Extension of the continuum time-dependent Hartree-Fock method to proton states

C. I. Pardi[*] and P. D. Stevenson[†]

Department of Physics, University of Surrey, Guildford, Surrey GU2 7XH, United Kingdom

K. Xu[‡]

Mathematical Institute, University of Oxford, Oxford, Oxfordshire OX1 3LB, United Kingdom

This paper deals with the solution of the spherically symmetric time-dependent Hartree-Fock approximation applied to nuclear giant monopole resonances in the small amplitude regime. The problem is spatially unbounded as the resonance state is in the continuum. The practical requirement to perform the calculation in a finite-sized spatial region yields an artificial boundary, which is not present physically. The question of how to ensure the boundary does not interfere with the internal solution, while keeping the overall calculation time low is studied. Here we propose an absorbing boundary condition scheme to handle the conflict. The derivation, via a Laplace transform method, and implementation is described. An inverse Laplace transform required by the absorbing boundaries is calculated using a method of nonlinear least squares. The accuracy and efficiency of the scheme is tested and results presented to support the case that they are an effective way of handling the artificial boundary.

I. INTRODUCTION

As a quantum system the behavior of a nucleus over a period of time obeys the N-particle time-dependent Schrödinger equation (TDSE). Solving the full many-body TDSE analytically or even numerically is generally not tractable. However, approximate solutions can be gained by solving the time-dependent Hartree-Fock (TDHF) equations [1,2]. The simplification still does not allow analytic solutions, but numerical techniques can be applied and the computational cost kept manageable.

The TDHF equations are a coupled set of initial-boundary-value problems for which it is common to apply finite differencing methods in both spatial and temporal coordinates. In this scheme the equations can be solved by, e.g., a series of matrix inversions. One difficulty with these types of computational solution is the limitation of calculating wave functions in a finite spatial region, which introduces an artificial boundary into calculations. Appropriate conditions for the boundary have to be chosen. In cases where the system can be fully contained in a region for all time, the values at the boundary can simply and correctly be set to zero. However, in many situations particles are emitted from a system into the continuum. This is common in the case of giant resonances, as most are above the particle decay threshold [3]. These particles move off into the continuum physically, yet computationally they will reach the artificial boundary [4,5].

The most crude, and simple, way of tackling this problem is to use reflecting boundaries, which rebound any matter that comes into contact with them. A fully accurate solution with these boundary conditions can occur in the case that the reflecting boundary is sufficiently distant from the original nucleus that emitted flux does not reach the boundary within the simulation time. Although such reflecting boundaries are easy to implement, in some cases the large spatial domain required results in inefficient calculations [6,7].

More efficient solutions have been sought in the form of absorbing potentials. These attempt to use reflecting boundaries in a sensibly sized region, and then use a complex potential to remove matter that approaches the boundary. This can prevent reflections taking place and works well in some situations [7]. However, in general, a given absorbing potential will not work with perfect efficiency at all frequencies, and these potentials may still require considerable spatial extent to work very well [6,8].

Here we present a method of implementing absorbing boundary conditions (ABCs) [9]. These rely on choosing the artificial boundary such that the potential outside of it has a simple form. The resulting equations in the exterior can be manipulated into a boundary condition for the interior problem which can be applied closer to the initial bulk of matter. The propagation of waves in the exterior region then does not have to be dealt with explicitly. In solving the TDHF equations, a simplified Skyrme interaction is used in the implementation which reproduces the magic numbers needed for $^{4}_{2}\text{He}$, $^{16}_{8}\text{O}$, and $^{40}_{20}\text{Ca}$ to be seen without the complexity of the full interaction [10], as a reasonable proof of concept. Spherical symmetry is also assumed inside and outside of the artificial boundary. The calculations involve various forms of differential equations, each of which require their own absorbing boundary conditions. Here a continuous absorbing boundary condition is implemented [9] which improves on previous work [11] by accounting for the long-ranged Coulomb potential. The previous work is modified by approximation of the required inverse Laplace transform via the use of a nonlinear least squares method [12].

The structure of this paper is as follows. Section II gives a brief summary of the types of giant resonance and their properties. The theory and discretization for the time-dependent Hartree-Fock approach is described in Sec. III. Section IV describes the absorbing boundary conditions, the nonlinear least squares method, and their application to TDHF. Testing of the ABCs implementation is given in Sec. V

---

[*]chris_pardi@yahoo.co.uk

[†]p.stevenson@surrey.ac.uk

[‡]kuan.xu@maths.ox.ac.uk

and results of the TDHF with ABC calculations are given in Sec. VI.

II. GIANT MONOPOLE RESONANCES

Giant monopole resonances (GMRs) are collective excitations of the nucleus, meaning most if not all particles are involved in the excitation [3]. They are well studied experimentally, being first observed in 1977 [13] and their study has continued to the present day [14–16]. Excitation of the monopole resonance is commonly performed with α scattering [14]. The requirement for angular momentum conservation excludes the possibility of excitation by a photon, as is performed for the dipole resonant mode [17]. Aside from shedding light on the structure of individual nuclei, further interest in GMRs is provided by their relation to the incompressibility of nuclear matter and the consequent light they shed on the equation of state with consequent importance in understanding neutron stars, supernovae explosions, and heavy-ion collision [18].

Our main interest in this phenomenon, however, is owing to the simplified analysis they allow for. Specifically they are a purely radial excitation and hence by considering only the subset of doubly magic nuclei we are able take advantage of spherical symmetry in the calculations. As is common when developing new methods a simplified Skyrme potential, containing just the t_0 and t_3 terms, is used [10,19–23]. As was commented on previously [11] this cannot be expected to give a detailed comparison with experiment, but is used to demonstrate the features of the new method.

The key quantity for comparison to experiment is the strength function, which can be related to the experimental cross section. This quantity has been noted to be particularly sensitive to the boundary conditions applied to the TDHF equations [6]. Therefore, we shall measure success as the accurate reproduction of this quantity, free of artifacts that may arise from the boundary conditions.

III. TDHF

Originating with a formulation by Dirac [24], the time-dependent Hartree-Fock method became practical for realistic calculations in nuclei only with the advent of sufficiently advanced computational facilities [25–27]. It has been widely applied to heavy-ion collisions and giant resonances, as well as selected other problems. A recent review [28] covers many such applications.

A. Theory

The TDHF method relies on the time-dependent variational principle in which the action, defined as

$$S[\Psi(t)] = \int_{t_0}^{t_1} \langle \Psi(t) | i\hbar \frac{\partial}{\partial t} - \hat{H} | \Psi(t) \rangle \, dt, \tag{1}$$

is minimized. If one considers a trial wave function $|\Psi(t)\rangle$ belonging to a general Hilbert space it can be shown that the Schrödinger equation is retrieved upon minimizing the above. The TDHF method considers a trial wave function in a restricted space of antisymmetric Slater determinants [29],

given in the spatial-spin-isospin basis as

$$\Psi^{(A)}(\vec{z}_1,\ldots,\vec{z}_N,t) = \frac{1}{\sqrt{N!}} \begin{vmatrix} \phi_1(\vec{z}_1,t) & \cdots & \phi_1(\vec{z}_N,t) \\ \vdots & \ddots & \vdots \\ \phi_N(\vec{z}_1,t) & \cdots & \phi_N(\vec{z}_N,t) \end{vmatrix}. \tag{2}$$

The coordinate $\vec{z}_i = (\vec{r}_i, \sigma_i, \tau_i)$ describes spatial, spin, and isospin degrees of freedom. The wanted result from minimizing in this space of restricted wave functions is to produce a numerically tractable problem. The Hamiltonian $\hat{H}$ for nuclear calculations is accepted to contain a kinetic operator and two and three body operators that describe the potential [1,30,31]. In the spatial-spin-isospin basis this takes the form

$$\hat{H}(\vec{z}_1,...,\vec{z}_N) = -\frac{\hbar^2}{2m} \sum_{i=1}^{N} \nabla_i^2(\vec{r}_i) + \sum_{i=1}^{N} \sum_{j=1}^{i-1} \hat{v}_{ij}^{(2)}(\vec{z}_i,\vec{z}_j)$$
$$+ \sum_{i=1}^{N} \sum_{j=1}^{i-1} \sum_{k=1}^{j-1} \hat{v}_{ijk}^{(3)}(\vec{z}_i,\vec{z}_j,\vec{z}_k). \tag{3}$$

In this work we shall use the simplified t_0-t_3 Skyrme interaction for the nuclear components of the potential and the electrostatic interaction for the Coulomb component. This yields the two body potential as [32,33]

$$v_{ij}^{(2)}(\vec{r},\vec{r}') = t_0 \delta(\vec{r} - \vec{r}') + \frac{\eta}{|\vec{r} - \vec{r}'|} P_{i,j}, \tag{4}$$

where $\eta = \frac{e^2}{4\pi\epsilon_0} \approx 1.44 e^2$ MeV^{-1} fm^{-1} and P_{ij} is zero if i and/or j is a neutron and one if i and j are protons. The three body potential is given as [32,33]

$$v_{ijk}^{(3)}(\vec{r},\vec{r}',\vec{r}'') = t_3 \delta(\vec{r} - \vec{r}') \delta(\vec{r}' - \vec{r}''). \tag{5}$$

The values $t_0 = -1090.0$ MeV fm^3 and $t_3 = 17288.0$ MeV fm^6 are used [11]. Performing the minimization of the action with the Hamiltonian as described above can be shown to produce the following set of equations for the reduced radial single particle wave functions [10,25]:

$$i\hbar \frac{\partial Q_{n,l}(r,t)}{\partial t} = \hat{H}_{\text{HF}} Q_{n,l}(r,t), \tag{6}$$

where the Hartree-Fock Hamiltonian is given as

$$\hat{H}_{\text{HF}} = \left[-\frac{\hbar^2}{2m} \frac{\partial}{\partial r^2} + V(r,t,\rho_{\text{n}},\rho_{\text{p}}) + \frac{\hbar^2}{2m} \frac{l(l+1)}{r^2} \right]. \tag{7}$$

The above equations are subject to the boundary conditions

$$Q_{n,l}(0,t) = 0, \tag{8}$$

$$Q_{n,l}(r,t) \to 0, \quad \text{as} \quad r \to \infty, \tag{9}$$

and an initial condition, described later. The spatial part of the three-dimensional single particle wave functions can be retrieved from the above from

$$\phi(\vec{r},t) = \frac{Q_{n,l}(r,t)}{r} Y_l^m(\theta,\varphi), \tag{10}$$

where $Y_l^m(\theta,\varphi)$ is a spherical harmonic and l and m are the orbital and magnetic quantum numbers, respectively. When

calculating a neutron single particle wave function the potential $V(r,t,\rho_n,\rho_p)$ is equal to

$$V_n(r,t,\rho_n,\rho_p) = t_0\left(\rho_p + \frac{1}{2}\rho_n\right) + \frac{t_3}{4}\rho_p(\rho_p + 2\rho_n), \quad (11)$$

and when calculating a proton

$$V_p(r,t,\rho_n,\rho_p) = t_0\left(\rho_n + \frac{1}{2}\rho_p\right)$$
$$+ \frac{t_3}{4}\rho_n(\rho_n + 2\rho_p) + V_c(r,t). \quad (12)$$

The densities are given by

$$\rho(r,t) = \frac{1}{4\pi r^2}\sum_{(n,l)\in S}(2l+1)|Q_{n,l}(r,t)|^2, \quad (13)$$

where the sets S_n and S_p replace S, in the above, for the neutron and proton densities, ρ_n and ρ_p, respectively. The sets S_p and S_n contain the values of (n,l) for the protons and neutrons within the system. The particular (n,l) values we take for each nuclei are shown in Table I.

The Coulomb potential $V_c(r,t)$ can be found by solving the following differential equation:

$$\frac{\partial^2 W_c(r,t)}{\partial r^2} = -4\pi\eta r\rho_p(r,t), \quad (14)$$

for W_c, subject to the boundary conditions

$$W_c(0,t) = 0, \quad (15)$$

$$\left.\frac{\partial W_c(r,t)}{\partial r}\right|_{r=R_{\text{Coul}}} = 0, \quad (16)$$

where $\rho_p = 0$ for $r \geqslant R_{\text{Coul}}$. The Coulomb potential can then be calculated from $W_c(r,t)$ via

$$V_c(r,t) = \frac{W_c(r,t)}{r}. \quad (17)$$

It is noted that the minimization also produces an exchange term for the Coulomb potential which, for simplicity, is excluded in this analysis. Extension of the method to include the full Coulomb exchange would not be straightforward, partly due to the extra long-ranged component to the fields, but also due to the practice of making the difficult Coulomb exchange practical by using an oscillator basis [34,35].

TABLE I. Table showing the explicit elements of the set S_n and S_p for the nuclei considered here.

Nucleus	$(n,l) \in S_n$ or S_p	$2l+1$
Helium-4	(0,0),(0,0)	1
Oxygen-16	(0,0),(0,0)	1
	(0,1),(0,1)	3
Calcium-40	(0,0),(0,0)	1
	(1,0),(1,0)	1
	(0,1),(0,1)	3
	(0,2),(0,2)	5

The initial condition

The initial condition in these calculations is defined to be the result from applying a boost operator on the ground state

$$\Psi(\vec{z_1},\ldots,\vec{z_N},t=0) = e^{i\epsilon r^2}\Psi_0(\vec{z_1},\ldots,\vec{z_N}). \quad (18)$$

The ground state $\Psi_0(\vec{z_1},\ldots,\vec{z_N})$ is found using the time-independent Hartree-Fock method in which the energy is minimized in a space of Slater determinants to produce

$$\hat{H}_{\text{HF}}Q_{n,l}(r) = E_{n,l}Q_{n,l}(r), \quad (19)$$

which relies on the time-independent equivalents of Eqs. (7)–(17).

B. Numerical procedure

Equations (6), (14), and (19) are all solved numerically by finite difference methods. So the following discrete spatial variable is defined:

$$r_m \equiv m\Delta r, \quad \Delta r = \frac{R}{M}, \quad (20)$$

where $m = 1,2,\ldots,M$. Time is chosen to be discretized by the equidistant set of points,

$$t_n = n\Delta t, \quad \Delta t = \frac{T}{N}, \quad (21)$$

where $n = 0,1,\ldots,N$.

We use the methods described previously [11] to calculate the ground state and time-dependent wave function on the spatial and temporal grid. Linear equations are produced for the stationary case through use of a self-consistent scheme, which can be expressed as a series of matrix eigenvalue problems by using central differences. An intermediate step via the evolution operator is used to produce linear equations in the time-dependent case, which are then discretized in time using the Crank-Nicholson scheme [36] and space using central differences, producing a series of matrix inversions. In practice the LAPACK subroutines [37] are used to solve the matrix equations.

IV. BOUNDARY CONDITIONS

In this section we discuss the method of treating the TDHF equations in the continuum. The first part of the section discusses the derivation of an absorbing boundary condition, applicable to nuclear calculations. This will be seen to require the inverse Laplace transform of a kernel. A nonlinear least squares approach [12,38,39] is then described to provide an accurate approximate of the kernel by a sum of poles, whose inversion can be found in tables [40,41]. Finally, discretization of the absorbing boundary condition for use with the Crank-Nicholson scheme is described.

A. The problem in the exterior

Application of absorbing boundary conditions require us to split the domain into two regions: an interior and an exterior [9,42]. In the nuclear problem we have two types of potential: the short-ranged Skymre potentials and the long-ranged Coulomb and centrifugal potentials. We choose

to split our domain so that the potential in the exterior contains just the long-ranged components.

In order to ensure the Skyrme potentials are only present in the interior, we assume that the density is zero in the exterior. So in our analysis we make the assumption

$$\rho_p = \rho_n = 0, \tag{22}$$

for $r \geqslant R$. Of course this will only be approximately true, as some of the density will move into the exterior during the calculation. However, as was shown previously in the linear regime the effect is not detrimental to the results [11]. Using the above we may write the potentials (11) and (12) in the exterior as

$$V_n(r,\rho_n,\rho_p,t) = 0, \tag{23}$$

$$V_p(r,\rho_n,\rho_p,t) = V_c(r,t). \tag{24}$$

Assumption (22) and Gauss' law also allows for the Coulomb potential to be simplified to

$$V_c(r) = \frac{\eta N_p}{r}. \tag{25}$$

Assuming elementary charge units, N_p is the number of protons. Using Eqs. (23), (24), and (25) with Eq. (6) we can write the following form of equation that all general single particle wave functions obey in the exterior:

$$i\hbar \frac{\partial Q(r,t)}{\partial t} = -\frac{\hbar^2}{2m} \frac{\partial^2 Q(r,t)}{\partial r^2} + \left(\frac{\sigma}{r} + \frac{\hbar^2}{m} \frac{l(l+1)}{2r^2} \right) Q(r,t), \tag{26}$$

which is subject to the boundary condition

$$\lim_{r \to \infty} Q(r,t) = 0. \tag{27}$$

Within Eq. (26) we use

$$\sigma = \begin{cases} \eta N_p, & \text{for protons} \\ 0, & \text{for neutrons.} \end{cases} \tag{28}$$

The n,l values are kept implicit in the notation for Q as we are considering the form of a general single particle wave function. We continue by deriving some absorbing boundary conditions for Schrödinger-like equations that have the above form in an exterior region.

B. Absorbing boundary conditions

Equation (26) may be written more compactly by letting $t \to \frac{m}{\hbar} t$ and $\sigma \to \frac{m}{\hbar^2} \sigma$, producing

$$i \frac{\partial Q(r,t)}{\partial t} = -\frac{1}{2} \frac{\partial^2 Q(r,t)}{\partial r^2} + \left(\frac{\sigma}{r} + \frac{l(l+1)}{2r^2} \right) Q(r,t). \tag{29}$$

We now recall the definition of the Laplace transform,

$$\hat{f}(s) = \int_0^\infty f(t)e^{-st}\,dt \tag{30}$$

and its inverse, the Bromwich integral [43],

$$f(t) = \frac{1}{2\pi i} \int_{c-i\infty}^{c+i\infty} \hat{f}(s)e^{st}\,ds. \tag{31}$$

c is chosen such that the poles of $\hat{f}(s)$ are to the left of the contour. The hat notation is now used to imply the Laplace transform of a function. We proceed by multiplying Eq. (29) by e^{-st} and integrating in time from 0 to ∞, to get the differential equation for the Laplace transform of $Q(r,t)$,

$$\frac{1}{2} \frac{\partial^2 \hat{Q}(r,s)}{\partial r^2} + \left(is - \frac{\sigma}{r} - \frac{l(l+1)}{2r^2} \right) \hat{Q}(r,s) = 0. \tag{32}$$

The above is simplified by assuming the initial condition is zero in the exterior region. This is not restrictive for our needs because the nuclear wave function is localized around the origin. Letting $z = br\sqrt{s}$, where $b = -2i\sqrt{2i}$ and choosing the square root to be on the branch having a positive real part, produces

$$\frac{\partial^2 \hat{Q}(r,s)}{\partial z^2} + \left(-\frac{1}{4} + \frac{\kappa(s)}{z} + \frac{\frac{1}{4} - \mu^2}{z^2} \right) \hat{Q}(r,s) = 0, \tag{33}$$

where

$$\kappa(s) = -\sigma \sqrt{\frac{i}{2s}}, \tag{34}$$

$$\mu = l + \frac{1}{2}. \tag{35}$$

Equation (33) has Whittaker M and W functions as a satisfactory pair of solutions [44] meaning the general solution is

$$\hat{Q}(r,s) = AM_{\kappa,\mu}(z) + BW_{\kappa,\mu}(z). \tag{36}$$

As the Laplace transform of boundary condition (27) is evaluated at infinity, its application can be achieved by inspection of an appropriate asymptotic series. Assuming $c > 0$ in the Bromwich integral implies that $-\frac{1}{2}\pi < \arg z = \arg br\sqrt{s} < 0$ along the integration path, so the following equations are valid [44] for $z \to \infty$:

$$M_{\kappa,\mu}(z) \sim \frac{\Gamma(1+2\mu)}{\Gamma\left(\frac{1}{2}+\mu-\kappa\right)} z^{-\kappa} e^{(1/2)z} {}_2F_0$$
$$\times \left(\frac{1}{2}+\mu+\kappa, \frac{1}{2}-\mu+\kappa, \frac{1}{z} \right)$$
$$+ \frac{\Gamma(1+2\mu)}{\Gamma\left(\frac{1}{2}+\mu+\kappa\right)} z^{\kappa} e^{-(1/2)z} e^{\pi i[\kappa-\mu-(1/2)]} {}_2F_0$$
$$\times \left(\frac{1}{2}+\mu-\kappa, \frac{1}{2}-\mu-\kappa, -\frac{1}{z} \right) \tag{37}$$

and

$$W_{\kappa,\mu}(z) \sim z^{\kappa} e^{-(1/2)z} {}_2F_0\left(\frac{1}{2}+\mu-\kappa, \frac{1}{2}-\mu-\kappa, -\frac{1}{z} \right), \tag{38}$$

where

$${}_2F_0(a_1,a_2,z) = \sum_{n=0}^{\infty} \frac{(a_1)_n(a_2)_n}{n!} z^n. \tag{39}$$

The Pochhammer notation $(a)_n \equiv a(a+1)(a+2)\cdots(a+n-1)$ with $(a)_0 = 1$ has been used. The dominant terms in Eqs. (37) and (38) are the exponential functions $e^{(1/2)z}$ and

$e^{(-1/2)z}$, respectively. As $\mathrm{Re}(z) > 0$ along the integration path then $z \to \infty$ as $r \to \infty$, so we must enforce $A = 0$, in (36), in order for the boundary condition to be satisfied. So

$$\hat{Q}(r,s) = B\,W_{\kappa,\mu}(br\sqrt{s}). \qquad (40)$$

Division of the above by its derivative and rearranging produces

$$\hat{Q}(r,s) = \frac{1}{b\sqrt{s}} \left(\frac{W_{\kappa,\mu}(br\sqrt{s})}{\frac{\partial W_{\kappa,\mu}(br\sqrt{s})}{\partial r}} \right) \frac{\partial \hat{Q}(r,s)}{\partial r}. \qquad (41)$$

Use of the convolution theorem [43] and evaluating the result on $r = R$ yields the absorbing boundary condition

$$Q(R,t) = \int_0^t G_{\kappa,\mu}(R,\tau) \left. \frac{\partial Q(r,t-\tau)}{\partial r} \right|_{r=R} d\tau, \qquad (42)$$

where

$$\hat{G}_{\kappa,\mu}(R,s) = \frac{1}{b\sqrt{s}} \left. \left(\frac{W_{\kappa,\mu}(br\sqrt{s})}{\frac{\partial W_{\kappa,\mu}(br\sqrt{s})}{\partial r}} \right) \right|_{r=R}. \qquad (43)$$

Once the inverse Laplace transform has been calculated to yield $G_{\kappa,\mu}(R,t)$ from $\hat{G}_{\kappa,\mu}(R,s)$, Eq. (42) can be discretized for use with the Crank-Nicholson scheme described in Sec. III B. We also note that (42) is nonlocal, meaning it depends on wave-function information from previous times, which will be seen to have consequences for its numerical implementation described later. To proceed to find the inverse Laplace transform an implementation of a nonlinear least squares method is used.

C. Laplace inversion of the kernels

Previously [11] we relied on deriving a partial fraction representation, containing a finite number of terms, for which there is a known inversion. Surveying the literature [40,44,45] it can be seen that the same technique cannot be applied to the kernel (43). Finding an exact inversion, then, appears unlikely. However, if we were to have an accurate approximate of the kernel given as a sum of some partial fractions, then an analytic inversion of the approximation could be performed. This can be achieved via a method of nonlinear least squares [46], where the mean square error,

$$\int_a^b \left| \frac{P_d(z)}{Q_d(z)} - f(z) \right|^2 dz, \qquad (44)$$

between a rational function, $\frac{P_d(z)}{Q_d(z)}$, and a kernel function, $f(z)$, is minimized. $P_d(z)$ and $Q_d(z)$ are polynomials of degree $d - 1$ and d, respectively, and a and b are two purely imaginary numbers. The particular values of a and b will be chosen later, when the nonlinear leastsquares method is tested in Sec. V A. The rational function can be expressed as the sum of poles,

$$\frac{P_d(z)}{Q_d(z)} = \sum_{k=1}^d \frac{w_k}{z - z_k}. \qquad (45)$$

The Laplace inversion of the above is known to be the sum of exponentials [40],

$$\mathcal{L}^{-1} \left\{ \frac{P_d(z)}{Q_d(z)} \right\} = \sum_{k=1}^d w_k e^{z_k \tau}. \qquad (46)$$

Finding an inverse Laplace transform is then reduced to calculating the values of the kernel function. However, for a Schrödinger equation's ABC kernel it was shown [38,39] that the method described in [46] could not be applied directly. This is due to the fact that more poles than can be calculated accurately with a numerical implementation of [46] are required to approximate it. As the kernel studied here also results from a Schrödinger equation, we expect the same to occur and so the modified bootstrap procedure given in [12,38,39] is used. We proceed by describing the nonlinear least squares method, before explaining how it is embedded into a bootstrap procedure.

1. Method of nonlinear least squares

We require a method for finding polynomials $P_d(z)$ and $Q_d(z)$, where $d = \deg P_d(z) + 1 = \deg Q_d(z)$, such that

$$\int_a^b \left| \frac{P_d(z)}{Q_d(z)} - f(z) \right|^2 dz \qquad (47)$$

is minimized. As in the solution to the stationary Hartree-Fock equation, self-consistency is used to linearize the problem and produce

$$\int_a^b \left| \frac{P_d^{(i+1)}(z) - f(z)Q_d^{(i+1)}(z)}{Q_d^{(i)}(z)} \right|^2 dz, \qquad (48)$$

where $i \geqslant 1$ and

$$P_d^{(i+1)}(z) = \sum_{j=0}^{d-1} p_j^{(i+1)} z^j, \qquad (49)$$

$$Q_d^{(i+1)}(z) = z^d + \sum_{j=0}^{d-1} q_j^{(i+1)} z^j. \qquad (50)$$

The scheme in Eq. (48) requires an initial guess, $Q_d^{(1)}(z)$, which we will describe later. It is hoped as we iterate through i finding a minimum of (48), then the differences between the values of $\frac{P_d^{(i+1)}(z)}{Q_d^{(i+1)}(z)}$ and $f(z)$ become small.

To minimize Eq. (48), $2d$ freedoms are introduced for the coefficients of $P_d^{(i+1)}(z)$ and $Q_d^{(i+1)}(z)$ which can be shown to produce the equations

$$\int_a^b \bar{z}^n \frac{P_d^{(i+1)}(z) - f(z)Q_d^{(i+1)}(z)}{\left| Q_d^{(i)}(z) \right|^2} dz = 0, \qquad (51)$$

$$\int_a^b \bar{z}^n \bar{f}(z) \frac{P_d^{(i+1)}(z) - f(z)Q_d^{(i+1)}(z)}{\left| Q_d^{(i)}(z) \right|^2} dz = 0, \qquad (52)$$

where $n = 1, \ldots, d$, as sufficient conditions for a minimum. Defining the weighted inner product

$$\langle f | g \rangle = \int_a^b \frac{\bar{f}(z)g(z)}{\left| Q_d^{(i)}(z) \right|^2} dz \qquad (53)$$

and the basis

$$
h_n(z) = \begin{cases} z^{(n-1)/2} f(z), & n = 1,3,\ldots,2d+1 \\ z^{(n/2)-1}, & n = 2,4,\ldots,2d \end{cases}, \quad (54)
$$

allows Eqs. (51) and (52) to be written simply as

$$
\langle h_n | - P + f Q \rangle = 0, \quad (55)
$$

for $n = 1,\ldots,2d$. We see from the above that the numerator of (48) is orthogonal to the first $2d$ elements of the basis (54). By inspection we can see that the numerator is also a linear combination of the entire basis. So orthogonalizing the $2d + 1$ functions in (54) will result in $-P(z) + f(z)Q(z)$ being the member of the resultant orthogonal basis that is in the span of $h_{2d+1}(z)$.

The restatement of this problem means we can apply the Gram-Schmidt process. This takes any set of linearly independent functions and produces a set of orthogonal functions, $g_n(z)$. The first two orthogonal functions given by the Gram-Schmidt process are

$$
g_1(z) = h_1(z), \quad (56)
$$

$$
g_2(z) = h_2(z) - \frac{\langle g_1 | h_2 \rangle}{\langle g_1 | g_1 \rangle} g_1(z). \quad (57)
$$

Now instead of proceeding by orthogonalizing the set $\{h_1(z),h_2(z),\ldots,h_{2d+1}(z)\}$, we take advantage of the Gram-Schmidt process being applicable to any set of linearly dependent functions. The basis holds the property $h_n(z) = zh_{n-2}(z)$ allowing us to continue by orthogonalizing the set $\{h_1(z),h_2(z),zg_1(z),zg_2(z),\ldots,zg_{2d-1}(z)\}$, meaning for $n > 2$

$$
g_n(z) = zg_{n-2}(z) - \sum_{j=1}^{n-1} \frac{\langle g_j | zg_{n-2} \rangle}{\langle g_j | g_j \rangle} g_j(z). \quad (58)
$$

This simplifies the orthogonalization and results in a recursion in terms of just $g_n(z)$ with two initial values. It can be shown that $\{h_1(z),h_2(z),zg_1(z),\ldots,zg_{n-2}(z)\}$ spans the same space as $\{h_1(z),h_2(z),\ldots,h_n(z)\}$ for all $n \geqslant 3$ via induction [47].

A final simplification is made by considering the value of the quantity $\langle g_j | zg_{n-2} \rangle$ along the integration path, where

$$
\langle g_j | zg_{n-2} \rangle = \int_{\mathrm{Im}(a)}^{\mathrm{Im}(b)} \frac{\bar{g}_j(iy) \left[iyg_{n-2}(iy) \right]}{\left| Q_d^{(i)}(iy) \right|^2} i\,dy
$$

$$
= \int_{\mathrm{Im}(a)}^{\mathrm{Im}(b)} \frac{-\overline{[iyg_j(iy)]}\, g_{n-2}(iy)}{\left| Q_d^{(i)}(iy) \right|^2} i\,dy
$$

$$
= -\langle zg_j | g_{n-2} \rangle. \quad (59)
$$

By insertion of Eq. (58), the quantity on the right-hand side of the above can then be shown to be the following:

$$
\overline{\langle zg_j | g_{n-2} \rangle}
$$
$$
= -\left(\langle g_{n-2} | g_{j+2} \rangle + \sum_{k=1}^{j+1} \frac{\langle g_k | zg_j \rangle}{\langle g_k | g_k \rangle} \langle g_{n-2} | g_k \rangle \right). \quad (60)
$$

By the orthogonality of the functions $g_n(z)$ we see, from the equations above, that

$$
\langle g_j | zg_{n-2} \rangle = 0, \quad (61)
$$

for $j < n - 4$. The index of the sum in Eq. (58) now runs from $n - 5$ to $n - 1$. Reexpressing the sum so that the index runs from 1 to 4 means the Gram-Schmidt process can be summarized as

$$
g_n(z) = \begin{cases} f(z), & n = 1 \\ 1 - c_{21}g_1(z), & n = 2 \\ zg_{n-2}(z) - \sum_{j=1}^{\min(n-1,4)} c_{nj}g_{n-j}(z), & n \geqslant 3, \end{cases}
$$
$$
(62)
$$

where

$$
c_{nj} = \begin{cases} \dfrac{\langle f | 1 \rangle}{\langle f | f \rangle}, & n = 2, \; j = 1 \\ \dfrac{\langle g_{n-j} | zg_{n-2} \rangle}{\langle g_{n-j} | g_{n-j} \rangle}, & \begin{array}{l} n = 3,\ldots,2d+1, \\ j = 1,\ldots,\min(4,n-1). \end{array} \end{cases} \quad (63)
$$

We see $g_{2d+1}(z) = -P_d^{(i+1)}(z) + f(z)Q_d^{(i+1)}(z)$, as this is the only member in the span of $h_{2d+1}(z)$. Comparing with Eq. (48),

$$
\langle g_{2d+1} | g_{2d+1} \rangle \quad (64)
$$

is seen to be the mean square error.

The recursion (62) is used to find the set of values c_{nj} and the mean square error, then by letting $g_n^{P,Q}(z) = g_n^P(z) + f(z)g_n^Q(z)$ and setting it equal to (62) we can find $P_d^{(i+1)}(z)$ and $Q_d^{(i+1)}(z)$ by considering

$$
g_n^{P,Q}(z) = zg_{n-2}^{P,Q}(z) - \sum_{j=1}^{\min(n-1,4)} c_{nj}g_{n-j}^{P,Q}(z), \quad (65)
$$

where

$$
g_1^P(z) = 0, \quad g_2^P(z) = -1, \quad g_{2d+1}^P(z) = P_d^{(i+1)}(z),
$$
$$
g_1^Q(z) = 1, \quad g_2^Q(z) = -c_{21}, \quad g_{2d+1}^Q(z) = Q_d^{(i+1)}(z).
$$

To find the pole weights, the derivative of $Q_d^{(i+1)}(z)$ is also required. Differentiation of (65) provides this via the formula

$$
(g_n^Q)'(z) = g_{n-2}^Q(z) + z(g_{n-2}^Q)'(z) - \sum_{j=1}^{\min(n-1,4)} c_{nj}(g_{n-j}^Q)'(z),
$$
$$
(66)
$$

where

$$
(g_1^Q)'(z) = 0, \quad (g_2^Q)'(z) = 0, \quad (67)
$$

$$
(g_{2d+1}^Q)'(z) = \frac{dQ_d^{(i+1)}(z)}{dz}. \quad (68)
$$

The above formulas give us the ability to calculate $P_d^{(i+1)}(z)$, $Q_d^{(i+1)}(z)$, and $\frac{dQ_d^{(i+1)}(z)}{dz}$ at any point between a and b along the imaginary axis. This provides a way to calculate the $Q_d^{(i+1)}(z)$ for the next iteration and also to expand $\frac{P_d^{(i+1)}(z)}{Q_d^{(i+1)}(z)}$ as partial fractions,

$$
\frac{P_d^{(i+1)}(z)}{Q_d^{(i+1)}(z)} = \sum_{k=1}^{d} \frac{w_k}{z - z_k}. \quad (69)
$$

Muller's method [48] is used to provide the poles, z_k, which are the roots of $Q_d^{(i+1)}(z)$, while the residue theorem [49] yields

the weights,

$$w_k = \frac{P_d^{(i+1)}(z_k)}{\frac{dQ_d^{(i+1)}(z_k)}{dz}}. \qquad (70)$$

In the implementation of the least squares procedure the integrals in Eq. (63) are discretized with the extended trapezium rule [50], using 41 points. We also choose to iterate Eq. (48) through $i = 1, i_{max}$ in all calculations, which is found to work well for $i_{max} = 5$.

The least squares method is reliant on access to a reasonable initial guess of the denominator $Q^{(0)}(z)$. Reference [46] used a continued fraction representation to gain a denominator. However, for Eq. (43) this is not available due to the square root in its argument. An initial guess is found by realizing that an approximation with d poles has to be calculated before we know the mean squared error. However, looping through $d = 1, 2, \ldots$ until the error is reduced sufficiently, gives an automatic way to produce an approximation with a given error. This can also be used to generate an initial guess from the previous step using [38,39]

$$Q_d^{(0)}(z) = \begin{cases} z - \left(\frac{b+a}{2} - i\frac{b-a}{5}\right), & d = 1 \\ (z - 2z_{d-1})Q_{d-1}^{(0)}(z), & d \geqslant 2. \end{cases} \qquad (71)$$

In the above z_{d-1} is the zero furthest from the imaginary axis. Calculation of the above specifies the initial guess for $d = 1$, after which the results from the approximation with $d - 1$ poles is used to give the guess for an approximation with d poles. Algorithm 1 summarizes the method so far.

Algorithm 1 Nonlinear least squares method

$d = 1$.
Set $Q_1^{(1)}(z) = z - \left(\frac{b+a}{2} - i\frac{b-a}{5}\right)$.
while $\langle g_{2d+1}|g_{2d+1}\rangle < \epsilon \int_a^b |f(z)|^2\, dz$ **do**
  for $i = 1, i_{max}$ **do**
    Calculate the coefficients c_{nj} and $g_{2d+1}(z)$ using Eq. (62).
    Calculate the poles by applying Muller's method to $Q_d^{(i)}(z)$,
      which is calculated using Eq. (65).
    Use the poles to calculate $Q_d^{(i+1)}(z)$.
  end for
  Calculate $Q_{d+1}^{(1)}(z)$ from (71).
  $d = d + 1$.
  Calculate the mean square error, $\langle g_{2d+1}|g_{2d+1}\rangle$.
end while
Calculate each pole's weight using Eq. (70).
Return the d poles and their corresponding weights.

---

2. Bootstrap method of nonlinear least squares

Now the ability to find a pole approximation on an interval has been gained, we look at how this can be embedded in the bootstrap procedure that allows for an accurate approximation to the kernel in Eq. (43).

The modification described in [38,39] is to split up the interval on the imaginary axis into subintervals on which the function is smooth enough to be approximated well by the least squares method. There are three considerations that must be made for this to be successful.

First, we require some way of joining the approximations made on each subinterval. The values of an approximation made on a particular subinterval are nonzero outside of it and so simply adding the results means each approximation will interfere with one another. This can be solved by specifying some order to make the approximations. Then, by approximating the kernel on the first subinterval, we continue by making approximations of the kernel with the previous results subtracted on the subsequent subintervals. In this way the current approximation takes account of the previous ones and adding the resultants will approximate the kernel.

Secondly, imagining the function on the complex plane, then it may be the case that, in an interval, the kernel can be well approximated by poles which make a larger contribution elsewhere on the imaginary axis. To see this, consider an approximation where some poles are far away from the interval along the imaginary line on which the approximation was calculated. Generally these contributions to the current subinterval are small, but can be large for other intervals which the poles are close to. So, approximations on the following subintervals will also have to describe these poles, not just the kernel. We therefore want to ensure poles located far from the subinterval they were calculated on are excluded.

It may also be the case that the size of a subinterval is much smaller than the absolute value of the real part of a pole. This time we would not expect the least squares algorithm to calculate this pole accurately because its contribution is over a larger interval than what we are analyzing. So poles holding this property are excluded too.

It seems sensible that only poles that are found near to the part of the imaginary axis they were calculated on should be included in the approximation. Therefore, a near pole is defined and we say that only the near poles should be included in the sum of poles approximation. Specifically, a pole z_k' is defined a near pole on the interval $[-1,1]$ if

$$12 \leqslant \int_{-1}^{1} \frac{1}{|x - z_k'|^2}\, dx = \frac{1}{Im(z_k')}\left[\arctan\left(\frac{Re(z_k') + 1}{Im(z_k')}\right) - \arctan\left(\frac{Re(z_k') - 1}{Im(z_k')}\right)\right]. \quad (72)$$

The notation z_k' used is to specify a pole z_k that has been scaled onto $[-1,1]$ using

$$z_k' = \frac{z_k - \frac{1}{2}(a+b)}{\frac{1}{2}(b-a)}. \qquad (73)$$

The criterion (72) describes an elliptical area surrounding the $[-1,1]$ interval [38,39]. The value on the left-hand side describes how tightly the ellipse is to enclose the interval, where a larger value would describe a smaller area. A value of 12 yields an ellipse that tightly surrounds the interval that has been found to be practical for our purposes.

Finally it may also be the case that a pole with a positive real part may also suffice to give a good approximation of the function on a subinterval. Here we make an assumption that no poles with positive real part should be included, as this would
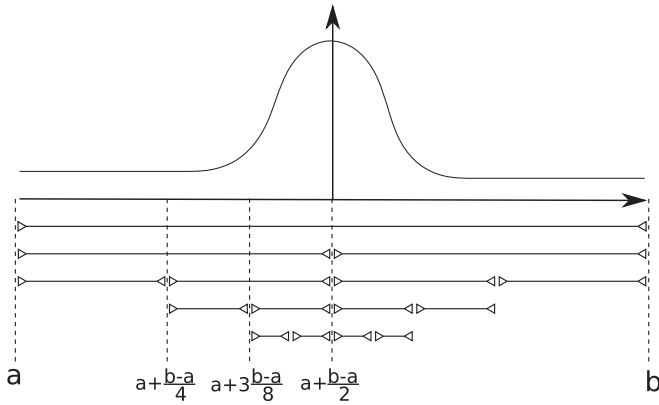
FIG. 1. A diagram of how the binary splitting is expected to behave, with the smallest subintervals being created near the complicated behavior of the function being approximated.

violate the requirement that all poles should be to the left of the contour in the Bromwich integral (31).

Splitting the imaginary axis. Now a method to split up the imaginary axis into subintervals, on which the kernel can be approximated well by the leastsquares method, is required. This can be done recursively, by considering an interval and splitting it into two equally sized subintervals. On each of the subintervals a Chebyshev polynomial approximation is made and a criterion of whether or not this approximation is accurate has to be specified. If the criterion is satisfied, then no further splitting is done, but if it is not, we split the subinterval into two further subintervals and repeat the procedure. Figure 1 gives a simple illustration on how we would like an interval to be split up.

3. The splitting criteria

Deciding whether or not a kernel on an interval is smooth can be done on the assumption that if the function can be well approximated by a low-order polynomial in that interval then the approximation will be successful. Therefore a Chebyshev polynomial approximation is calculated for the kernel on the interval and if the coefficients have gotten sufficiently small, implying convergence, we set the splitting criterion to false and the interval will not be split any further. A K-term Chebyshev approximation $f_{\text{approx}}(z)$ of a function $f(z)$ on $z \in [a,b]$ can be calculated using the formula [50]

$$f_{\text{approx}}(z) = \sum_{k=0}^{K-1} \alpha_k T_k(z). \tag{74}$$

The coefficients a_j are given by

$$\alpha_0 = \frac{1}{K} \sum_{j=1}^{K} f(y_k), \tag{75}$$

$$\alpha_j = \frac{2}{K} \sum_{k=0}^{K-1} f(y_k) T_j(x_k). \tag{76}$$

The values of x_k, which are the roots of the Chebyshev polynomial, are given by

$$x_k = \cos\left(\frac{\pi\left(k + \frac{1}{2}\right)}{n}\right), \tag{77}$$

and y_k scales x_k from $[-1,1]$ to $[a,b]$:

$$y_k = \frac{(b + a) + (b - a)x_k}{2}. \tag{78}$$

If $f_{\text{approx}}(z)$ approximates $f(z)$ well, the coefficients of final terms in the series (74) should be relatively small. So, in practice we only require the coefficients and calculate

$$S = \frac{|\alpha_{K-1}| + |\alpha_{K-2}|}{\sum_{k=0}^{K-2} |\alpha_k|}, \tag{79}$$

which can be thought of as a measure of convergence. A δ is defined so that if $S \leqslant \delta$, the splitting criterion is set to false and if $S > \delta$ the splitting criterion is set to true. For all the results in this work the values $K = 10$ and $\delta = 10^{-3}$ are used.

a. Binary tree description of an interval. Information on the subintervals is stored in a binary tree [51]. A binary tree is a collection of nodes which contain at least an association to a parent node and associations to left and right child nodes. These associations are called branches and give the tree its structure. It can be that a node's associations to both children are not specified, in which case we call it a leaf. There must be one and only one node without a parent, which we call the root. This defines a structure which has a single starting point, the root, and branches out to multiple end points, the leaves, like a tree.

To make the binary tree useful for storing the splitting of our interval, we must append some additional information to each node. We choose to append the boundaries of each interval and what is called the node depth. The node depth is equal to the node depth of its parent plus one. The root's node depth is defined to be zero. For the interval split shown in Fig. 1 we would have a binary tree as shown in Fig. 2.
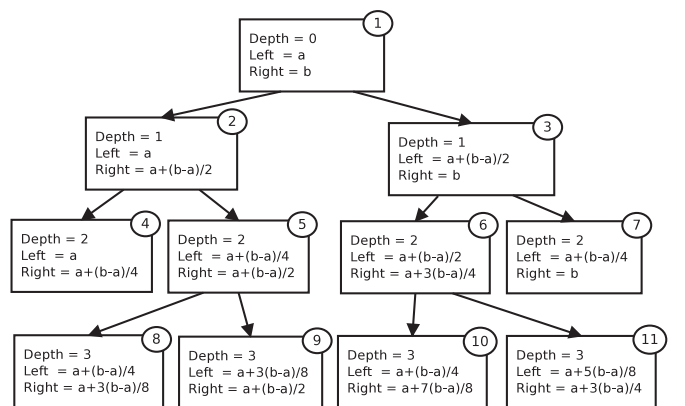

FIG. 2. A diagram of the binary tree that describes the splitting, of the interval, in Fig. 1. Each box is a node with two arrows pointing away from it toward its children and an arrow from another node pointing towards it from its parent. Left and right specifies the two end points of the intervals.

b. Creating the binary tree. Now that we have shown how we may use a binary tree to describe the splitting of an interval, we go on to describe how the binary tree is created. The procedure relies heavily on recursion and is described in Algorithm 2.

Algorithm 2 Insert (node)

---

if node.depth > maxdepth **then** maxdepth = node.depth
end if
if (Splitting Criterion True) **then**
  node.left.a = node.a
  node.left.b = $\frac{1}{2}$(node.a + node.b)
  node.left.depth = node.depth + 1
  call insert(node.left)

  node.right.a = $\frac{1}{2}$(node.a + node.b)
  node.right.b = node.b
  node.right.depth = node.depth + 1
  call insert(node.right)
end if

Following the algorithm through we see that when a node is split in two, we move to its left child and check whether it needs to be split. If it does, two children are created and we move to the left again; if not then we move the parents right child and repeat the procedure. This process begins at the root of the tree.

c. The bootstrap method. How the previous results are used to create an approximation to a kernel function is now specified. First create the binary tree, then begin at the leftmost subinterval at the maximum depth, and approximate the kernel and keep only the near poles. Then move rightward through the rest of the nodes at that depth and approximate the kernel with all the previous near poles subtracted. Then move up to the next deepest and repeat the process, until the root is reached. Approximate the root, and keep all found poles. The order we would take for the tree shown in Fig. 2 would be 8, 9, 10, 11, 4, 5, 6, 7, 2, 3, 1. To return the poles at a certain depth a modified in-order tree transversal [51], as shown in Algorithm 3, is used to produce a linked list. The entire bootstrap procedure is summarized in Algorithm 4.

Algorithm 3 NodesAtDepth(node)

---

if node.left exists **then**
  NodesAtDepth(node.left)
end if
if node.depth = d **then**
  Add node to end of linked list
end if
if node.right exists **then**
  NodesAtDepth(node.right)
end if

Algorithm 4 Bootstrap nonlinear least squares

Specify an interval $[a,b]$.
Use Algorithm 2 to create a binary tree.
for d = maxdepth : -1 : 1 **do**
  Use Algorithm 4 to return N subintervals $[a_n, b_n]$ at depth d.
  for $n = 1, N$ **do**
    Use Algorithm 1 to return the poles and corresponding
      weights on $[a_n, b_n]$.
    Discard poles and weights which do not meet criterion (72).
    Add remaining poles and weights to list.
  end for
end for
Use Algorithm 1 to return poles and corresponding weights on
  root interval $[a,b]$ and add to list.
Return list of weights and poles.

D. Boundary kernel calculation

How we calculate the proton kernel

$$\hat{G}_{\kappa,\mu}(R,s) = \frac{1}{b\sqrt{s}} \left(\frac{W_{\kappa,\mu}(br\sqrt{s})}{\frac{\partial W_{\kappa,\mu}(br\sqrt{s})}{\partial r}} \right) \Bigg|_{r=R} \qquad (80)$$

is now described. The kernels studied in [38,39,46] had continued fraction representations which provided an efficient and accurate means to calculate values over the entire complex plane. We choose a similar strategy and use the continued fraction [45]

$$\frac{W_{\kappa,\mu}(z)}{\sqrt{z}\, W_{\kappa-\frac{1}{2},\mu-\frac{1}{2}}(z)} = 1 + \cfrac{\frac{v_1}{z}}{1 + \cfrac{\frac{v_2}{z}}{1+\cdots}}, \qquad (81)$$

where

$$v_{2n+1} = \tfrac{1}{2} + \mu - \kappa + n, \qquad (82)$$

$$v_{2n} = \tfrac{1}{2} - \mu - \kappa + n. \qquad (83)$$

The above converges for $|\arg(z)| < \frac{1}{2}$ and $\mu + \frac{1}{2} \pm (\kappa+1) \neq -1, -2, \ldots$. From Sec. IV B we know $-\frac{1}{2}\pi < \arg z < 0$ and so the above equation is valid for our considerations. The use of the recurrence relation [44,45,52]

$$W_{\kappa-(1/2),\mu-(1/2)}(z) = \frac{1 - 2\mu - z}{(1 - 2\mu - 2\kappa)\sqrt{z}} W_{\kappa,\mu}(z)$$
$$+ \frac{\sqrt{z}}{\kappa - \frac{1}{2} + \mu} \frac{d W_{\kappa,\mu}(z)}{dz} \qquad (84)$$

allows us to express (81) in terms of a Whittaker function and its derivative as given in the kernel. To do this write the reciprocal of Eq. (81) and substitute Eq. (84) into it to produce

$$\frac{1 - 2\mu - z}{(1 - 2\mu - 2\kappa)} + \frac{z}{\kappa - \frac{1}{2} + \mu} \left(\frac{\frac{dW_{\kappa,\mu}(z)}{dz}}{W_{\kappa,\mu}(z)} \right) = \frac{1}{1 + \cfrac{\frac{v_1}{z}}{1+\cfrac{\frac{v_2}{z}}{1+\cdots}}}. \qquad (85)$$
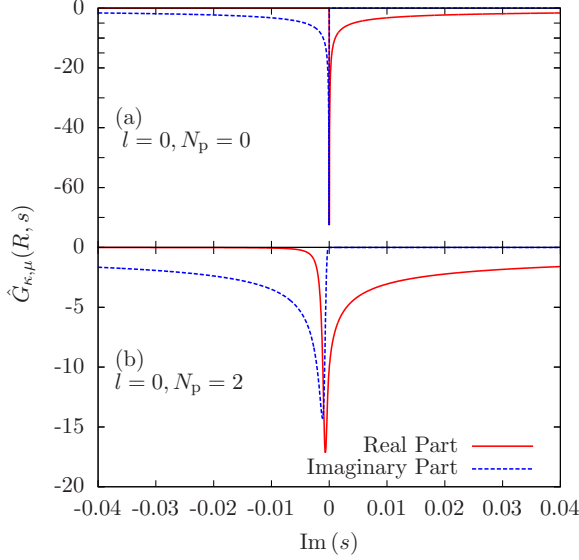
FIG. 3. (Color online) A plot showing the values of the kernel (80) using the values shown and $R = 9.9$.

We are then able to rearrange to find $\frac{\frac{dW_{\kappa,\mu}(z)}{dz}}{W_{\kappa,\mu}(z)}$ and therefore its reciprocal $\frac{W_{\kappa,\mu}(z)}{\frac{dW_{\kappa,\mu}(z)}{dr}}$. Finally, upon multiplying by $\frac{1}{b\sqrt{s}}$ the following continued fraction can be written for the kernel:

$$\hat{G}_{\kappa,\mu}(R,s) = \frac{1}{b\sqrt{s}} \frac{2z}{1 - 2\mu - z + \cfrac{\frac{2(\kappa+\mu)-1}{v_1}}{1+\cfrac{\frac{z}{v_2}}{1+\frac{z}{1+\dots}}}}, \quad (86)$$

using z as defined below Eq. (32). We calculate the above using Lentz's algorithm [50,53,54]. It should be noted that this continued fraction does not terminate after a finite number of terms, therefore we are not able to use it to find a finite sum of poles for the kernel. Therefore, the least squares method is used to find an accurate sum-of-poles approximation with a finite number of terms.

In Fig. 3 the kernel is plotted for two different parameter sets: $l = 0$, $N_{\mathrm{p}} = 0$ and $l = 0$, $N_{\mathrm{p}} = 2$.

Figure 3 shows that the complicated behavior of the functions is centered nearby the origin and that the $N_{\mathrm{p}} = 0$ kernel appears less smooth than that of $N_{\mathrm{p}} = 2$. This occurs when $N_{\mathrm{p}} = 0$ with small l and will be shown to have consequences when the approximations are made for these kernels. Extending the plot's x axis outward would show the function slowly decaying to zero.

E. Boundary discretization

How we discretize the ABC with a proton kernel is now described. As $G_{\kappa,\mu}(R,\tau)$ and $\frac{\partial Q(r,t-\tau)}{\partial r}|_{r=R}$ are both continuous we expect their numerical integration, by the trapezium rule, to be accurate. However, we have observed this is not the case when the trapezium rule is applied directly. This can be resolved by considering the case of $\sigma = 0$ where it is known the analytic form of the kernel contains a square root singularity [55]. The sum of exponentials should accurately describe this behavior and hence not be integrated accurately

by the trapezium rule. So, in analogy with [11] integration by parts is used on Eq. (42), before it is discretized, to produce

$$Q(R,t) + \frac{\partial Q(R,t)}{\partial r} H_{\kappa,\mu}(R,0)$$
$$= -\int_0^t H_{\kappa,\mu}(R,\tau) \frac{\partial}{\partial \tau} \left(\frac{\partial Q(r,t-\tau)}{\partial r} \bigg|_{r=R} \right) d\tau.$$

During the by-part manipulation $G_{\kappa,\mu}(R,\tau)$ is integrated to give

$$H_{\kappa,\mu}(R,\tau) = \sum_{k=1}^{d} \frac{w_k}{s_k} e^{s_k \tau}, \quad (87)$$

which is also a sum of exponentials and can be easily evaluated.

Time and space discretization

A semidiscrete equation can be gained by evaluating τ at values on the temporal grid defined in Sec. III B for which $\tau = t_n$ and $t = t_N$. Use of the extended midpoint rule [50],

$$\int_0^t f(\tau) d\tau = \Delta t \sum_{n=0}^{N-1} f\left(t_{n+\frac{1}{2}}\right) + O(\Delta t^2), \quad (88)$$

to evaluate the integral and the difference formulas

$$f(r,t_{n-(1/2)}) = \frac{f(r,t_n) + f(r,t_{n-1})}{2} + O(\Delta t^2), \quad (89)$$

$$\frac{\partial f(r,t_{n-(1/2)})}{\partial t} = \frac{f(r,t_n) - f(r,t_{n-1})}{\Delta t} + O(\Delta t^2), \quad (90)$$

for functions evaluated at a half time step, allows us to write the semidiscrete equation

$$Q(R,t_N) + \frac{\partial Q(R,t_N)}{\partial r} H(R,0)$$
$$= -\sum_{n=0}^{N-1} H(R,t_{n+(1/2)}) \left[\frac{\partial Q(R,t_{N-n-1})}{\partial r} - \frac{\partial Q(R,t_{N-n})}{\partial r} \right]$$
$$+ O(\Delta t^2). \quad (91)$$

For the spatial discretization the absorbing boundary is applied at $R = r_{M-(1/2)}$ between the penultimate and final spatial grid points. The following difference formulas are used:

$$f(r_{M-(1/2)},t) = \frac{f(r_M,t) + f(r_{M-1},t)}{2} + O(\Delta r^2), \quad (92)$$

$$\frac{\partial f(r_{M-(1/2)},t)}{\partial r} = \frac{f(r_M,t) - f(r_{M-1},t)}{\Delta r} + O(\Delta r^2), \quad (93)$$

at the points between the spatial grid, yielding the following fully discretized ABC:

$$(1 + B) Q(r_M,t_N) + (1 - B) Q(r_{M-1},t_N)$$
$$= -A H(r_{M-(1/2)},t_{(1/2)}) [Q(r_M,t_{N-1}) - Q(r_{M-1},t_{N-1})]$$
$$- A \sum_{n=1}^{N-1} H(r_{M-(1/2)},t_{n+(1/2)})[Q(r_M,t_{N-n-1})$$
$$- Q(r_{M-1},t_{N-n-1}) - Q(r_M,t_{N-n})$$
$$+ Q(r_{M-1},t_{N-n})] + O(\Delta r^2, \Delta t^2), \quad (94)$$

where

$$A = \frac{2}{\Delta r}, \tag{95}$$

$$B = A[H(r_{M-(1/2)}, 0) - H(r_{M-(1/2)}, t_{1/2})]. \tag{96}$$

Once the poles and weights have been calculated by using Algorithm 4, they can be used with Eq. (87) to calculate the integral of the kernel for any required time. In general, we are required to recalculate Algorithm 4 for different values of l, N_p, and R. Replacement of the last equation of the matrix described in Sec. III B will then impose the boundary condition (9) on the calculation.

The fully discrete equation shows the consequence of the temporal nonlocality of the ABC, noted at the end of Sec. IV B, as it contains a sum with upper bound $N - 1$. This requires evaluating an increasing number of terms as the calculation progresses, which of course has implications on the computational cost. However, it has been noted that this can be remedied by a recursive evaluation of the absorbing boundary condition [56].

V. TESTING OF THE ABCS

In this section the implementations of the bootstrap nonlinear least squares and the absorbing boundary conditions are tested separately from the TDHF calculations. We start with various tests of the bootstrap implementation and then move on to show the results of applying the ABCs to some simple calculations of Schrödinger equations.

A. Testing of the bootstrap implementation

The results of applying the least squares approximation to the kernel (43) are now shown. We have found that Algorithm 4 produces the smallest mean square error when used to make an approximation on an asymmetric interval. Therefore the values $a = -10^9 i$ and $b = 10^8 i$ are chosen, so that the interval considered encloses the one used in [38,39] while being asymmetric. First an example binary tree, produced by Algorithm 2, is shown in Fig. 4 for $l = 0$, $N_p = 2$, and $R = 9.9$.

By comparing Fig. 4 to Fig. 3 we can see that the width of the intervals becomes smallest around the complicated behavior of the function, as wanted. It is noted that the tree depth is limited to 42 in the implementation, as for $l = 0$ and $N_p = 0$ we want to prevent oversplitting, which may compromise the accuracy of the method. Figure 3 shows why this occurs, as the $N_p = 0$ kernel is not as smooth as the $N_p \neq 0$ kernel.

A selection of results is shown in Table II for values of l, N_p and R required by the Hartree-Fock calculations.

We see that for most cases the kernels are accurately approximated by the bootstrap method. Only for $N_p = 0$ does the mean square error become significantly larger and as l increases the accuracy is recovered. This appears to be due to the more complicated behavior of the kernel around the origin, as shown in Fig. 3, as the mean square errors with a small interval around the origin excluded are all similar. The larger error at the origin is confirmed by Fig. 5 where examples are given to show how the relative error, between the
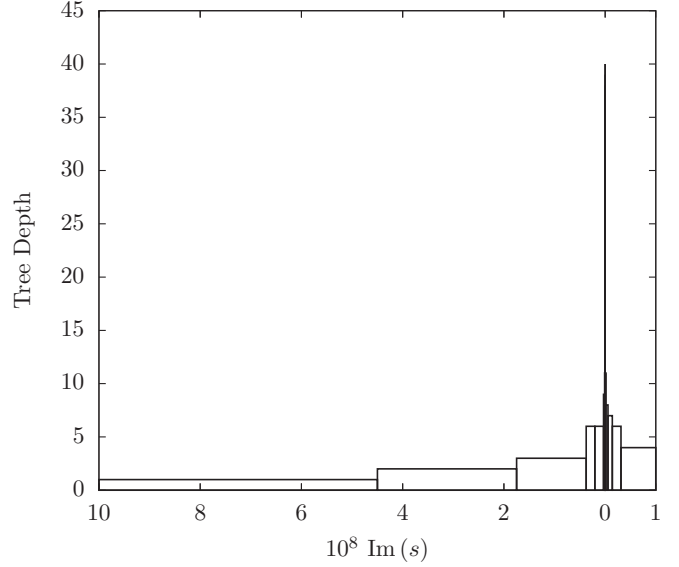

FIG. 4. A box plot that shows the depths and subinterval locations of the binary tree produced for a calculation using $l = 0$ and $N_p = 2$.

TABLE II. Table showing the number of poles used to produce an approximation with the relative error specified. We show results for the values of l and N_p that are required by the Hartree-Fock calculations, for selected artificial boundaries. The first three approximations for each R are used within calculations of the neutron single particle states and the remaining for the proton single particle states.

| R	N_P	l	No. of poles	Error on $[a,b]$	Error on $[a,b]/(-10^{-4}, 10^{-4})$
	0	0	118	9.30×10^{-2}	2.56×10^{-16}
	0	1	109	2.25×10^{-13}	1.36×10^{-16}
	0	2	112	4.08×10^{-14}	6.53×10^{-16}
	2	0	114	8.44×10^{-17}	7.92×10^{-17}
9.9	8	0	103	8.77×10^{-17}	8.66×10^{-17}
	8	1	104	1.58×10^{-16}	1.59×10^{-16}
	20	0	97	1.06×10^{-16}	1.05×10^{-16}
	20	1	91	1.77×10^{-16}	1.77×10^{-16}
	20	2	97	3.32×10^{-16}	3.33×10^{-16}
	0	0	117	9.30×10^{-2}	2.48×10^{-16}
	0	1	117	7.73×10^{-13}	9.20×10^{-17}
	0	2	112	1.46×10^{-16}	6.31×10^{-17}
	2	0	108	1.15×10^{-16}	1.25×10^{-16}
19.9	8	0	104	7.77×10^{-17}	7.80×10^{-17}
	8	1	108	7.03×10^{-17}	6.87×10^{-17}
	20	0	101	6.09×10^{-17}	6.06×10^{-17}
	20	1	100	7.37×10^{-17}	7.33×10^{-17}
	20	2	92	6.73×10^{-17}	6.72×10^{-17}
	0	0	117	9.30×10^{-2}	2.50×10^{-16}
	0	1	104	1.05×10^{-11}	1.50×10^{-16}
	0	2	109	7.11×10^{-15}	1.95×10^{-16}
	2	0	109	1.83×10^{-16}	1.96×10^{-16}
29.9	8	0	108	1.13×10^{-16}	6.91×10^{-17}
	8	1	106	1.35×10^{-16}	1.29×10^{-16}
	20	0	102	2.54×10^{-16}	2.56×10^{-16}
	20	1	98	1.21×10^{-16}	1.21×10^{-16}
	20	2	101	7.02×10^{-17}	6.94×10^{-17}
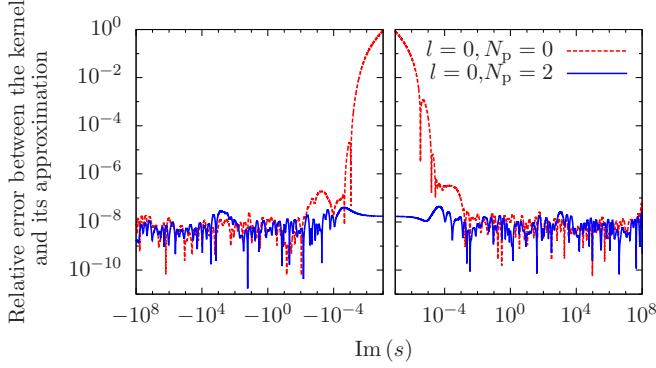
FIG. 5. (Color online) A graph showing how the relative errors between two kernels and their approximations are distributed over the imaginary axis.

approximation and the kernel, is distributed over the imaginary axis.

Both approximations have a similar magnitude of error away from the origin. However, in the approximation of N_p we see that the error between the approximation and the kernel spikes. It will be shown later that the results presented are accurate enough for our needs.

Figure 6 shows the pole locations in the complex plane of the poles found by the bootstrap least squares procedure. We denote the poles of the kernel (80) by s_k.

We see that the real and imaginary parts of the poles are similar in magnitude, which is a result of the near pole criterion.

B. Testing of the absorbing boundary conditions

In this section the discretized ABC, Eq. (94), is tested for a simplified case of a Schrödinger equation with the same form within the interior as required by the exterior. Specifically the
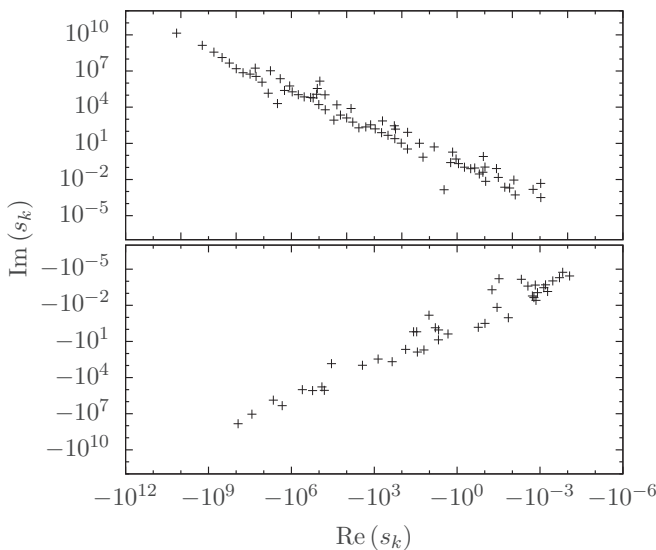

FIG. 6. A graph of the complex plane, showing the pole locations found by the bootstrap method for the values $l = 0$, $N_p = 2$, and $R = 9.9$.

following will be solved:

$$i \frac{\partial Q_{l,N_p}(r,t)}{\partial t} = \frac{\partial^2 Q_{l,N_p}(r,t)}{\partial r^2}$$
$$+ \left(\frac{\eta N_p}{r} + \frac{l(l+1)}{r^2} \right) Q_{l,N_p}(r,t), \quad (97)$$

subject to the initial and boundary conditions

$$Q_{l,N_p}(r,0) = Are^{-(r-5)^2}, \quad (98)$$

$$Q_{l,N_p}(0,t) = 0, \quad \lim_{r \to \infty} Q_{l,N_p}(r,t) = 0. \quad (99)$$

In the above A is chosen to normalize $Q_l(r,0)$. The values of l and N_p considered will be just those required by the TDHF calculations shown later.

C. Radial comparison of wave function

Testing is begun by considering how the error from the absorbing boundaries affects the interior solution, by plotting the maximum absolute error that has occurred during the calculation. At each r

$$\max_{t \in [0,50]} \left| Q_{l,N_p}^{(Ref)}(r,t) - Q_{l,N_p}^{(ABC)}(r,t) \right| \quad (100)$$

is plotted, where $Q_l^{(Ref)}(r,t)$ and $Q_l^{(ABC)}(r,t)$ are the calculations with reflecting and absorbing boundaries, respectively. Figure 7 shows the results for the various l and N_p values and three different grid spacings. The spacings $\Delta r = 0.2$ and $\Delta t = 0.2$ are chosen because it is the spacing we use in the Hartree-Fock calculations; the two other spacings are used to show the dependence of the error on the discretization. The reference
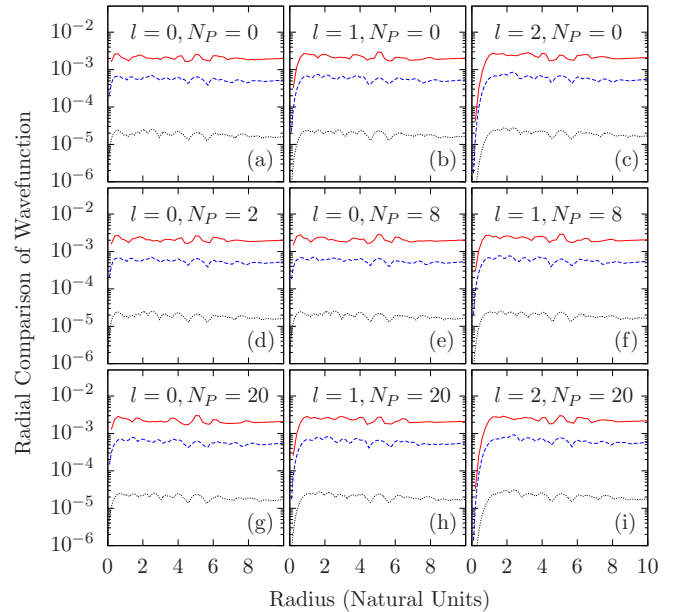

FIG. 7. (Color online) The figures shows the maximum error of the radial component of the wave functions from times 0 to 15, for angular momenta and proton number shown, calculated with each technique. The value in Eq. (100) is plotted against the radius. The solid red lines show the result from using grid spacings $\Delta r = 0.2$ and $\Delta t = 0.2$, the dashed blue lines using $\Delta r = 0.1$ and $\Delta t = 0.1$, and the dotted black lines using $\Delta r = 0.01$ and $\Delta t = 0.01$.

solution is calculated on a grid with an outer boundary at 200 fm, which is far enough away to stop reflection from occurring.

We see that in all cases the error has remained small throughout the interior, for the $\Delta r = 0.2$, $\Delta t = 0.2$ case bounded by 10^{-2}, for $\Delta r = 0.1$, $\Delta t = 0.1$ bounded by 10^{-3}, and for $\Delta r = 0.01$, $\Delta t = 0.01$ bounded by 10^{-5}. The errors can be seen to be bounded similarly to those presented previously [11]. There also appears to be no ill effects from the drop in accuracy, near the origin, of the $N_p = 0$ approximations, with a similar magnitude of error being seen for all cases. This is presumably due to the region of low accuracy being a rather small part of the whole region, and with sufficient unimportance to cause a serious problem.

D. Temporal comparison of probability

We now test how the error evolves through time. This is done by calculating the probability of finding a particle inside the interior region over time. Mathematically

$$P(t) = \int_0^{10} \left| Q_{l,N_p}(r,t) \right|^2 dr \qquad (101)$$

is calculated with reflecting and absorbing boundaries and the absolute value of the difference taken. Again the time interval of the calculation is $[0, 50]$ and we choose the reflecting boundary to be at $r = 200$. Figure 8 shows the results.

We see that in time, also, the error remains bounded. From the plots it appears the bound on the error is proportional to the grid spacings. With the errors coming from the ABCs being small for all test cases, we proceed to use them with confidence.
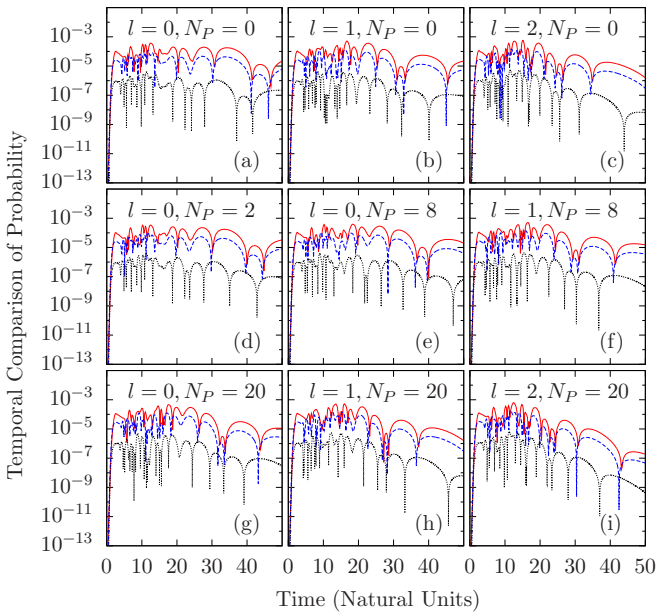
FIG. 8. (Color online) These plots show how the error in the probability from the absorbing boundaries changes through time. Equation (101) is calculated with reflecting and absorbing boundaries and the absolute value of their difference taken. The solid red lines show the result from using grid spacings $\Delta r = 0.2$ and $\Delta t = 0.2$, the dashed blue lines using $\Delta r = 0.1$ and $\Delta t = 0.1$, and the dotted black lines using $\Delta r = 0.01$ and $\Delta t = 0.01$.
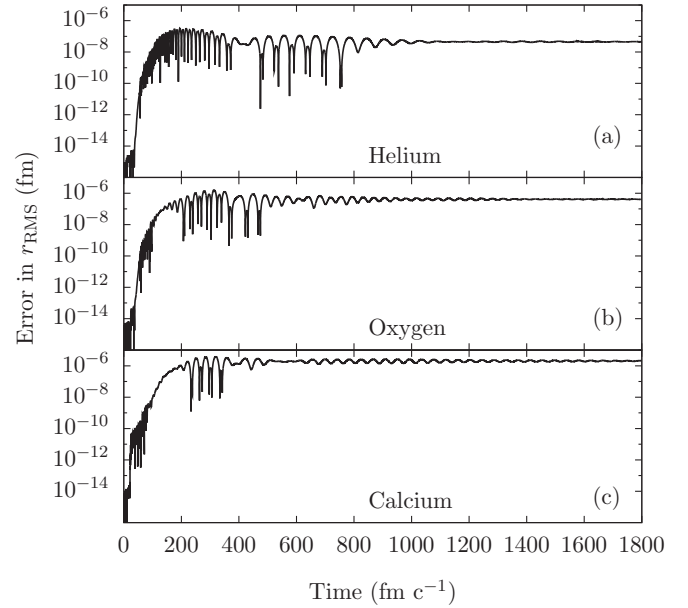
FIG. 9. Plots showing the difference in the root mean square radii of reference calculations and a calculation on a grid with an outer boundary at 30 fm with ABCs applied.

VI. RESULTS FROM THE TDHF

In this section comparison will be made to reference solutions, calculated with reflecting boundaries on a grid with outer boundary at 700 fm to ensure reflection does not occur. Figure 9 shows the absolute error in the rootmean square radius

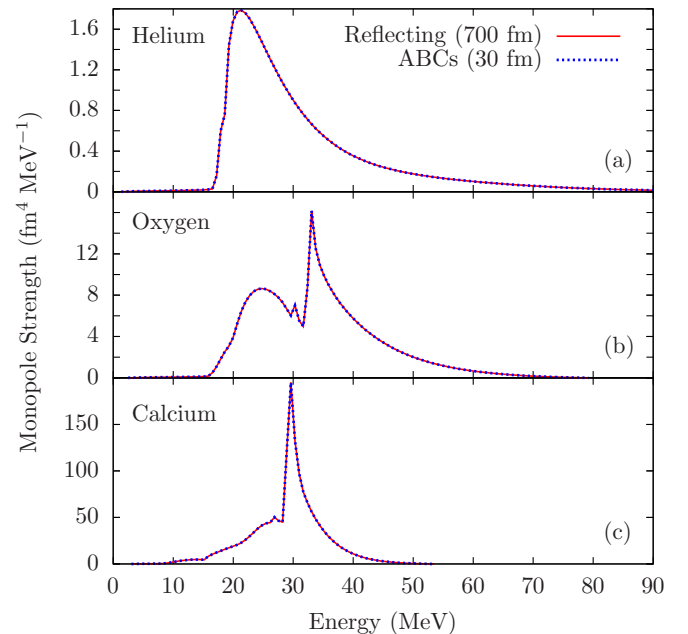$$\left(\int_0^8 4\pi r^4 \rho(r,t) \, dr \right)^{1/2}, \qquad (102)$$

FIG. 10. (Color online) Plots showing the strength functions for various nuclei. The solid red line shows the reference strength, whereas the blue dashes show the strength from a calculation on a grid with an outer boundary at 30 fm with ABCs applied.

TABLE III. Table showing the time taken to calculate the various stages of the time-dependent code using ABCs. The values in the column labeled by BNLS are the times taken to calculate the bootstrap method of nonlinear least squares for all kernels required, by the time-dependent Hartree-Fock calculation. The column labeled TDHF + ABCs shows the times taken to calculate the solution to the TDHF equations with ABCs applied at 30 fm. The column labeled "Total" contains the sum of the times for the BNLS and TDHF + ABCs calculations. Finally, the values in the column labeled "Ref. sol." are the time taken to calculate reference solution of the TDHF equations.

Nucleus	BNLS (s)	TDHF + ABCs (s)	Total (s)	Ref. sol. (s)
Helium	1.99	7.42	9.41	135.16
Oxygen	2.79	13.64	16.43	267.38
Calcium	3.78	24.12	27.90	475.12

between the reference solution and a calculation made with ABCs at 30 fm. Placement of the cutoff for the integration in the above is a parameter within TDHF calculations [21], but 8 fm appears to work well.

We see in each case the errors are consistently small and appear to be bounded by 10^{-5} fm. We therefore conclude that the nonlinear portion of the potential is not large enough to disturb this type of calculation. However, these results will be of no use if the strength function is particularly sensitive to these errors and cannot be resolved properly. Therefore, a comparison of the reference strength functions with those calculated using ABCs is shown in Fig. 10.

The plots show this is not the case and the strength function calculated with ABCs is indistinguishable by eye to the reference. Therefore, strength functions from calculations using ABCs should be accurate enough to be successfully compared to experiment.

Finally the times taken to calculate the results and references are shown in Fig. 10 are given in Table III, as we wish to see if there is any improvement in efficiency.

The table shows us that the time for completion of each calculation has been drastically reduced. In this simplified case the absorbing boundary conditions approach has shown to be efficient and accurate.

VII. CONCLUSION AND OUTLOOK

In this work we have presented an application of ABCs to Hartree-Fock calculations of spherical nuclei. ABCs with a centrifugal barrier and Coulomb potential in an exterior domain were considered. It was shown the ABCs required an inverse Laplace transform that was too complex for an analytical expression to be found. So, a bootstrap nonlinear least squares method was implemented to produce an accurate sum-of-poles approximation to the kernel within the inverse Laplace transforms. The approximation was shown to be accurate and had an inverse known as a textbook result. Results of the ABC's application to TDHF calculations were similar to the Coulombless case considered previously [11], being accurate and efficient.

On the physical side the outlook for this work is to include the full Skyrme interaction, allowing realistic calculations to be carried out [32,33]. It is expected that the ABCs would perform just as effectively for the full interaction, since the splitting between the interior and exterior regions is not affected. Secondly one would like to remove the restriction of spherical symmetry and extend the method to full three-dimensional calculations [57–59]. This would allow other resonant modes to be studied and allow the calculation of nonspherical nuclei. Two possibilities under consideration to achieve this are via an expansion of the density in spherical harmonics, or an appropriate operator splitting method.

On the mathematical side we would like to offset some of the extra computational cost coming from physical improvements by increasing the efficiency of the implementation. A first improvement would be to change to a recursive evaluation of the ABCs, which is possible when using a sum-of-exponentials kernel [56]. This would result in boundary conditions requiring just $O(1)$ evaluations at each time step. As well as this, an implementation of a temporal discretization scheme that is more suited to nonlinear equations is desired. The literature already provides some methods applicable to the nonlinear Schrödinger equation [60], which offer the possibility to be generalized to the TDHF equations.

We conclude by remarking that the results presented highlight the ABC approach presented here as a valid method to handle the artificial boundary within TDHF calculations in the spatial basis. The current work also offers various avenues for improvement.

[1] P. Ring and P. Schuck, *The Nuclear Many-Body Problem*, Texts and Monographs in Physics (Springer, Berlin, 2005).

[2] R. Nesbet, *Variational Principles and Methods in Theoretical Physics and Chemistry* (Cambridge University Press, Cambridge, 2003).

[3] M. Harakeh and A. Van Der Woude, *Giant Resonances: Fundamental High-Frequency Modes of Nuclear Excitation*, Oxford Studies in Nuclear Physics Series (Oxford University Press, Oxford, 2001).

[4] P.-G. Reinhard, L. Guo, and J. Maruhn, Eur. Phys. J. A **32**, 19 (2007).

[5] M. L. Gorelik, I. V. Safonov, and M. H. Urin, Phys. Rev. C **69**, 054322 (2004).

[6] P.-G. Reinhard, P. D. Stevenson, D. Almehed, J. A. Maruhn, and M. R. Strayer, Phys. Rev. E **73**, 036709 (2006).

[7] T. Nakatsukasa and K. Yabana, Eur. Phys. J. A **25**, 527 (2005).

[8] T. Nakatsukasa and K. Yabana, Phys. Rev. C **71**, 024301 (2005).

[9] X. Antoine, A. Arnold, C. Besse, M. Ehrhardt, and A. Schädle, Commun. Comput. Phys. **4**, 729 (2008).

[10] J.-S. Wu, M. R. Strayer, and M. Baranger, Phys. Rev. C **60**, 044302 (1999).

[11] C. I. Pardi and P. D. Stevenson, Phys. Rev. C **87**, 014330 (2013).

[12] K. Xu and S. Jiang, J. Sci. Comput. **55**, 16 (2013).

[13] M. N. Harakeh, K. van der Borg, T. Ishimatsu, H. P. Morsch, A. van der Woude, and F. E. Bertrand, Phys. Rev. Lett. **38**, 676 (1977).

[14] D. Patel, U. Garg, M. Fujiwara, H. Akimune, G. Berg, M. Harakeh, M. Itoh, T. Kawabata, K. Kawase, B. Nayak, T. Ohta, H. Ouchi, J. Piekarewicz, M. Uchida, H. Yoshida, and M. Yosoi, Phys. Lett. B **718**, 447 (2012).

[15] D. H. Youngblood, P. Bogucki, J. D. Bronson, U. Garg, Y. W. Lui, and C. M. Rozsa, Phys. Rev. C **23**, 1997 (1981).

[16] F. E. Bertrand, G. R. Satchler, D. J. Horen, J. R. Wu, A. D. Bacher, G. T. Emery, W. P. Jones, D. W. Miller, and A. van der Woude, Phys. Rev. C **22**, 1832 (1980).

[17] B. L. Berman and S. C. Fultz, Rev. Mod. Phys. **47**, 713 (1975).

[18] D. H. Youngblood, H. L. Clark, and Y.-W. Lui, Phys. Rev. Lett. **82**, 691 (1999).

[19] S. Koonin, Phys. Lett. B **61**, 227 (1976).

[20] S. Stringari and D. Vautherin, Phys. Lett. B **88**, 1 (1979).

[21] P. D. Stevenson and S. Fracasso, J. Phys. G: Nucl. Part. Phys. **37**, 064030 (2010).

[22] D. Almehed and P. D. Stevenson, *Isoscalar and Isovector Giant Monopole Resonances Using TDHF*, AIP Conf. Proc. No. 802 (AIP, Melville, NY, 2005), p. 305.

[23] D. Almehed and P. D. Stevenson, J. Phys. G: Nucl. Part. Phys. **31**, S1819 (2005).

[24] P. A. M. Dirac, Proc. Cambridge Philos. Soc. **26**, 376 (1930).

[25] P. Bonche, S. Koonin, and J. Negele, Phys. Rev. C **13**, 1226 (1976).

[26] R. Cusson, R. Smith, and J. A. Maruhn, Phys. Rev. Lett. **36**, 1166 (1976).

[27] K. T. R. Davies, K. R. S. Devi, S. E. Koonin, and M. R. Strayer, in *Treatise on Heavy-Ion Science*, edited by D. A. Bromley (Plenum Press, New York, 1985), Vol. 3, Chap. 1, pp. 1–80.

[28] C. Simenel, Eur. Phys. J. A **48**, 152 (2012).

[29] E. Gross, E. Runge, and O. Heinonen, *Many-Particle Theory* (Adam Hilger, Bristol, 1991).

[30] W. Greiner and J. Maruhn, *Nuclear Models* (Springer-Verlag, Berlin, 1996).

[31] J. W. Negele, Rev. Mod. Phys. **54**, 913 (1982).

[32] T. H. R. Skyrme, Nucl. Phys. **9**, 615 (1958).

[33] T. H. R. Skyrme, Philos. Mag. **1**, 1043 (1956).

[34] J. Le Bloas, M.-H. Koh, P. Quentin, L. Bonneau, and J. I. A. Ithnin, Phys. Rev. C **84**, 014310 (2011).

[35] C. Titin-Schnaider and P. Quentin, Phys. Lett. B **49**, 397 (1974).

[36] J. Crank and P. Nicolson, Math. Proc. Cambridge Philos. Soc. **43**, 50 (1947).

[37] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users' Guide*, 3rd ed. (Society for Industrial and Applied Mathematics, Philadelphia, PA, 1999).

[38] S. Jiang, Ph.D. thesis, Courant Institute of Mathematical Sciences, New York University, 2001.

[39] S. Jiang and L. Greengard, Comput. Math. Appl. **47**, 955 (2004).

[40] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions: with Formulas, Graphs, and Mathematical Tables*, 3rd ed. (Dover, New York, 1965).

[41] A. Erdelyi, *Tables of Integral Transforms* (McGraw-Hill, New York, 1954).

[42] B. Mayfield, Ph.D. thesis, University of Rhode Island, Providence, 1989.

[43] D. G. Duffy, *Transform Methods for Solving Partial Differential Equations*, 2nd ed. (Chapman and Hall, London, 2004).

[44] L. Slater, *Confluent Hypergeometric Functions* (Cambridge University Press, Cambridge, 1960).

[45] DLMF, NIST Digital Library of Mathematical Functions, http://dlmf.nist.gov/, Release 1.0.5 of 2012-10-01, online companion to [52].

[46] B. Alpert, L. Greengard, and T. Hagstrom, SIAM J. Numer. Anal. **37**, 1138 (2000).

[47] C. Pardi, Ph.D. thesis, University of Surrey, 2013.

[48] D. E. Muller, Mathematical Tables and Other Aids to Computation **10**, 208 (1956).

[49] F. Byron and R. Fuller, *Mathematics of Classical and Quantum Physics* (Dover, New York, 1992).

[50] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed. (Cambridge University Press, Cambridge, 2007).

[51] D. Knuth, *The Art of Computer Programming: Fundamental Algorithms* (Addison-Wesley, London, 1968).

[52] *NIST Handbook of Mathematical Functions*, F. W. J. Olver, D. W. Lozier, R. F. Boisvert, and C. W. Clark (Cambridge University Press, New York, 2010), print companion to [45].

[53] W. J. Lentz, Appl. Opt. **15**, 668 (1976).

[54] I. J. Thompson and A. R. Barnett, J. Comput. Phys. **64**, 490 (1986).

[55] M. Heinen and H.-J. Kull, Laser Phys. **20**, 581 (2010).

[56] S. Jiang and L. Greengard, Commun. Pure Appl. Math. **61**, 261 (2008).

[57] J. A. Maruhn, P. G. Reinhard, P. D. Stevenson, J. R. Stone, and M. R. Strayer, Phys. Rev. C **71**, 064328 (2005).

[58] M. P. Brine, P. D. Stevenson, J. A. Maruhn, and P.-G. Reinhard, Int. J. Mod. Phys. E **15**, 1417 (2006).

[59] A. S. Umar and V. E. Oberacker, Phys. Rev. C **73**, 054607 (2006).

[60] C. Besse, SIAM J. Numer. Anal. **42**, 934 (2005).