing collisionless self-gravitating matter is approximated by a Schrödinger-Poisson system for a complex wave-function in three dimensions. This was proposed as a way for simulating Cold Dark Matter in [31]. It was later shown, by solving the Schrödinger-Poisson system and comparing to the Vlasov solver CoLDICE [32] in two dimensions, that one has excellent qualitative and quantitative agreement of the solution [33]. A similar study is given in [34], where the system is solved using a spectral method and demonstrated that one recovers the classical behavior in the limit where $\hbar \rightarrow 0$.

Unfortunately, the methods employed in the above mentioned papers, in spite of being very accurate, are too expensive to perform high-resolution simulations in three dimensions. The first cosmological, high-resolution, simulation of Fuzzy Dark Matter in three dimensions was performed in [35] using the code GADGET [36, 37]. In there an explicit method, similar to the one we will present in this paper, was used. Since this work there have been a handful papers that have performed such simulations more in line with the hydrodynamical formulation. In [38] a new technique to discretise the quantum pressure was proposed and shown to reproduce the expected density profile of dark matter halos. In [39] a module AX-GADGET for cosmological simulations using Smoothed Particle Hydrodynamics (SPH) inside the P-GADGET3 code was presented. These methods do not solve for the wave-function, but have the advantage of being much less expensive to run than a full wave-function solver like ours.

In this paper we present SCALAR (Simulation Code for Ultra Light Axions in RAMSES); a general Adaptive Mesh Refinement (AMR) code to solve the Schrödinger-Poisson system. Our implementation is in the hydrodynamics and N-body code RAMSES [40].

The structure of the paper is as follows: in Sec. II we present the equations we are to solve, in Sec. III we present the numerical implementation, in Sec. IV we present tests of the code before concluding in Sec. V.

II. THEORETICAL MODEL

A Bose-Einstein Condensate (BEC) is a system of identical bosons, where a large fraction of particles occupies the lowest quantum energy state, or the ground state. This phenomenon typically takes place in gases, at very low temperatures or very high densities and it was observed for the first time in [41, 42]. In the condensate regime, these quantum systems behave as a macroscopic fluid and their peculiar features are a macroscopic manifestation of quantum effects.

In general, when Bose-Einstein condensation occurs, thermal de-Broglie wavelengths of particles start to overlap, as they become grater than the mean inter-particle distance. At this point, a coherent state develops and the system behaves as a macroscopic fluid, where only binary collisions are relevant. The dynamics of BECs is complicated, due to the difficulty in modelling particle self-interactions.

However, in the Hartree mean-field theory and in the limit of $T \rightarrow 0$, binary collisions are modelled via an effective po-

* mattia.mina@astro.uio.no
tential and the whole quantum system can be described by a single-particle wave-function \( \psi(x, t) \) obeying the non-linear Schrödinger equation:

\[
\frac{i\hbar}{\partial t} \psi = \left[ -\frac{\hbar^2}{2m} \nabla^2 + g|\psi|^2 + mV_{\text{ext}} \right] \psi, \quad (1)
\]

where \( m \) is the mass of a boson and \( g \) is the self-interaction coupling constant. Often, the trapping potential \( V_{\text{ext}}(x, t) \) is introduced by hand in order to model the presence of a trap, which is responsible for keeping particles confined.

The single-particle wave-function is normalised such that:

\[
\int |\psi|^2 \, d^3x = N, \quad (2)
\]

where \( N \) is the total number of particles present in the system. As a consequence, the quantity \( |\psi(x, t)|^2 \) represents the number density of particles.

An alternative description of the macroscopic fluid is provided by the so-called Madelung formulation of the Schrödinger equation. In this case, by expressing the single-particle wave-function in polar coordinates:

\[
\psi = \sqrt{\frac{\rho}{m}} \exp \left( \frac{i}{\hbar} \theta \right), \quad (3)
\]

the dynamics of the system is described in terms of mass density and velocity, which are macroscopic physical quantities and they are respectively defined as:

\[
\rho(x, t) = m|\psi(x, t)|^2, \quad (4)
\]

\[
\mathbf{v}(x, t) = \frac{1}{m} \nabla \theta(x, t). \quad (5)
\]

Thus, the Schrödinger equation can be cast into the following system of equations:

\[
\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \quad (6)
\]

\[
\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} = -\nabla \left( V_{\text{ext}} + \frac{g}{m^2} \rho + Q \right), \quad (7)
\]

which are known as the Madelung or quantum Euler equations. We recognise Eq. (6) as a continuity equation which expresses conservation of mass. Although the second Madelung equation, Eq. (7), expresses conservation of momentum, it is not the same as the classical momentum equation, as it contains an additional term \( Q \), which is called quantum pressure and it is defined as:

\[
Q \equiv -\frac{\hbar^2}{2m^2} \nabla^2 \sqrt{\rho}. \quad (8)
\]

The quantum pressure is a macroscopic manifestation of quantum effects and it is characteristic of Bose-Einstein condensates.

In this formulation, by defining the velocity as in Eq. (5), we are intrinsically assuming that the fluid is irrotational, since:

\[
\nabla \times \mathbf{v} = \nabla \times \nabla \theta = 0. \quad (9)
\]

In cosmology, these kinds of systems can be used to model the dark matter contribution to the energy budget of the Universe. In particular, in the last few decades, models where dark matter is a light boson, such as Ultra-Light Axions or Fuzzy Dark Matter, have received a lot of attention. Due to the small mass of these bosons, macroscopic quantum effects manifest at astronomically relevant scales. In these alternative dark matter models, new signatures are expected within the structure formation process at highly non-linear scales and, therefore, numerical simulations are required in order to explore these scenarios.

Here, the dynamics of dark matter is also described by a system of identical bosons gravitationally bounded. Therefore, the governing equation is a non-linear Schrödinger equation, Eq. (1), where the external potential is replaced by the gravitational potential. In this class of alternative dark matter models, self-interactions between bosons are often neglected, as the coupling constant \( g \) is usually parametrically small. The resulting system of equations describing the dynamics of the dark matter fluid is called Schrödinger-Poisson and, for an expanding Universe, it reads:

\[
\frac{i\hbar}{\partial t} \left( \frac{\partial \psi}{\partial t} + \frac{3}{2} H \psi \right) = \left( -\frac{\hbar^2}{2m} \frac{\nabla^2}{\sqrt{\rho}} + ma \Phi \right) \psi, \quad (10)
\]

\[
\nabla^2 \Phi = 4\pi Ga^2 \left( |\psi|^2 - |\psi(a)|^2 \right), \quad (11)
\]

where \( a \) is the scale-factor of the Universe, \( H \equiv d\log(a)/dt \) is the Hubble rate of expansion and \( \Phi \) is the gravitational potential. With a change of variables \( \psi \rightarrow a^{3/2} \psi \) the Schrödinger equation above takes on the form of Eq. (1).

III. NUMERICAL METHODS

In this section we provide a brief overview of the code RAMSES and the AMR technique. Then, we discuss in details the numerical aspects of the algorithm we implemented in order to solve the non-linear Schrödinger equation. Throughout this section, the dimensionality of the problem is denoted by \( \text{dim} \) and it can be 1, 2 or 3.

A. Overview of RAMSES

The RAMSES code was originally designed for cosmological simulations of structure formation and subsequently extended to astrophysical applications. It consists of an N-body Particle Mesh (PM) code, which solves the gravitational dynamics of a set of macroparticles, sampling the phase space distribution of the dark matter component in the Universe. Through PM algorithms, the mass of each macroparticle is interpolated on a grid and the Poisson equation is solved in order to compute the gravitational potential. Thus, the gravitational force acting on each element of the system and the new phase space position of each macroparticle are computed by solving the corresponding N-body equation with a leapfrog scheme. In addition, RAMSES can solve the dynamics of the baryonic component present in the Universe. In this case, the
grid is also used to sample gas parcels and the evolution of the system is described by the equations of hydrodynamics, which are solved by means of a Godunov scheme. For this purpose, Riemann solvers can be used for computing fluxes of conserved physical quantities among cells.

The RAMSES code implements an AMR strategy, where a hierarchy of nested grids is created in order to increase the local resolution according to a set of refinement criteria. In this way, RAMSES can solve accurately gas dynamics and gravitational potential only where more precision is actually needed. This approach reduces consistently the amount of memory needed in cosmological and hydrodynamical simulations, compared to the case where a uniform high-resolution grid is used.

In SCALAR, we rely on the efficient AMR implementation of RAMSES. In order to solve the dynamics of our theoretical model, the single-particle wave-function is sampled by using the original grid allocated by RAMSES for the Poisson and hydrodynamics equations. Also in this case, the AMR approach provides the opportunity to solve the Schrödinger equation with higher resolution only where features of the wave-function are more demanding.

B. Adaptive Mesh Refinement

The basic unit of the AMR hierarchy is an oct, which consists of a set of \(2^{\dim} \) cells. At each level in the AMR hierarchy, a grid is a collection of octs with the same resolution. The grid with the coarsest resolution is called domain grid and it covers the whole computational domain. During the evolution of the physical system, when the solution starts to develop features and its tracking requires higher resolution, any cell at a given level can be split into a child oct, with double the resolution of the parent cell.

At each time-step, the AMR structure is modified according to a set of refinement criteria. First, for a generic level of refinement \( \ell \), a refinement map is created by marking all those cells satisfying at least one refinement criterion. Also cells violating the strict refinement rule are marked for refinement, in order to guarantee that each child oct at level \( \ell + 1 \) is surrounded by, at least, \( 3^{\dim} - 1 \) neighbors at the coarser level. However, if a given cell at level \( \ell \) does not satisfy any refinement criteria anymore, it is marked for de-refinement and subsequently its child octs are destroyed. Then, a new child oct is created at level \( \ell + 1 \) for each marked cell and all the relevant physical quantities are interpolated from level \( \ell \). Coarse-fine data interpolation, in an AMR context, is often called prolongation and it can be done by using any of the interpolation schemes which are described in the section below.

Often, there are regions where physical quantities fluctuate around a criterion threshold and the refinement map tends to be noisy. In order to avoid this issue, a mesh smoothing operator is applied and a cubic buffer of \( n_{\text{expand}} \) cells surrounding marked cells is refined as well.

C. The Schrödinger equation

The SCALAR code evolves the solution of the non-linear Schrödinger equation by using a Taylor’s method, similar to the one designed in GAMER. Given the wave-function \( \psi(x, t_0) \), the formal solution of the non-linear Schrödinger equation, Eq. (1), at time \( t_1 = t_0 + \Delta t \) reads:

\[
\psi(x, t_1) = \hat{U}(t_1, t_0)\psi(x, t_0),
\]  

(12)

where \( \hat{U}(t_1, t_0) \) is the time evolution operator and it maps the solution of the Schrödinger equation at two different times. In the general case, the time evolution operator is defined as:

\[
\hat{U}(t_1, t_0) = \exp \left( -\frac{i}{\hbar} \int_{t_0}^{t_1} \hat{H}(x, t') \, dt' \right),
\]  

(13)

where \( \hat{H}(x, t) \) denotes the Hamiltonian of the system. The operator \( \hat{U}(t_1, t_0) \) has the main following properties:

- \( \hat{U}(t, t) = 1 \),
- \( \hat{U}(t_1, t_2) \hat{U}(t_2, t_3) = \hat{U}(t_1, t_3) \),
- \( \hat{U}(t_1, t_2) = \hat{U}(t_2, t_1) = \hat{U}^{-1}(t_2, t_1) \).

In the limit of \( \Delta t \ll 1 \), then the following approximation holds:

\[
\int_{t_0}^{t_1} \hat{H}(x, t') \, dt' \approx \hat{H}(x, t_0) \Delta t,
\]  

(14)

and, therefore, Eq. (13) can be approximated as:

\[
\hat{U}(t_1, t_0) \approx \exp \left( -i \hat{H}(x, t_0) \Delta t \right).
\]  

(15)

In the general case, the Hamiltonian \( \hat{H}(x, t) \) contains different contributions to the total energy of the system. In particular, we can express \( \hat{H}(x, t) \) as a sum of contributions describing kinetic and potential energies. Here, we denote these two operators respectively as \( \hat{K}(x, t) \) and \( \hat{W}(x, t) \), and they are defined as:

\[
\hat{K}(x, t) \equiv -\frac{\hbar^2}{2m} \nabla^2,
\]  

(16)

\[
\hat{W}(x, t) \equiv m \left( \frac{V(x, t) + g}{m} |\psi(x, t)|^2 \right).
\]  

(17)

By means of the Lie-Trotter formula [43], the time evolution operator can be split as well:

\[
\hat{U}(t_1, t_0) \approx \exp \left( -i \hat{W}(x, t_0) \Delta t \right) \exp \left( -i \hat{K}(x, t_0) \Delta t \right).
\]  

(18)

As a consequence, the formal solution of the Schrödinger equation can be written as:

\[
\psi(x, t_1) = \exp \left( -i \hat{W}(x, t_0) \Delta t \right) \exp \left( -i \hat{K}(x, t_0) \Delta t \right) \psi(x, t_0).
\]  

(19)
In \textsc{scalar}, the two contributions to the time evolution operator are applied separately. First, the "drift" due to the kinetic part of the Hamiltonian is approximated via Taylor expansion (here for \(\dim = 3\)):

\[
\tilde{\psi}^{n+1}_{i,j,k} = \exp \left( -i\hat{K}(x, t_0)\Delta t \right) \psi^n_{i,j,k} = \left[ \sum_{N=0}^{\infty} \frac{1}{N!} \left( -\frac{i}{\hbar} \hat{K} \Delta t \right)^N \right] \psi^n_{i,j,k} = \left[ 1 + \left( \frac{\hbar^2 \Delta t}{2m} \nabla^2 \right) + \frac{1}{2} \left( \frac{\hbar^2 \Delta t}{2m} \nabla^2 \right)^2 + \ldots \right] \psi^n_{i,j,k},
\]

where, for a generic operator \(\hat{O}\), the notation \(\hat{O}^N\) denotes \(N\) consecutive applications of the same operator. In the \textsc{scalar} code, the Taylor expansion is performed up to \(O(\Delta t^3)\), which is the minimum required by the stability analysis of the numerical scheme, and the Laplacian operator is discretised by a standard second-order finite difference formula:

\[
\nabla^2 \psi^n_{i,j,k} = \frac{\psi^n_{i+1,j,k} + \psi^n_{i-1,j,k} - 2\psi^n_{i,j,k}}{\Delta x^2} + \frac{\psi^n_{i,j+1,k} + \psi^n_{i,j-1,k} - 2\psi^n_{i,j,k}}{\Delta y^2} + \frac{\psi^n_{i,j,k+1} + \psi^n_{i,j,k-1} - 2\psi^n_{i,j,k}}{\Delta z^2}.
\]

Then, the "kick" due to the potential is computed and the wave-function at the new time-step reads:

\[
\tilde{\psi}^{n+1}_{i,j,k} = \exp \left( -i\hat{W}^n_{i,j,k} \Delta t \right) \psi^n_{i,j,k}.
\]

Here, the advantage of the Lie-Trotter splitting is clear: while the kinetic contribution to the time evolution operator needs Taylor expansion in order to be applied, the potential contribution only provides a phase rotation of the wave-function and it can be computed exactly.

Once the new wave-function is computed, the new mass density is computed by:

\[
\rho^{n+1}_{i,j,k} |_S = \left| \tilde{\psi}^{n+1}_{i,j,k} \right|^2.
\]

\textbf{D. The continuity equation}

In quantum mechanics, the time evolution operator is unitary, as expressed by its properties. This means that the mass density carried by the wave-function is conserved. This is true also if we consider separately the two contributions to the Hamiltonian. However, the Taylor expansion, Eq. (20), breaks the unitarity of the time evolution operator. Therefore, in order to improve the conservation properties of our main numerical scheme, we implement a secondary solver for the continuity equation associated to the Schrödinger equation.

Eq. (6) can be written in its conservative form:

\[
\frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{j} = 0,
\]

where \(\rho(x, t)\) represents the mass density. Here, the quantity \(\mathbf{j}(x, t)\) is the associated density current, or flux, and it is defined as:

\[
\mathbf{j} = -\frac{\hbar}{2m} \left\{ \psi^n \nabla \psi - \psi \nabla \psi^* \right\}.
\]

In \textsc{scalar}, Eq. (24) is discretised by using a first-order Godunov scheme:

\[
\frac{\rho^{n+1}_{i,j,k} - \rho^n_{i,j,k}}{\Delta t} + \frac{\left( j^{n+\frac{1}{2}}_{i,j,k} - j^{n-\frac{1}{2}}_{i,j,k} \right)}{\Delta x} + \frac{\left( j^{n+\frac{1}{2}}_{i,j+1,k} - j^{n-\frac{1}{2}}_{i,j-1,k} \right)}{\Delta y} + \frac{\left( j^{n+\frac{1}{2}}_{i,j,k+1} - j^{n-\frac{1}{2}}_{i,j,k-1} \right)}{\Delta z} = 0,
\]

where the time centred fluxes are computed at cell interfaces. However, this solver is not used to explicitly advance in time the mass density, but only to enforce the conservation of mass. Indeed, by denoting \(\rho^{n+1}_{i,j,k} |_S\) and \(\rho^n_{i,j,k} |_C\), the new mass densities computed by the main and the secondary solvers respectively, a correcting factor is computed as follows:

\[
R = \frac{\rho^{n+1}_{i,j,k} |_C}{\rho^n_{i,j,k} |_S},
\]

which is used to rescale the wave-function.

This process does not ensure perfect conservation of mass, because the order of accuracy of the method we use to solve the continuity equation is lower than the order of accuracy of the kinetic solver. For this reason, by solving the continuity equation on top of the Schrödinger equation results in an overall better conservation properties of the algorithm.

\textbf{E. The solver}

\textsc{scalar} solves the Schrödinger equation from the coarser to the finer level in the AMR hierarchy. For a generic refinement level \(\ell\), the optimal time-step is chosen as:

\[
\Delta t = \min \left[ C_K \cdot \frac{m(\Delta x)^2}{2\sqrt{3}h}, C_W \cdot \frac{2\pi \hbar}{m V_{\text{max}}}, \right],
\]

where \(|V_{\text{max}}|\) denotes the maximum absolute value of the effective potential \(V + \frac{\hbar^2}{2m} |\psi|^2\). Here, \(C_K\) and \(C_W\) are Courant factors which are required to be smaller than one. The first term in the square brackets is determined by the Von Neumann stability analysis of the kinetic part of the solver. The second term, instead, requires that the phase of the wave-function does not rotate by a bigger angle than \(2\pi C_W\) within a time-step. In Appendix A we provide a detailed discussion of the Von Neumann stability analysis of the numerical scheme.

In the original \textsc{ramses} code two different options are available regarding the choice of the time-step: a single or...
an adaptive time-step. While the former consists in using the same time-step for all refinement levels and it is determined by the finest level in the AMR hierarchy, the latter allows to use smaller time-steps for finer refinement levels. However, in case of adaptive time-step, for each coarse time-step at level $\ell$ it is possible to perform only two fine time-steps at level $\ell+1$. In SCALAR, an additional option is available: a flexible time-step, where for each coarse time-step at level $\ell$, the number of fine steps at level $\ell+1$ is flexible and it is determined by a level dependent CFL condition. From Eq. (28), when the optimal time-step is chosen by the kinetic CFL condition $\Delta t \propto \Delta x^2$, which represents a stricter condition than the usual case of hydrodynamics equations. Therefore, a flexible time-step can reduce significantly the total amount of coarse time-step in a simulation.

Within a generic level of refinement $\ell$, SCALAR solves the Schrödinger equation for each oct separately. Thus, in order to advance the solution over a time-step, the solver proceeds as follows:

1. For a given oct, a cubic buffer of neighbours cells is collected. The equations are actually solved only for the central oct, while the buffer cells are used to compute Laplacians at each order of the Taylor expansion. If the central oct lies next to the coarse-fine boundary, the wave-function is interpolated into ghost cells from level $\ell-1$.

2. The kinetic solver evolves the wave-function at the new time-step by means of Eq. (20). First, by using a linear combination of these coefficients, the half time-step solution $\psi_{i,j,k}^{n+1/2}$ is estimated from $\psi_{i,j,k}^n$ and it will be used later in order to compute the density currents. Then, the full-step solution $\psi_{i,j,k}^{n+1}$ and the new density $\rho_{i,j,k}^{n+1}$ are computed.

3. The wave-function at half time-step $\psi_{i,j,k}^{n+1/2}$ is interpolated at cell interfaces and the time-centred density currents are computed by means of Eq. (25).

4. The continuity equation is solved and the new density $\rho_{i,j,k}^{n+1}|^C$ is computed via Eq. (26).

5. The rescaling factor given by Eq. (27) is computed and $\psi_{i,j,k}^{n+1}$ is rescaled in order to preserve mass conservation.

6. If the given oct lies next to the coarse-fine boundary, the estimated flux is stored for the subsequent reflux operation.

7. Finally, the phase rotation due to the potential is computed by applying Eq. (22) to $\psi_{i,j,k}^{n+1}$ and $\psi_{i,j,k}^{n+1}$ is evaluated.

All steps, except the first and the last ones, are performed separately for each physical dimension. This procedure is called dimensional splitting and it reduces a $N$-dimensional problem into a system of $N$ one dimensional problems. It has the advantage of relaxing the Courant condition of the solver and, therefore, it allows bigger time-steps. A flowchart of our solver is shown in Fig. (1).

FIG. 1. Flowchart of our solver. The kinetic solver is shown in blue (left) and the continuity solver is shown in red (right).

F. Refinement strategy.

In SCALAR, we implement the same "quasi-Lagrangian" as RAMSES uses for hydrodynamics: when the total mass of a given cell exceeds a given threshold, the cell is marked for refinement. The level dependent density threshold is defined as:

$$\rho_\ell = \frac{M_c}{(\Delta x)^2} \text{dim}, \quad (29)$$

where $M_c$ corresponds to the maximum mass allowed per cell.

In addition, following [35] and the FLASH code [44], we implement support for the invariant version of the Löhner error estimator. It is based on the second derivative of a given physical quantity, normalised by the first derivative. Considering a generic physical quantity $f$, the error estimator $E_L$ reads:
where the indices \(i,j\) run over each physical dimensions. Small fluctuations of the physical quantity \(f\) are filtered out due to the presence of the second term at denominator. The quantity \(\bar{f}_{ij}\) is an average of \(f\) over dimensions \(i,j\) and \(\epsilon\) is a small constant. This error estimator is dimensionless and therefore it can be applied to any physical quantity. Furthermore, in Eq. (30), \(E_L\) is bounded in the interval \([0,1]\). In SCALAR, we apply the Löhner error estimator separately to \(\Re[\psi]\) and \(\Im[\psi]\). Then, the final estimation of the error on the wave-function is given by:

\[
E_L = \sqrt{\left(E_{L}^{\Re}\right)^2 + \left(E_{L}^{\Im}\right)^2};
\]

and if it exceeds a user-defined threshold, the cell is marked for refinement.

**G. Spatial and temporal interpolation.**

In SCALAR, interpolation is required when:

- a generic level \(\ell\) in the AMR hierarchy is refined and new child octs are created at level \(\ell + 1\);
- during the solving process, boundary conditions need to be specified for a fine-grid patch and ghost cells are created.

In both cases, the wave-function at the coarse level \(\ell\) is interpolated down to level \(\ell + 1\). In order to solve the equations of motion, when the Laplacian operator is applied, any discontinuity in the second derivative of the wave-function introduces an error, which propagates into the solution of the non-linear Schrödinger equation and it can destroy the wave-function. Therefore, high-order interpolation schemes are implemented in order to keep the wave-function as smooth as possible.

In particular, in SCALAR, we implement two high-order interpolation schemes. In both cases, the interpolant function is a fourth-order polynomial, but the coefficients of the polynomials are chosen in different ways. In one case, Lagrange basis polynomials are computed in order to set the coefficients, resulting in a fourth-order Lagrange interpolation scheme. In the second case, fourth-order conservative interpolation is performed and the coefficients of the interpolant are set by requiring that cell averages of the interpolated quantities are preserved. In case of adaptive time integration, linear temporal interpolation can also be applied when computing boundary conditions for a fine level patch, since coarse-grid and fine-grid wave-functions can be discretised at different times.

Furthermore, the interpolation can be performed on two different sets of variables: the original set of variables \(\Re[\psi]\) and \(\Im[\psi]\), or derived variables \(|\psi|^2\) and \(\Arg[\psi]\), corresponding to mass density and phase of the wave-function.

The interpolation schemes and the set of variables used for the interpolation process can be specified by the user in the parameter file.

Further details on the interpolation schemes can be found in Appendix B.

**H. Artificial viscosity**

In the tests shown in the upcoming sections, when they were done at the domain level only, the solution of the non-linear Schrödinger equation remains stable for as long as we could run SCALAR. However, when refinements were included, the solver had the tendency to develop spurious high-frequency waves at coarse-fine boundary, even after improving the order of accuracy of interpolation schemes. In order to artificially dump spurious oscillations, we introduced a viscosity term in the non-linear Schrödinger equation by performing the following replacement:

\[
\nabla^2 \psi \rightarrow (1 - i\epsilon)\nabla^2 \psi,
\]

where \(\epsilon > 0\) quantifies the strength of the damping term. If we consider, as an example, a single plane-wave:

\[
\psi \propto \exp(i\omega t - ikx),
\]

the effect of such an artificial viscosity term is to damp the wave-function by a factor:

\[
\exp\left(-\frac{k^2\epsilon\ell}{2m}\right).
\]

In general, an artificial viscosity term would affect mass conservation. However, by solving the continuity equation on top of the non-linear Schrödinger equation, mass conservation is enforced and the artificial viscosity acts as a viscous force

\[
-\frac{\epsilon}{2m} \nabla \left(\frac{\nabla (\rho v)}{\rho}\right),
\]

in the momentum equation and it helps preventing high-frequency waves to build up in time. With the addition of such a viscosity term, we are able to evolve without any issues the wave-function over hundreds of oscillation periods in our tests.
and, at the same time, preserving mass, energy and agreement with analytical solutions. There is no unique prescription for solving these issues we encountered and, in general, it is possible to design more elaborate artificial viscosity terms.

I. Code units

We adopt the set of “super-comoving coordinates” introduced in [45] and already used in RAMSES. Thus, the following change of variables is performed:

$$\begin{align*}
\tilde{x} &= \frac{x}{aL}, \\
\tilde{t} &= \frac{H_0 dt}{a^2}, \\
\tilde{\psi} &= \psi / \tilde{\psi}, \\
\tilde{V} &= \frac{V a^2}{(H_0 L)^2},
\end{align*}$$

where $H_0$ is the Hubble constant, $L$ is the box size and $\tilde{\psi}$ is chosen to ensure that $\int |\tilde{\psi}|^2 \, d^{\text{dim}} x = 1$. As a consequence, the resulting non-linear Schrödinger equation reads:

$$i \frac{d\tilde{\psi}}{d\tilde{t}} + \frac{1}{2\tilde{m}} \tilde{V}^2 \tilde{\psi} - \tilde{m} \tilde{V} \tilde{\psi} - \tilde{g}|\tilde{\psi}|^2 \tilde{\psi} = 0. \tag{37}$$

This set of coordinates was specifically designed for cosmological applications. However, it can be used for any application by setting the scale factor $a$ to unity and replacing $H_0$ by a general inverse time scale $T^{-1}$. In the remainder of the paper all equations will be in these code units.

For the particular case of axion dark matter in a cosmological setting (see Eq. (10)) we have $\tilde{g} = 0$, $\tilde{V} \propto a^{-3/2}$ and the potential is determined via the Poisson equation:

$$\tilde{\nabla}^2 \tilde{V} = \frac{3}{2} a \left[ \Omega_{\text{axions}} (|\tilde{\psi}|^2 - 1) + \Omega_{\text{CDM}} (\tilde{\rho}_{\text{CDM}} - 1) + \Omega_{\text{baryons}} (\tilde{\rho}_{\text{baryons}} - 1) + \ldots \right], \tag{38}$$

where $\Omega_i$ is the fraction of the energy budget of our Universe that is in matter component $i$ (axions, baryons, Cold Dark Matter etc.) and the mean value of $\tilde{\rho}_i$ over the box is set to unity.

IV. TESTS OF THE CODE

In this section we present the numerical experiments we performed in order to test the main features of SCALAR. When testing the accuracy of our numerical schemes, we rely on three main tests: conservation of mass, energy and reproduction of analytical solutions.

1. Conservation of mass

The non-linear Schrödinger equation has the conserved quantity:

$$M = \int |\psi|^2 \, d^{\text{dim}} x,$$

which in the Mandelung formulation is just the total mass of the fluid. Mass and energy are not manifestly conserved by the main solver, therefore monitoring them is a useful test.

In simulations with no refinements and without enforcing mass conservation, we typically find the error on the conservation of mass on the order of (the prefactor is here for $\Delta x = 2^{-6}$):

$$\frac{\Delta M}{M} \sim 10^{-6} \left( \frac{t}{T} \right),$$

where $T$ is the oscillation period. When we allow for refinements, the situation is typically worse and it is not good enough for cosmological simulations. However, by solving the continuity equation on top of the Schrödinger equation, we observe an improvement on the conservation of mass up to:

$$\frac{\Delta M}{M} \sim 10^{-13} \left( \frac{t}{T} \right).$$

This does not change when we allow refinements and, even though the error grows linearly in time, it is good enough in order to perform cosmological simulations.

This is shown in Fig. (2), where we perform the sine wave test on the domain grid only, with a resolution of $N_{\text{cell}} = 2^6$ cells, corresponding to $\ell = 6$ and $\Delta x = 2^{-6}$, in one dimension. Details regarding the sine wave test are described in Sec. IV A.

2. Conservation of energy

Since we enforce mass conservation by solving the continuity equation, energy conservation is a better accuracy test for our code. By defining kinetic and potential energy as:

$$K = \frac{1}{2m} \int |\nabla \psi|^2 \, d^{\text{dim}} x,$$

$$W = \frac{1}{2} \int V_{\text{eff}} |\psi|^2 \, d^{\text{dim}} x,$$

the temporal change in the total energy of the system is expressed by:

$$\frac{d}{dt} (K + W) = \frac{1}{2} \int \frac{\partial V_{\text{eff}}}{\partial t} |\psi|^2 \, d^{\text{dim}} x. \tag{39}$$

As we can see, in the case where the effective potential has no explicit time-derivatives the energy $E = K + W$ will be conserved under the evolution.

In a cosmological setting, the potential $V$ depends on time via the scale-factor and this leads to a Lazer-Irvine equation [33]:

$$\frac{d}{dt} (K + W) - HW = 0,$$

which can be monitored by integrating it up while performing the simulation.
It evolves in a constant potential, which is defined as:

$$V = 1 - \frac{2\pi^2 n^2}{m^2},$$  \hspace{1cm} (41)$$

where $m$ is the mass carried by the wave-function and the period of oscillation is given by:

$$T = \frac{2\pi}{m}.\quad(42)$$

The full analytical solution reads:

$$\psi(x, t) = e^{-i\frac{2\pi}{T}t} \sin(2\pi n x).\quad(43)$$

This test is designed to test the creation of ghost cells when computing fine levels boundary conditions. Here we only refine according to the mass criterion, Eq. (29) and, since the density profile does not evolve in time, there is no dynamical creation or destruction of grids: once the refinement map is computed at the beginning, it does not change. We evolve the solution of the Schrödinger equation over 100 periods of oscillation. It is possible to show that the solution of the discretised equation is the same as Eq. (43), up to second order in space, but with a slightly different period of oscillation. Therefore, we correct the analytical solution by replacing $T$ with $T_{\text{discrete}}$, where:

$$\frac{T_{\text{discrete}}}{T} = \frac{1}{1 + T^2 n^2 \left(\frac{\cos(2\pi n \Delta x)}{(2\pi n \Delta x)^2} - \frac{1}{2}\right)}$$

$$\approx \frac{1}{1 - \frac{\pi^2 T^2 n^4}{12}\Delta x^2}.\quad(44)$$

This test was performed for $\ell_{\text{num}} = 1$. The non-linear Schrödinger equation is solved together with the continuity equation, in order to enforce conservation of mass. The domain grid resolution is $N_{\text{cell}} = 2^\ell$, corresponding to $\ell = 6$ and $\Delta x = 2^{-6}$, and the maximum refinement level is set to $\ell_{\text{max}} = 8$. When boundary conditions for fine-levels are needed, phase and density are interpolated in ghost cells by means of fourth-order conservative interpolation. We used artificial viscosity with $\epsilon = 0.2$. The results from this test are shown in Fig. (3).

We have also performed a similar test using a quadratic potential leading to a Gaussian profile:

$$\psi \propto \exp \left(-\frac{x^2}{\sigma^2}\right),$$

with very similar results.

### B. Travelling wave

This test simulates a one dimensional wave-packet travelling through a periodic box. Here, we test dynamical creation and destruction of grids, since the AMR hierarchy follows the density profile moving towards the direction of the wave. In this case, we have no potential and the initial conditions are defined as:

$$\psi(x, 0) = \frac{1}{\sqrt{2}} \left[ e^{i k_1 x} + e^{i k_2 x} \right],$$  \hspace{1cm} (45)$$
where $k_1 = 2\pi n_1$, $k_2 = 2\pi n_2$ with $n_1 \neq n_2 \in \mathbb{N}$. The oscillation frequency of a single mode is:

$$\omega(k) = \frac{k^2}{2m},$$

and the analytical solution of the Schrödinger equation reads:

$$\psi(x, t) = \frac{1}{\sqrt{2}} \left[ e^{i(k_1 x - \omega(k_1) t)} + e^{i(k_2 x - \omega(k_2) t)} \right].$$

As a consequence, the density is given by:

$$|\psi(x, t)|^2 = 1 + \cos \left( 2\pi (n_2 - n_1) + \frac{2\pi t}{T} \right),$$

where the oscillation period is defined as:

$$T = \frac{m}{\pi(n_1^2 - n_2^2)}.$$  

The wave-function is evolved in time over 100 oscillation periods. The non-linear Schrödinger equation is solved together with the continuity equation, in order to enforce conservation of mass. Also in this case, the coarse-fine data interpolation is made by fourth-order conservative interpolation. However, while density and phase are interpolated in ghost cells, new refinements are made by interpolating real and imaginary parts of the wave-function. The domain grid has the same resolution as in the previous test and refinements are allowed up to $\ell_{\text{max}} = 8$. We used artificial viscosity with $\epsilon = 0.2$. The results from this test are shown in Fig. (4).

C. Soliton

In a cosmological context, SCALAR can be used to simulate the structure formation process with Fuzzy Dark Matter. In this case, the density profiles of the dark matter halos differs from the case of the standard CDM. We can find a stationary solution which can be tested by taking:

$$\psi(x, t) = e^{-i\frac{2\pi}{T} \chi(x)},$$

and solving the resulting ODE for $\chi(x)$. A numerical fit to the density profile in three dimensions was in [46] found to be on the form

$$\rho \propto \frac{1}{(1 + (r/r_{\text{core}})^2)^n},$$

where $r_{\text{core}}$ can be chosen as a free parameter, see Appendix C for more details.

We set this density profile analytically and evolve the system. The density profile will remain approximately stationary while the wave-function oscillates as:

$$\psi(x, t) \propto e^{-i\frac{2\pi}{T}}.$$  

Since the system evolves under the effect of its own gravitational field, this case represents a good test for the Poisson equation. This test was performed for $\dim = 3$ and the non-linear Schrödinger equation is solved together with the continuity equation. The domain grid contains $N_{\text{cell}} = (2^6)^3$ cells, corresponding to $\ell = 6$ and $\Delta x = 2^{-6}$, and the maximum refinement level allowed is $\ell = 8$. In this case, both refinement and ghost cells are made by fourth-order conservative interpolation on density and phase. We did not need to include artificial viscosity. The results from this test are shown in Fig. (5).

V. CONCLUSIONS

In SCALAR, we implemented a set of numerical algorithms developed in order to solve the non-linear Schrödinger equa-
The evolution of the three errors as a function of time for the travelling wave test. While the error on the conservation of mass (green) evolves accordingly to $\Delta M/M \sim 10^{-13}(t/T)$, the error on the conservation of energy (purple) stays constant. Furthermore, the error with respect the analytical solution (blue) does not evolve over time.

The code will be soon publicly available through our GitHub repository.

ACKNOWLEDGMENTS

We thank the Research Council of Norway for their support. Computations were performed on resources provided by UNINETT Sigma2 – the National Infrastructure for High Performance Computing and Data Storage in Norway. This paper is partly based upon work from the COST action CA15117 (CANTATA), supported by COST (European Cooperation in Science and Technology). HAW was supported by the European Research Council through 646702 (CosTesGrav).

[1] J. Monroe and P. Fisher, Phys. Rev. D76, 033007 (2007), arXiv:0706.3019.

1 http://github.com/mattiamina
Therefore, the stability condition reads:
\[ |\beta| < 1. \]

\[ (A7) \]

As mentioned in the previous sections, in SCALAR we use a second-order finite difference formula in order to approximate the Laplacian of the wave-function. Thus, for a generic quantity, in one physical dimension we have:

\[ \nabla^2 f = \frac{f_{i+1} + f_{i-1} - 2f_i}{\Delta x^2}. \]

\[ (A2) \]

As a consequence, the second derivative of the error can be written as:

\[ \nabla^2 \epsilon = -\frac{4}{(\Delta x)^2} \sin^2 \left( \frac{k\Delta x}{2} \right) \epsilon, \]

\[ (A3) \]

and the amplification factor can be computed as:

\[ \xi = \frac{\epsilon(t + \Delta t)}{\epsilon(t)} = e^{-imV\Delta t} \left[ 1 + i\beta - \frac{\beta^2}{2!} - \ldots + \frac{(i\beta)^n}{n!} \right], \]

\[ (A4) \]

where \( \beta \) corresponds to:

\[ \beta = -\frac{2\hbar^2 \Delta t}{m(\Delta x)^2} \sin^2 \left( \frac{k\Delta x}{2} \right). \]

\[ (A6) \]

In order to avoid exponential growth, we require that \( |\xi| < 1 \). Therefore, the stability condition reads:

\[ |\xi|^2 = \cos^2_n(\beta) + \sin^2_n(\beta) < 1, \]

\[ (A7) \]
where \( \cos_n \) and \( \sin_n \) denote to \( n \)th order Taylor polynomials of \( \cos(x) \) and \( \sin(x) \), respectively. Furthermore, \( n \) corresponds to the order of the Taylor expansion of the kinetic contribution to the time evolution operator, Eq.\((20)\).

In particular, we find that for \( n < 3 \) the numerical scheme is unconditionally unstable. For \( n = 3 \), instead, the stability condition is satisfied as long as:

\[
|\beta| < \sqrt{3} \implies \Delta t < C_K \frac{\sqrt{3}}{2\Delta x} m (\Delta x)^2. \quad (A8)
\]

The generalisation to \( D \) can be done by replacing \( (\Delta x)^2 \to (\Delta x)^2 / D \) in the formula above.

We also require that the phase angle does not rotate more than \( 2\pi C_W \) within a time-step. Thus, for the kinetic term we require that:

\[
\Delta t < C_K \cdot \frac{\pi n (\Delta x)^2}{h}. \quad (A9)
\]

while for the potential term:

\[
\Delta t < C_W \cdot \frac{2\pi h}{m|V_{\text{max}}|}. \quad (A10)
\]

Combining the three conditions above, the optimal time-step is chosen as:

\[
\Delta t < \min \left[ \frac{C_K}{2\sqrt{3}h} \cdot m (\Delta x)^2, \frac{C_W}{m|V_{\text{max}}|} \right], \quad (A11)
\]

where we require the safety factors to be \( C_W, C_K < 1 \).

**Appendix B: Prolongation operators**

The details related to the interpolation schemes we implement in SCALAR are given for the one dimensional case. In case of multidimensional interpolation, the same formulas derived in this section are applied sequentially in each direction.

For simplicity, in order to derive the interpolation formulas we use in SCALAR, we assume that there is an odd number \( 2N + 1 \) of coarse data points \( \{x_i, y_i\} \) with \(-N \leq i \leq N\) and the interpolation is always done for children of the central cell, as shown in Fig.\((6)\).

In this way, we can construct an interpolating polynomial of order \( 2N \) starting from its general definition:

\[
P(x) = \sum_{n=-N}^{N} a_n x^n. \quad (B1)
\]

In order to find an explicit expression for \( P(x) \), the coefficients \( \{a_n\} \) have to be computed and, depending on the constraints imposed on the interpolating polynomial, different schemes can be implemented.

**Lagrange polynomial interpolation**

For this interpolation scheme, instead of explicitly computing the coefficients \( a_n \), the interpolating polynomial is expressed as a linear combination of Lagrange basis functions \( l_j(x) \), with \( j = -N, \ldots, N \), which are defined as:

\[
l_j(x) = \prod_{m=-N, m \neq j}^{N} \frac{x - x_m}{x_j - x_m}. \quad (B2)
\]

Thus, coarse data points \( y_j \) weight the Lagrange basis functions and the interpolating polynomial is constructed as follows:

\[
P(x) = \sum_{n=-N}^{N} y_n l_n(x). \quad (B3)
\]

In this way, the interpolating polynomial is forced to pass through the data points in the sample. In order to find an explicit formula for a fourth-order Lagrange interpolating polynomial, we directly compute the Lagrange basis functions by means of Eq.\((B2)\). Here, we express the differences between cell positions in Eq.\((B2)\) as a function of \( \Delta x \). Then, the interpolating polynomial is computed at children cell positions, leading to the following interpolation formulas:

\[
y_m = \begin{cases} 45 \frac{y_{-2}}{2048} - 420 \frac{y_{-1}}{2048} + 1890 \frac{y_0}{2048} - 252 \frac{y_{+1}}{2048} + 35 \frac{y_{+2}}{2048} & \text{if } m = -N \text{ or } N, \\ 35 \frac{y_{-2}}{2048} - 252 \frac{y_{-1}}{2048} + 1890 \frac{y_0}{2048} + 420 \frac{y_{+1}}{2048} - 45 \frac{y_{+2}}{2048} & \text{otherwise}, \end{cases} \quad (B4)
\]

\[
y_p = \begin{cases} 45 \frac{y_{-2}}{2048} + 420 \frac{y_{-1}}{2048} + 1890 \frac{y_0}{2048} - 252 \frac{y_{+1}}{2048} + 35 \frac{y_{+2}}{2048} & \text{if } m = -N \text{ or } N, \\ 35 \frac{y_{-2}}{2048} + 252 \frac{y_{-1}}{2048} - 1890 \frac{y_0}{2048} - 420 \frac{y_{+1}}{2048} + 45 \frac{y_{+2}}{2048} & \text{otherwise}. \end{cases} \quad (B5)
\]

**Conservative polynomial interpolation**

In this case, constraints on the coefficients of the interpolating polynomial are set by imposing that the mean of the interpolated data on fine cells is equal to the data stored in the coarse cell.

\[
\bar{y}_i = \frac{1}{\Delta x} \int_{\frac{-\Delta x}{2}}^{\frac{\Delta x}{2}} P(x) \, dx, \quad (B6)
\]

and the corresponding linear system is solved in order to de-
rives an explicit expression for the coefficients \( \{a_n\} \) of the interpolating polynomial. Thus, assuming that the parent cell is split into two children cells, the fine data is obtained by solving the following integrals:

\[
y_m = \frac{1}{\Delta x} \int_{-\Delta x}^{0} P(x) \, dx, \tag{B7}
\]

\[
y_p = \frac{1}{\Delta x} \int_{0}^{+\Delta x} P(x) \, dx, \tag{B8}
\]

where \( y_m \) and \( y_p \) denote the left and right children cells respectively.

For a fourth-order polynomial, the solution of the linear system reads:

\[
a = \frac{\tilde{y}_{-2} - 4\tilde{y}_{-1} + 6\tilde{y}_0 - 4\tilde{y}_{+1} + \tilde{y}_{+2}}{24(\Delta x)^4}, \tag{B9}
\]

\[
b = \frac{-\tilde{y}_{-2} + 2\tilde{y}_{-1} - 2\tilde{y}_{+1} + \tilde{y}_{+2}}{12(\Delta x)^3}, \tag{B10}
\]

\[
c = \frac{-\tilde{y}_{-2} + 12\tilde{y}_{-1} - 22\tilde{y}_0 + 12\tilde{y}_{+1} - \tilde{y}_{+2}}{16(\Delta x)^2}, \tag{B11}
\]

\[
d = \frac{5\tilde{y}_{-2} - 34\tilde{y}_{-1} + 34\tilde{y}_0 - 5\tilde{y}_{+2}}{48(\Delta x)}, \tag{B12}
\]

\[
e = \frac{9\tilde{y}_{-2} - 116\tilde{y}_{-1} + 2134\tilde{y}_0 - 116\tilde{y}_{+1} + 9\tilde{y}_{+2}}{1920}, \tag{B13}
\]

and the corresponding interpolation formulas are:

\[
y_m = \tilde{y}_0 - \frac{3\tilde{y}_{-2} - 22\tilde{y}_{-1} + 22\tilde{y}_{+1} - 3\tilde{y}_{+2}}{128}, \tag{B14}
\]

\[
y_p = \tilde{y}_0 + \frac{3\tilde{y}_{-2} - 22\tilde{y}_{-1} + 22\tilde{y}_{+1} - 3\tilde{y}_{+2}}{128}. \tag{B15}
\]

It is trivial to check the arithmetic average of \( y \) over the children cells corresponds exactly to the value stored in the parent cell.

### Appendix C: Soliton solutions

Self-gravitating bosonic fields can support stable and localised field configurations, where the density profile is static. Such configurations, called solitons, are ubiquitous in models of axion dark matter and exist for \( d > 1 \). Starting with the Schrödinger-Poisson system (in code units):

\[
\frac{i}{\hbar} \frac{\partial \psi}{\partial t} = -\frac{1}{2m} \nabla^2 \psi + mV, \quad \nabla^2 V = \kappa |\psi|^2, \tag{C1}
\]

we take the spherically symmetric ansatz:

\[
\psi(r, t) = \chi(r)e^{-2\pi i \frac{r}{D}}, \tag{C2}
\]

The ODE describing the static configuration of the field can be derived by replacing Eq. (C2) into the Schrödinger-Poisson system:

\[
\nabla^2 \left[ \frac{\nabla^2 \chi}{\chi} \right] = 2\kappa m^2 \chi^2, \tag{C3}
\]

where the Laplacian is now expressed in polar coordinates:

\[
\nabla^2 = \frac{d^2}{dr^2} + \frac{2}{r} \frac{d}{dr},
\]

and \( \kappa \) represents a free parameter (in our code units \( \kappa \) equals \( \frac{2}{\Omega_\text{m}a^2} \) for a cosmological simulation with axion-like dark matter). Thus, the system can be solved by considering the initial conditions:

\[
\chi(0) = 1, \quad \chi'(0) = 0, \quad \chi''(0) = 0,
\]

where \( \chi''(0) \) is a free parameter, which is set by requiring asymptotic vanishing solution \( \chi(\pm \infty) = 0 \). The oscillation period \( T \) can then be computed from the resulting solution, for more details see [46].

The soliton profile, normalised such that the total mass is unity, is given by:

\[
\chi(r) = \frac{32}{\sqrt{33} \pi r \gamma} \left( 1 + \left( \frac{r}{r_{\text{core}}} \right)^2 \right)^4, \tag{C4}
\]

where \( r \) is the distance from the center of the box and:

\[
m = \sqrt{\frac{33}{r_{\text{core}} \sqrt{\kappa} \alpha^2}}, \quad T = \sqrt{\frac{33 \pi r_{\text{core}} \kappa}{16 \gamma^2}}, \tag{C5}
\]

with \( \alpha = 0.230 \) and \( \gamma = -0.692 \). In our tests, even if this derivation assumes an infinite box, we use either fixed or periodic boundary conditions. As long as the core radius of the soliton is small enough compared to the simulation box, this solution represents a good approximation.