PHYSICAL REVIEW C **95**, 044302 (2017)

Coordinate-space solver for superfluid many-fermion systems with the shifted conjugate-orthogonal conjugate-gradient method

Shi Jin,[1,*] Aurel Bulgac,[1,†] Kenneth Roche,[1,2,‡] and Gabriel Wlazłowski[1,3,§]
[1]*Department of Physics, University of Washington, Seattle, Washington 98195-1560, USA*
[2]*Pacific Northwest National Laboratory, Richland, Washington 99352, USA*
[3]*Faculty of Physics, Warsaw University of Technology, Ulica Koszykowa 75, 00-662 Warsaw, Poland*

(Received 14 August 2016; revised manuscript received 26 October 2016; published 3 April 2017)

Self-consistent approaches to superfluid many-fermion systems in three dimensions (and their subsequent use in time-dependent studies) require a large number of diagonalizations of very large dimension Hermitian matrices, which results in enormous computational costs. We present an approach based on the shifted conjugate-orthogonal conjugate-gradient (COCG) Krylov method for the evaluation of the Green's function, from which we subsequently extract various densities (particle number, spin, current, kinetic energy, anomalous, etc.) of a nuclear system. The approach eschews the determination of the quasiparticle wave functions and their corresponding quasiparticle energies, which never explicitly appear in the construction of a single-particle Hamiltonian or needed for the calculation of various static nuclear properties, which depend only on densities. As benchmarks we present calculations for nuclei with axial symmetry, including the ground state of spherical (magic or semimagic) and axially deformed nuclei, the saddle point in the ^{240}Pu constrained fission path, and a vortex in the neutron star crust, and demonstrate the superior efficiency of the shifted COCG Krylov method over traditional approaches.

DOI: 10.1103/PhysRevC.95.044302

I. INTRODUCTION

Density functional theory (DFT) and other self-consistent approaches [Hartree-Fock (HF), Hartree-Fock-Bogoliubov (HFB), or Hartree–de Gennes] have played an essential role in studying the properties of most nuclei across the nuclear chart [1]. Present phenomenological nuclear energy density functionals (EDF) allow for accurate descriptions of many bulk properties of nuclei such as masses, radii and shapes, transition matrix elements, potential energy surfaces and related inertial parameters, and even nonequilibrium properties, when extended to time-dependent phenomena. The time-dependent extension is straightforward [2–4] and provides a unified approach for the study of both structure and reaction dynamics. While pairing correlations are absent in closed-shell magic nuclei, there are a lot of nuclear problems where accurate description of the pairing correlations is crucial. Examples of such problems include large-amplitude collective motion of open-shell nuclei and dynamics of vortices in neutron star crust [5–11].

The evaluation of the nuclear DFT is numerically demanding, particularly if one considers large fermionic systems in three dimensions (3D) and large deformations, without any symmetry constraints. Over the years, many iterative approaches for solving the HFB equations have been proposed, including successive diagonalizations of the HFB or HF+BCS matrices, imaginary time evolution [12,13], and gradient methods [14,15], which typically need significant memory requirements and operations of complexity $O(N^3)$, where N is the dimension of the HFB matrix. For a review of modern diagonalization software, see Ref. [16].

The most standard approach, via series of direct diagonalizations of the HFB Hamiltonian, can be divided into two main classes. In the first one, the HFB problem is formulated in the configuration space by expanding the quasiparticle states of HFB on a discrete basis of orthogonal functions, usually provided by a (deformed) harmonic oscillator (HO) basis [17,18]. Although typically very fast, this approach suffers from truncation errors that typically lead to the incorrect description of the asymptotic behavior of the system or of the large deformations of the systems, e.g., when a nucleus fissions. An HO basis does not usually provide a very effective coverage of the relevant phase space. Sometimes this can be improved by introducing wavelets [19] and complex energy Berggren states [20] to describe the continuum spectrum. Nuclear systems are adequately described when the phase space in which the dynamics occurs is properly covered by the single-particle basis wave functions. This space is characterized typically by a volume $V = L^3$, where L is a few times the nucleus radius and by a maximum single-particle momentum $p_{\text{cut}} = \hbar k_{\text{cut}}$ proportional to the Fermi momentum $\hbar k_F$. The total number of quantum states in such phase space is

$$\mathcal{N}_{PS} = 4\frac{(2p_{\text{cut}}L)^3}{(2\pi\hbar)^3} \propto k_F^3 R^3, \tag{1}$$

where the factor 4 arises from accounting for spin and isospin degrees of freedom; see the discussion in Ref. [21]. The spatial extension L (that can be different in each Cartesian direction) is chosen depending on the specific needs. These extensions in each spatial direction are different for a nucleus with very extended density tails, for the collision of two nuclei, for the fission of a heavy nucleus and its split into two fragments, or for a nucleus in the presence or absence of a vortex in the neutron

---

[*] js1421@uw.edu
[†] bulgac@uw.edu
[‡] kenneth.roche@pnl.gov
[§] gabrielw@if.pw.edu.pl

star crust. In an HO basis, by increasing the size of the basis set to cover the required spatial volume, one usually goes well beyond p_{cut}, leading to an inefficient coverage of the needed phase space. Pairing correlations typically lead to a filling of all the momentum states up to p_{cut}. The need to describe large deformations, the tails of the density distributions, and particularly the large amplitude dynamics of various nuclear systems requires large spatial simulation volumes and a large momentum cutoff, thus resulting in a large number of phase-space cells $\mathcal{N}_{PS}$.

Another approach is the direct HFB matrix diagonalization in the coordinate space with a lattice spacing $dx = \pi/k_{\text{cut}}$ chosen to ensure an adequate coverage of the phase space [13,21,22]. Thus, one can obtain numerically accurate results for weakly bound nuclei and large deformations [23,24]. A number of coordinate-space HFB solvers have been published over the years [19,25–30], but solving the HFB equations in full three-dimensional (3D) coordinate space is still a challenging problem because of the large dimension of the HFB matrix discretized in a large box. To put this in perspective, consider calculations in medium size volume $32 \times 32 \times 48$ fm^3 with lattice resolution $dx = 1$ fm corresponding to $p_{\text{cut}} = \hbar\pi/dx \approx 600$ MeV/c. Then, the $N \times N$ HFB matrix has $N^2 = (4 \times 32 \times 32 \times 48)^2 \approx 200\,000^2$ matrix elements, and requires more than 0.5 TB memory just to store it. A typical diagonalization (which requires $O(N^3)$ operations separately for protons and neutrons) takes about 40 min using the high-performance linear algebra library SCALAPACK [31] on the Edison supercomputer at NERSC with 36 864 processor cores and a charged computational cost of 49 152 CPU hours [32] (CPU, central processing unit). For self-consistent convergence ~100 diagonalizations are typically required, generating an enormous computational cost on the order of about 10 million CPU hours per converged calculation. After each diagonalization, the eigenfunctions are reduced to local densities needed to construct new quasiparticle Hamiltonian matrix coefficients before the start of a next iteration. The manner in which new matrix coefficients are constructed reflects the fact that, for density functional theory, the many-body wave function contains vastly more information than is needed in each iteration.

Here we present a new method for extracting densities directly from the HFB Hamiltonian without calculating wave functions. The method is especially well suited for large-scale calculations that inevitably require an efficient use of supercomputers. It is important to compare both the computational (numerical) complexity of different algorithms that solve the same problem, and the ease with which the methods can be partitioned into smaller problems that can be effectively executed in parallel so that the entire process scales well on today's computers. An important indicator that characterizes this property of an given algorithm is the *strong scaling*, which describes how the time to compute a fixed problem depends on the aggregated scale of the computing resources used to finish the problem. Ideal (linear) strong scaling for an algorithm is achieved when the (wall) time to completion can be reduced by a constant factor k while increasing the aggregated machine scale by the same factor k. The dense, direct eigenvalue decomposition based on data decomposing the Hamiltonian over a set of processes does not exhibit perfect strong scaling on modern parallel computers, particularly for very large matrices, when communication between computational nodes starts dominating the computational cost. The factorization requires frequent interruptions both in communicating partial results and coordinating coarse phases of the algorithm between computing units, and these synchronizations, coupled to increasingly smaller work fractions for a fixed problem, dominate the strong scaling behavior. Eventually more time is spent managing the computation than evaluating the algorithm. Moreover, a typical local energy density functional leads to a sparse HFB matrix in the coordinate representation, a feature that is not efficiently utilized by the eigenvalue decomposition with direct methods. The method proposed here removes these weaknesses. The algorithm can be constructed in a nearly communication-free manner and thus exhibits almost prefect strong scaling over the number of points in the coordinate space offering a near complete reduction of $O(N)$ in complexity when evaluating the method in parallel. The only operation that involves the HFB matrix is a matrix-vector (MV) multiplication, which can benefit easily from matrix sparsity. In our implementation the MV multiplication is extremely efficiently implemented using fast Fourier transform (FFT). The programming method is straightforward in hybrid processing models that combine traditional CPUs with hardware accelerators such as general purpose graphics processing units (GPUs) [33]. Presently, many leadership class computers are of this type.

In order to grasp the new method, consider a Hartree-Fock equation [5]

$$H\psi_k(\boldsymbol{r}) = \varepsilon_k \psi_k(\boldsymbol{r}), \tag{2}$$

where $\psi_k(\boldsymbol{r})$ is the wave function corresponding to the energy level ε_k. Without loss of generality, we assume that the eigenvalues ε_k are positive. Our goal is to compute the particle number density

$$\rho(\boldsymbol{r}) = \sum_{\varepsilon_k < \varepsilon_F} \psi_k(\boldsymbol{r})\psi_k^*(\boldsymbol{r}), \tag{3}$$

where the summation includes only states up to a fixed Fermi energy ε_F for normal systems. The density can be obtained from the Green's function $G(z,\boldsymbol{r};\boldsymbol{r}')$, defined by the linear equation (spin degrees freedom are suppressed here)

$$(z - H)G(z,\boldsymbol{r};\boldsymbol{r}') = \delta(\boldsymbol{r} - \boldsymbol{r}'), \tag{4}$$

where z is a complex number. Notice that in this equation $\boldsymbol{r}'$ can be treated as parameter and thus for a fixed value of $\boldsymbol{r}'$ the Green's function $G(z,\boldsymbol{r};\boldsymbol{r}')$ is the solution of an inhomogeneous Schrödinger equation. The formal solution of this equation is

$$G(z,\boldsymbol{r};\boldsymbol{r}') = \sum_k \frac{\psi_k(\boldsymbol{r})\psi_k^*(\boldsymbol{r}')}{z - \varepsilon_k}. \tag{5}$$

Once we obtain the Green's function, forcing $\boldsymbol{r} = \boldsymbol{r}'$, the particle density can be calculated via a contour integral

$$\rho(\boldsymbol{r}') = \frac{1}{2\pi i} \oint_C dz\, G(z,\boldsymbol{r};\boldsymbol{r}')|_{\boldsymbol{r}=\boldsymbol{r}'}. \tag{6}$$

A contour can be chosen arbitrarily, but is required to enclose only the eigenvalues in the the interval $[0,\varepsilon_F]$ on the real axis. The problem of computing the density for a fixed point $\boldsymbol{r}'$ is reduced to the calculation of the Green's function $G(z,\boldsymbol{r};\boldsymbol{r}')$ for all z's on the contour, which is equivalent to solving a set of independent linear equations (4). For any given local Hamiltonian the density at any given point $\boldsymbol{r}$ can be extracted independently of any other point, enabling this step to be executed in parallel, with no communications between processes. Thus by construction, the present method exhibits perfect strong scaling up to the number of coordinate points.

For fixed z and $\boldsymbol{r}'$ the algebraic problem (4) can be solved using iterative methods. The conjugate-orthogonal conjugate-gradient (COCG) method is a very efficient algorithm for solving a set of linear equations $Ax = b$, assuming that matrix A is symmetric and complex [34]. Recently, an extension of the method called the shifted COCG has been implemented for electron systems [35,36], nuclear shell model [37], the computation of the level density in nuclei [38], and other generalized eigenvalue problems [39–41]. The shifted variant solves simultaneously a family of algebraic problems $(A-\sigma)x^\sigma = b$ for many shifts $\sigma \in \mathbb{C}$ simultaneously, essentially with the same speed as standard COCG for a single shift. In this way, the accuracy of numerical estimation of the contour integral (6) can be refined to the desired accuracy with almost no extra calculation cost. Taking as many computing units as points in coordinate space, the computation time will be limited only by the time needed to solve this single algebraic problem.

The purpose of the present work is to introduce the procedure in the context of solving the HFB equation in 3D coordinate space. The Green's function $G(z,\boldsymbol{r};\boldsymbol{r}')$ of a HFB equation can be obtained in a similar way as solving Eq. (4) using the shifted COCG method, but for a generalized multicomponent system with pairing and spin-orbit coupling. The calculation of particle density in Eq. (6) will be generalized to all types of local densities in the realistic nuclear energy density functional (NEDF) [42,43]. With the aim of taking advantage of existing and future computational resources, we developed a highly efficient parallelized GPU code as the so-called engine of the shifted COCG iteration to replace the direct parallel diagonalization procedure in the code used in Refs. [9,44,45]. As a benchmark, full self-consistent HFB calculations are performed in this work for the ground states of spherical (magic or semimagic) as well as axially deformed nuclei. The constrained HFB (CHFB) calculation is also tested for the saddle point of ^{240}Pu in the induced fission studied in [9]. Finally, we apply the method to generate nontrivial states, relevant for astrophysical applications, containing quantized vortices and nuclear defects immersed in a superfluid neutron matter.

We emphasize that the concept of DFT is extensively used across many fields dealing with fermionic systems, like quantum chemistry, solid-state physics, ultracold fermionic gases, and many others. Here we focus only on the nuclear case, as typically the nuclear EDF is very complicated in comparison to functionals encountered in other fields. The method presented here is general and can be applied to other fermionic systems as well.

II. THEORETICAL FRAMEWORK

A. Nuclear DFT within local density approximation

Local density approximation is one of the most successful concepts introduced to DFT approaches. It was extended to superfluid systems, namely superfluid local density approximation (SLDA) [46,47], and assumes that energy density functional (EDF) $\mathcal{E}$ depends on various *local* densities. In nuclear systems, a generic EDF is represented as a sum of the kinetic $\mathcal{E}_{\text{kin}}$, the nuclear $\mathcal{E}_{\text{nuclear}}$, the Coulomb $\mathcal{E}_{\text{Coul}}$, and the pairing $\mathcal{E}_{\text{pair}}$ contributions

$$\mathcal{E} = \mathcal{E}_{\text{kin}} + \mathcal{E}_{\text{nuclear}} + \mathcal{E}_{\text{Coul}} + \mathcal{E}_{\text{pair}}. \tag{7}$$

The kinetic part depends on the kinetic densities and the effective nucleon masses

$$\mathcal{E}_{\text{kin}}(\boldsymbol{r}) = \sum_{q=n,p} \frac{\hbar^2 \tau_q(\boldsymbol{r})}{2m_q(\boldsymbol{r})}. \tag{8}$$

In calculations we included the simplest center-of-mass correction by replacing the bare nucleon mass m with $m/(1-1/A)$. The total particle number is $A = N + Z$ where N and Z are respectively the neutron (n) and proton (p) numbers. The Coulomb contribution is composed of the direct part and the exchange part

$$\begin{aligned}\mathcal{E}_{\text{Coul}}(\boldsymbol{r}) &= \mathcal{E}^d_{\text{Coul}}(\boldsymbol{r}) + \mathcal{E}^e_{\text{Coul}}(\boldsymbol{r}) \\ &= \frac{e^2}{2}\int \frac{\rho_p(\boldsymbol{r})\rho_p(\boldsymbol{r}')}{|\boldsymbol{r}-\boldsymbol{r}'|} d^3\boldsymbol{r}' - \frac{3e^2}{4}\left(\frac{3}{\pi}\right)^{1/3} \rho_p^{4/3}(\boldsymbol{r}).\end{aligned} \tag{9}$$

The pairing energy in SLDA depends on the local anomalous density,

$$\mathcal{E}_{\text{pair}}(\boldsymbol{r}) = \sum_{q=n,p} g_{\text{eff}}(\boldsymbol{r}) |\nu_q(\boldsymbol{r})|^2, \tag{10}$$

and the effective pairing coupling strength $g_{\text{eff}}(\boldsymbol{r})$ is obtained via a renormalization [46–48] of the bare pairing strength, typically parametrized as

$$g_0(\boldsymbol{r}) = g_0\left[1 - \alpha\frac{\rho(\boldsymbol{r})}{\rho_0}\right], \tag{11}$$

where $\rho_0 = 0.16\,\text{fm}^{-3}$ is the saturation density. The parameter $\alpha = 0,1,1/2$ corresponds to volume, surface, and mixed pairing respectively [42,49].

The nuclear part is the most complicated. Over the years, many forms of the nuclear functional have been proposed; see Refs. [1,50,51] for review. Typically, the functional depends on various proton and neutron densities, including normal $\rho(\boldsymbol{r})$, kinetic $\tau(\boldsymbol{r})$, spin $\boldsymbol{s}(\boldsymbol{r})$, or spin kinetic energy densities $\boldsymbol{T}(\boldsymbol{r})$. In a high-accuracy nuclear EDF (NEDF) various currents are present as well, such as the normal current $\boldsymbol{j}(\boldsymbol{r})$, and the spin current densities $\boldsymbol{J}(\boldsymbol{r})$. In our previous works [9,44,45] we used the popular parametrization SLy4 [1,43] of the Skyrme NEDF, that has a rather generic form,

$$\begin{aligned}\mathcal{E}_{\text{Skyrme}} &= \mathcal{E}_{\rho^2} + \mathcal{E}_{\rho^\gamma} + \mathcal{E}_{\rho\Delta\rho} + \mathcal{E}_{\rho\tau} + \mathcal{E}_{\rho\nabla J} \\ &= \sum_{t=0,1}\big(C_t^\rho \rho_t^2 + C_t^\gamma \rho_t^2 \rho_0^\gamma + C_t^{\rho\Delta\rho} \rho_t \Delta\rho_t \\ &\quad + C_t^\tau \rho_t \tau_t + C_t^{\nabla J} \rho_t \nabla\cdot\boldsymbol{J}_t\big),\end{aligned} \tag{12}$$

where $\rho_0 = \rho_n + \rho_p$ and $\rho_1 = \rho_n - \rho_p$ (and similar for $\tau_{0,1}$ and $\boldsymbol{J}_{0,1}$) are isoscalar and isovector densities respectively, and C's are coupling constants.

Starting from the NEDF defined above, the derived HFB equation is a 4-component eigenvalue problem:

$$H\begin{pmatrix} u_{k\uparrow} \\ u_{k\downarrow} \\ v_{k\uparrow} \\ v_{k\downarrow} \end{pmatrix} = E_k \begin{pmatrix} u_{k\uparrow} \\ u_{k\downarrow} \\ v_{k\uparrow} \\ v_{k\downarrow} \end{pmatrix},$$

$$H = \begin{pmatrix} h_{\uparrow\uparrow} - \mu & h_{\uparrow\downarrow} & 0 & \Delta \\ h_{\downarrow\uparrow} & h_{\downarrow\downarrow} - \mu & -\Delta & 0 \\ 0 & -\Delta^* & -h^*_{\uparrow\uparrow} + \mu & -h^*_{\uparrow\downarrow} \\ \Delta^* & 0 & -h^*_{\downarrow\uparrow} & -h^*_{\downarrow\downarrow} + \mu \end{pmatrix}, \tag{13}$$

where we have suppressed the spatial coordinate $\boldsymbol{r}$ and k is the label of each quasiparticle wave function $[u_{k\sigma}(\boldsymbol{r}), v_{k\sigma}(\boldsymbol{r})]$, where $\sigma = \uparrow,\downarrow$. The local particle-hole Hamiltonian h is obtained by taking the appropriate functional derivatives of the energy density functional. For the Skyrme functional (12), it takes the form [5]

$$h_{\sigma,\sigma'}(\boldsymbol{r}) = \left(-\nabla \cdot \frac{\hbar^2}{2m^*}\nabla + U\right)\delta_{\sigma,\sigma'} - i\boldsymbol{W} \cdot (\nabla \times \boldsymbol{\sigma})_{\sigma,\sigma'}, \tag{14}$$

where $m^*(\boldsymbol{r})$ is the effective mass, $U(\boldsymbol{r})$ is the central part of the mean-field potential, and $\boldsymbol{W}(\boldsymbol{r})$ is the spin-orbit potential (for their explicit forms, see Ref. [43]). The local pairing field $\Delta(\boldsymbol{r})$ is defined as a function of the anomalous density

$$\Delta(\boldsymbol{r}) = -g_{\text{eff}}(\boldsymbol{r})\nu(\boldsymbol{r}). \tag{15}$$

The HFB Hamiltonian is a functional of local densities, which are determined by the quasiparticle wave functions $[u_{k\sigma}(\boldsymbol{r}), v_{k\sigma}(\boldsymbol{r})]$. The explicit expressions for the most important ones are

$$\rho(\mathbf{r}) = \sum_{k,\sigma} v^*_{k\sigma}(\mathbf{r})v_{k\sigma}(\mathbf{r}), \tag{16}$$

$$\nu(\boldsymbol{r}) = \sum_{k} v^*_{k\uparrow}(\boldsymbol{r})u_{k\downarrow}(\boldsymbol{r}), \tag{17}$$

$$\tau(\boldsymbol{r}) = \sum_{k,\sigma} \nabla v^*_{k\sigma}(\mathbf{r}) \cdot \nabla v_{k\sigma}(\mathbf{r}), \tag{18}$$

$$\boldsymbol{J}(\boldsymbol{r}) = \frac{1}{2i}(\nabla - \nabla') \times \boldsymbol{s}(\boldsymbol{r},\boldsymbol{r}')|_{\boldsymbol{r}=\boldsymbol{r}'}, \tag{19}$$

where

$$\begin{aligned} s_x(\boldsymbol{r},\boldsymbol{r}') &= \sum_k [v^*_{k\uparrow}(\boldsymbol{r})v_{k\downarrow}(\boldsymbol{r}') + v^*_{k\downarrow}(\boldsymbol{r})v_{k\uparrow}(\boldsymbol{r}')], \\ s_y(\boldsymbol{r},\boldsymbol{r}') &= i\sum_k [v^*_{k\uparrow}(\boldsymbol{r})v_{k\downarrow}(\boldsymbol{r}') - v^*_{k\downarrow}(\boldsymbol{r})v_{k\uparrow}(\boldsymbol{r}')], \\ s_z(\boldsymbol{r},\boldsymbol{r}') &= \sum_k [v^*_{k\uparrow}(\boldsymbol{r})v_{k\uparrow}(\boldsymbol{r}') - v^*_{k\downarrow}(\boldsymbol{r})v_{k\downarrow}(\boldsymbol{r}')]. \end{aligned} \tag{20}$$

The summations over k should be performed for quasiparticle states with quasiparticle energies E_k that satisfy $0 < E_k < E_{\text{cut}}$, where E_{cut} is the energy cutoff related to the momentum cutoff $E_{\text{cut}} = \hbar^2 k^2_{\text{cut}}/(2m)$ (m stands for the mass of nucleon). It should be sufficiently large to ensure the convergence of all observables. This convention is applied to all summations over k throughout this paper. In the next section, we will show how to extract these densities directly from the Green's function without explicit diagonalization of the HFB matrix. The method can be extended to other (not listed) densities.

B. Green's function and local densities

Denoting the 4×4 HFB matrix in Eq. (13) as H, the Green's function $G(\boldsymbol{r},\boldsymbol{r}',z)$ is the solution of a matrix equation,

$$(zI_4 - H)G(z,\boldsymbol{r};\boldsymbol{r}') = \delta(\boldsymbol{r} - \boldsymbol{r}')I_4, \tag{21}$$

where I_4 stands for a 4×4 unit matrix. Now the Green's function is a 4×4 blocked matrix in the form

$$G(z,\boldsymbol{r};\boldsymbol{r}') = \sum_k \frac{1}{z - E_k}\begin{pmatrix} u_{k\uparrow}(\boldsymbol{r}) \\ u_{k\downarrow}(\boldsymbol{r}) \\ v_{k\uparrow}(\boldsymbol{r}) \\ v_{k\downarrow}(\boldsymbol{r}) \end{pmatrix}\begin{pmatrix} u^*_{k\uparrow}(\boldsymbol{r}') \\ u^*_{k\downarrow}(\boldsymbol{r}') \\ v^*_{k\uparrow}(\boldsymbol{r}') \\ v^*_{k\downarrow}(\boldsymbol{r}') \end{pmatrix}^T. \tag{22}$$

In the above equation the summation over k includes all eigenstates. As we discussed in Sec. II A, we need to calculate the normal, anomalous, kinetic, and spin-orbit densities and to construct the HFB matrix in the self-consistent iterations. A closer look at the explicit expressions for the densities (16)–(20) reveals that we need to extract only 5 of 16 entries, containing $\{v_{k\uparrow}(\boldsymbol{r}), v_{k\downarrow}(\boldsymbol{r})\} \otimes \{v^*_{k\uparrow}(\boldsymbol{r}'), v^*_{k\downarrow}(\boldsymbol{r}')\}$ and $v^*_{k\uparrow}(\boldsymbol{r})u_{k\downarrow}(\boldsymbol{r}')$ products. In the next subsections, we provide prescriptions for extraction of the local densities assuming that Green's function can be efficiently computed. For simplification of the formulas we introduce the notation

$$G_{\phi\rho,\psi\sigma}(z,\boldsymbol{r};\boldsymbol{r}') = \sum_k \frac{\phi_{k\rho}(\boldsymbol{r})\psi^*_{k\sigma}(\boldsymbol{r}')}{z - E_k} \tag{23}$$

for submatrices of the Green's function, where $\phi,\psi = \{u,v\}$ are wave-function coordinates and $\rho,\sigma = \{\uparrow,\downarrow\}$ are spin coordinates.

1. *Normal and anomalous density*

The simplest products in the Green's function are the normal density $\rho(\boldsymbol{r})$ in Eq. (16) and the anomalous density $\nu(\boldsymbol{r})$ in Eq. (17). For $\rho(\boldsymbol{r})$, one needs to extract the $G_{v\uparrow,v\uparrow}$ and $G_{v\downarrow,v\downarrow}$ components from the G matrix and add them. Next, by performing the contour integral, like in Eq. (6), we obtain the normal density for a selected $\boldsymbol{r}'$ point

$$\rho(\boldsymbol{r}') = \frac{1}{2\pi i}\sum_\sigma \oint_C dz\, G_{v\sigma,v\sigma}(z,\boldsymbol{r};\boldsymbol{r}')|_{\boldsymbol{r}=\boldsymbol{r}'}, \tag{24}$$

where the contour integral encompasses the interval $[0,E_{\text{cut}}]$ of the real axis. The expression for the anomalous density requires only the $G_{u\downarrow,v\uparrow}$ component and by analogy reads

$$\nu(\boldsymbol{r}') = \frac{1}{2\pi i}\oint_C dz\, G_{u\downarrow,v\uparrow}(z,\boldsymbol{r};\boldsymbol{r}')|_{\boldsymbol{r}=\boldsymbol{r}'}. \tag{25}$$

The extension to spin densities (20) is straightforward.

2. Kinetic density

Without losing generality, we suppress the spin degrees of freedom in the equations and write the (normal) Eq. (16) and the kinetic energy densities Eq. (18) in a simplified form:

$$\begin{aligned}\rho(\boldsymbol{r}) &= \sum_k \psi_k^*(\boldsymbol{r})\psi_k(\boldsymbol{r}),\\ \tau(\boldsymbol{r}) &= \sum_k \nabla\psi_k^*(\boldsymbol{r})\cdot\nabla\psi_k(\boldsymbol{r}).\end{aligned} \tag{26}$$

In order to evaluate $\tau(\boldsymbol{r})$, we first calculate the Laplacian of the number density:

$$\Delta\rho(\boldsymbol{r}) = 2\mathrm{Re}\left[\sum_k \psi_k^*(\boldsymbol{r})\Delta\psi_k(\boldsymbol{r}) + \nabla\psi_k^*(\boldsymbol{r})\cdot\nabla\psi_k(\boldsymbol{r})\right]. \tag{27}$$

Then $\tau(\boldsymbol{r})$ can be obtained via

$$\tau(\boldsymbol{r}) = \frac{1}{2}\Delta\rho(\boldsymbol{r}) - \mathrm{Re}\left(\sum_k \psi_k^*(\boldsymbol{r})\Delta\psi_k(\boldsymbol{r})\right). \tag{28}$$

Now the problem is reduced to the calculation of the quantity $\sum_k \psi_k^*(\boldsymbol{r})\Delta\psi_k(\boldsymbol{r})$. Recall that

$$\sum_k \psi_k(\boldsymbol{r})\psi_k^*(\boldsymbol{r}') = \frac{1}{2\pi i}\oint_C dz G(z,\boldsymbol{r};\boldsymbol{r}') \tag{29}$$

and if we apply the Laplacian Δ on $\boldsymbol{r}$ on both sides of the equation and set $\boldsymbol{r} = \boldsymbol{r}'$ we obtain

$$\sum_k \Delta\psi_k(\boldsymbol{r})\psi_k^*(\boldsymbol{r}')|_{\boldsymbol{r}=\boldsymbol{r}'} = \frac{1}{2\pi i}\oint_c dz\Delta G(z,\boldsymbol{r};\boldsymbol{r}')|_{\boldsymbol{r}=\boldsymbol{r}'}. \tag{30}$$

Computation of the derivatives of the Green's functions introduces significant numerical costs. We avoid the calculation of the Laplacian by taking advantage of an integration by parts

$$\begin{aligned}\Delta G(z,\boldsymbol{r};\boldsymbol{r}')|_{\boldsymbol{r}=\boldsymbol{r}'} &= \int d\boldsymbol{r}\delta(\boldsymbol{r}-\boldsymbol{r}')\Delta G(z,\boldsymbol{r};\boldsymbol{r}')\\ &= \int d\boldsymbol{r}(\Delta\delta(\boldsymbol{r}-\boldsymbol{r}'))G(z,\boldsymbol{r};\boldsymbol{r}')\end{aligned} \tag{31}$$

where the second equality uses the boundary condition $\lim_{r\to\infty} G(z,\boldsymbol{r};\boldsymbol{r}') = 0$. Finally

$$\begin{aligned}&\sum_k \psi_k^*(\boldsymbol{r}')\Delta\psi_k(\boldsymbol{r}')\\ &\quad= \frac{1}{2\pi i}\oint_C dz\int d\boldsymbol{r}[\Delta\delta(\boldsymbol{r}-\boldsymbol{r}')]G(z,\boldsymbol{r};\boldsymbol{r}'),\end{aligned} \tag{32}$$

which is a contour integral of a convolution. Therefore, to obtain the total kinetic energy density Eq. (18), one just needs to extract $G_{v\uparrow,v\uparrow}$ and $G_{v\downarrow,v\downarrow}$ components from the G matrix, and calculate the kinetic energy density $\tau = \tau_\uparrow + \tau_\downarrow$ based on Eqs. (28) and (32). When implemented numerically $\Delta\delta(\boldsymbol{r}-\boldsymbol{r}')$ is the numerical implementation of the Laplacian applied to the δ function on the lattice.

3. Spin-current density

The spin-current density Eq. (19) can be written in the explicit form:

$$\begin{aligned}J_x(\boldsymbol{r}) &= \mathrm{Im}\sum_k\left[v_{k\downarrow}^*(\boldsymbol{r})\frac{\partial}{\partial y}v_{k\downarrow}(\boldsymbol{r}) - v_{k\uparrow}^*(\boldsymbol{r})\frac{\partial}{\partial y}v_{k\uparrow}(\boldsymbol{r})\right]\\ &\quad+\mathrm{Re}\sum_k\left[v_{k\uparrow}^*(\boldsymbol{r})\frac{\partial}{\partial z}v_{k\downarrow}(\boldsymbol{r}) - v_{k\downarrow}^*(\boldsymbol{r})\frac{\partial}{\partial z}v_{k\uparrow}(\boldsymbol{r})\right],\\ J_y(\boldsymbol{r}) &= \mathrm{Im}\sum_k\left[v_{k\uparrow}^*(\boldsymbol{r})\frac{\partial}{\partial x}v_{k\uparrow}(\boldsymbol{r}) + v_{k\downarrow}^*(\boldsymbol{r})\frac{\partial}{\partial x}v_{k\downarrow}(\boldsymbol{r})\right.\\ &\quad\left.- v_{k\downarrow}^*(\boldsymbol{r})\frac{\partial}{\partial z}v_{k\uparrow}(\boldsymbol{r}) - v_{k\uparrow}^*(\boldsymbol{r})\frac{\partial}{\partial z}v_{k\downarrow}(\boldsymbol{r})\right],\\ J_z(\boldsymbol{r}) &= \mathrm{Im}\sum_k\left[v_{k\downarrow}^*(\boldsymbol{r})\frac{\partial}{\partial y}v_{k\uparrow}(\boldsymbol{r}) + v_{k\uparrow}^*(\boldsymbol{r})\frac{\partial}{\partial y}v_{k\downarrow}(\boldsymbol{r})\right]\\ &\quad+\mathrm{Re}\sum_k\left[v_{k\downarrow}^*(\boldsymbol{r})\frac{\partial}{\partial x}v_{k\uparrow}(\boldsymbol{r}) - v_{k\uparrow}^*(\boldsymbol{r})\frac{\partial}{\partial x}v_{k\downarrow}(\boldsymbol{r})\right],\end{aligned} \tag{33}$$

which are combinations of the quantities $\sum_k \psi_k^*\nabla\psi_k$ for different spin combinations. The evaluation procedure in this case is similar to the one we used for the density $\sum_k \psi_k^*\Delta\psi_k$ and we obtain

$$\begin{aligned}&\sum_k \psi_k^*(\boldsymbol{r}')\nabla\psi_k(\boldsymbol{r}')\\ &\quad= -\frac{1}{2\pi i}\oint_C dz\int d\boldsymbol{r}[\nabla\delta(\boldsymbol{r}-\boldsymbol{r}')]G(z,\boldsymbol{r};\boldsymbol{r}'),\end{aligned} \tag{34}$$

where ∇ represents the gradient operator on $\boldsymbol{r}$. One needs to extract all spin components $G_{v\rho,v\sigma}$ from the G matrix and form the appropriate combinations.

III. NUMERICAL IMPLEMENTATION

A. Shifted COCG method

Throughout our theoretical framework, the core problem is to solve Eq. (21) for all z's on the contour. We solve these equations separately for each coordinate point $\boldsymbol{r}'$. When discretized, this problem is reduced to the linear equations for a given set of contour points z_m ($m = 0,1,\ldots,m_{\max}$). These linear equations are called shifted linear equations or shifted linear systems because the matrices $z_m - H$ are connected by scalar shifts. We solve this problem by using the shifted COCG method [35,36], which is an iterative method for solving large-scale shifted linear systems with symmetric matrices. The details of this algorithm are available in Ref. [35,36] and here we give only a brief review and illustrate how to apply it to the HFB matrix in Eq. (13).

For a given symmetric matrix A, we want to solve the linear equation

$$A\boldsymbol{x} = \boldsymbol{b}, \tag{35}$$

which is called the reference system, and its shifted equations

$$(A + \sigma I)\boldsymbol{x}^\sigma = \boldsymbol{b}, \tag{36}$$

where σ is a scalar complex shift and $\boldsymbol{x}^{\sigma}$ is the solution of the corresponding shifted system. The reference system is solved by the COCG method [34]. We define $\boldsymbol{x}_n$ as the approximate solution in the nth iteration, $\boldsymbol{r}_n$ as the corresponding residual vector $\boldsymbol{r}_n = \boldsymbol{b} - A\boldsymbol{x}_n$, search direction vector $\boldsymbol{p}_n$, and other coefficients α_n,β_n. With the initial condition $\boldsymbol{x}_0 = \boldsymbol{0}$, $\boldsymbol{r}_0 = \boldsymbol{b}$, $\boldsymbol{p}_0 = \boldsymbol{b}$, $\alpha_{-1} = 1$, $\beta_{-1} = 0$ we have to perform the following iterations:

$$\alpha_{n-1} = \frac{\boldsymbol{r}_{n-1}^T \cdot \boldsymbol{r}_{n-1}}{\boldsymbol{p}_{n-1}^T \cdot A\boldsymbol{p}_{n-1}}, \tag{37}$$

$$\boldsymbol{x}_n = \boldsymbol{x}_{n-1} + \alpha_{n-1}\boldsymbol{p}_{n-1}, \tag{38}$$

$$\boldsymbol{r}_n = \boldsymbol{r}_{n-1} - \alpha_{n-1}A\boldsymbol{p}_{n-1}, \tag{39}$$

$$\beta_{n-1} = \frac{\boldsymbol{r}_n^T \cdot \boldsymbol{r}_n}{\boldsymbol{r}_{n-1}^T \cdot \boldsymbol{r}_{n-1}}, \tag{40}$$

$$\boldsymbol{p}_n = \boldsymbol{r}_n + \beta_{n-1}\boldsymbol{p}_{n-1} \tag{41}$$

where the T represents transpose only, without complex conjugation, and $\cdot$ stands for the implied scalar product between left and right vectors. Note, the evaluation of $A\boldsymbol{p}_{n-1}$ is the most computationally consuming part of an iteration, as it requires a matrix-vector multiplication. The residual vector measures the accuracy of solution in the nth iteration and it is used as a breaking condition for iterations.

Shifted systems Eq. (36) can also be solved by COCG algorithm. For each shifted equation we introduce the corresponding vectors $\boldsymbol{x}_n^{\sigma}$, $\boldsymbol{r}_n^{\sigma}$, and $\boldsymbol{p}_n^{\sigma}$ and coefficients α_n^{σ} and β_n^{σ}, and initialize them with the same initial conditions as the reference system. However, the iterations (37)–(41) for the shifted systems can benefit from their collinearity, meaning that the residuals vectors for the reference and shifted systems are connected [35]

$$\boldsymbol{r}_n^{\sigma} = \frac{1}{\pi_n^{\sigma}}\boldsymbol{r}_n, \tag{42}$$

where the proportionality constant in each iteration is given by ($\pi_0^{\sigma} = \pi_{-1}^{\sigma} = 1$)

$$\pi_n^{\sigma} = \left(1 + \alpha_{n-1}\sigma + \alpha_{n-1}\frac{\beta_{n-2}}{\alpha_{n-2}}\right)\pi_{n-1}^{\sigma} - \alpha_{n-1}\frac{\beta_{n-2}}{\alpha_{n-2}}\pi_{n-2}^{\sigma}. \tag{43}$$

Thus, for the shifted systems, instead of evaluating the time-consuming Eq. (39) one can evaluate the simpler Eq. (42). Moreover, the coefficients α^{σ} and β^{σ} are also connected with corresponding coefficients of the reference system

$$\alpha_{n-1}^{\sigma} = \frac{\pi_{n-1}^{\sigma}}{\pi_n^{\sigma}}\alpha_{n-1}, \tag{44}$$

$$\beta_{n-1}^{\sigma} = \left(\frac{\pi_{n-1}^{\sigma}}{\pi_n^{\sigma}}\right)^2\beta_{n-1}. \tag{45}$$

These equations replace Eqs. (37) and (40). Often the reference system is called the seed system as it "seeds" data for shifted systems. The iterations are executed simultaneously for all systems and they end when the desired accuracy is achieved for all equations. Since when evaluating the density we set $\boldsymbol{r} = \boldsymbol{r}'$ in the Green's function in the vector Eq. (42), one needs to evaluate only one component, and not the entire vector; see also Sec. III E.

Assume for a moment that the HFB matrix is symmetric. This is the case when the spin-orbit term in the nuclear density functional is ignored and the pairing potential is real. Then we can straightforwardly let $A = z_0 I - H$, where z_0 is selected point from contour. Shifts are given by $\sigma_m = z_m - z_0$. We solve the problem on a Cartesian mesh grid with a lattice size $N_x \times N_y \times N_z$ and lattice spacing $dx = dy = dz$. After discretization, the HFB matrix has dimensions $N \times N$ with $N = 4N_xN_yN_z$, but in this case spin-up and spin-down components are decoupled and the matrix size is effectively $2N_xN_yN_z$. For a fixed spatial point $\boldsymbol{r}'$ function $\delta(\boldsymbol{r} - \boldsymbol{r}')$ becomes vector, where all N elements are zeros except one $[= 1/(dxdydz)]$, corresponding to the selected position. We denote this vector by $\delta_{\boldsymbol{r}'}(\boldsymbol{r})$. Since we do not need all elements of G matrix, it is sufficient to solve problems $A\boldsymbol{x}_1 = \boldsymbol{b}_1$ and $A\boldsymbol{x}_2 = \boldsymbol{b}_2$ (together with shifted counterparts), where $\boldsymbol{b}_1 = (0,0,\delta_{\boldsymbol{r}'},0)^T$ and $\boldsymbol{b}_2 = (0,0,0,\delta_{\boldsymbol{r}'})^T$. Then solutions $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are third and fourth columns of Eq. (22).

In general, the HFB matrix is a Hermitian but not a symmetric matrix, which cannot be solved using the shifted COCG method directly. However, one can use the COCG method designed for symmetric matrices in the case of Hermitian matrices by performing a simple matrix transformation. Assume that H is an arbitrary Hermitian matrix which can be divided into real and imaginary parts: $H = H_x + iH_y$, with obvious symmetry properties $H_x = H_x^T$ and $H_y = -H_y^T$. If we also divide the eigenvectors of H into real and imaginary parts as $\psi = x + iy$. The eigenvalue problem $H\psi = \lambda\psi$ will be converted to

$$(H_x + iH_y)(x + iy) = \lambda(x + iy). \tag{46}$$

After collecting the real and imaginary terms it will become an equivalent eigenvalue problem:

$$\begin{pmatrix} H_x & -H_y \\ H_y & H_x \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} = \lambda\begin{pmatrix} x \\ y \end{pmatrix}, \tag{47}$$

but now the matrix on the left-hand side is real symmetric. The Green's function $G = G_x + iG_y$ of a general HFB matrix H is then the solution of the equation [equivalent to Eq. (21)]

$$(zI - H')\begin{pmatrix} G_x \\ G_y \end{pmatrix} = \begin{pmatrix} \delta^{(4)} \\ 0 \end{pmatrix}, \tag{48}$$

where

$$H' = \begin{pmatrix} H_x & -H_y \\ H_y & H_y \end{pmatrix}, \quad \delta^{(4)} = \delta(\boldsymbol{r} - \boldsymbol{r}') \otimes I_4. \tag{49}$$

As before we need to solve two linear equations $A'\boldsymbol{x}_1' = \boldsymbol{b}_1'$ and $A'\boldsymbol{x}_2 = \boldsymbol{b}_2'$ (and their shifted equation) with $A' = zI - H'$ in doubled space. Now, vectors of length $2N$ need to be set for $\boldsymbol{b}_1' = (0,0,\delta_{\boldsymbol{r}'},0,0,0,0,0)^T$ and $\boldsymbol{b}_2' = (0,0,0,\delta_{\boldsymbol{r}'},0,0,0,0)^T$.

B. The form of the integration contour

Before a further discussion of the shifted COCG method, we should provide a clear definition of the contour, which determines the shifted systems and is fundamental for the convergence behavior and error control of the whole algorithm.

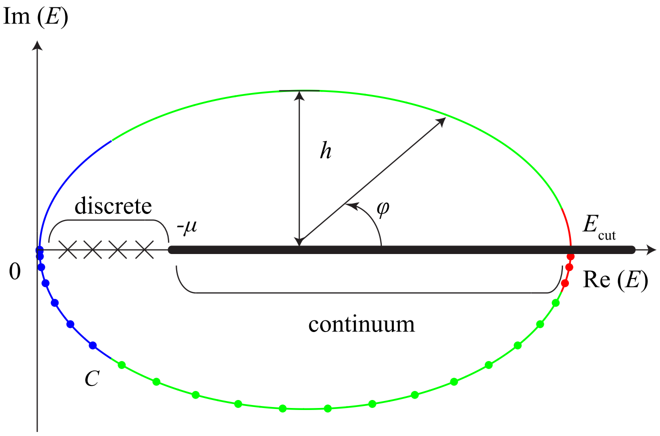


FIG. 1. The ellipse contour $C : z(\varphi)$ in Eq. (50) adopted to perform the contour integral. The thick solid line represents the positive energy continuum quasiparticle states with $E > -\mu$, while the crosses represent the positive energy discrete bound quasiparticle states $0 < E_i < -\mu$. Density of integration points is larger for parts of the contour close to the real axis (depicted by blue and red colors) and decreases as we go far away from the real axis (depicted by green color).

We need to calculate an integral of the form $\frac{1}{2\pi i}\oint_C dz f(z)$ where the contour encloses exactly a segment of the real axis $[0,E_{\rm cut}]$. It is natural to choose the contour to be symmetric with respect to the real axis since the poles (the HFB spectrum) are real. We can parametrize the contour as $z(\varphi)$ and a simple choice of its form is an ellipse:

$$\begin{aligned} z(\varphi) &= \frac{E_{\rm cut}}{2} + \frac{E_{\rm cut}}{2}\cos\varphi + ih\sin\varphi \\ &= E_{\rm cut}\cos^2\frac{\varphi}{2} + ih\sin\varphi, \end{aligned} \qquad (50)$$

where $0 \leqslant \varphi \leqslant 2\pi$ and the height of ellipse h is the parameter that should be chosen carefully; see Fig. 1.

The integrand function behaves smoothly, only for points that are far away from real axis. As the contour approaches real axis, the function $f(z) \sim \sum_k \frac{1}{z-E_k}$ as a function of z starts to exhibit fast oscillations, making accurate numerical integration difficult. For this reason, we choose nonuniformly distributed integration points along the contour. The density of points distribution depends on distance from the real axis with more points closer to the real axis, and less far away from the real axis. The height h of the eclipse is set to be significantly larger than the expected average separation between poles, and we compute the integrand function for angles φ given by a distribution function (u and the l subscript corresponds to upper and lower parts of the contour with respect to the real axis)

$$\begin{aligned} \varphi_u &= \frac{\pi}{2}\left\{1 + \tanh\left[\alpha\tan\left(\phi - \frac{\pi}{2}\right)\right]\right\}, \\ \varphi_l &= \varphi_u + \pi, \quad 0 \leqslant \phi \leqslant \pi, \end{aligned} \qquad (51)$$

where α is a parameter (usually $\alpha \sim$ 1–5) and the integral over ϕ is discretized with a set of evenly distributed points ϕ_n in the interval $[0,\pi]$ with step size $\Delta\phi$. Thus the contour integral can be converted into an weighted sum

$$\frac{1}{2\pi i}\oint_C dz f(z) \cong \sum_{\sigma=u,l;n} f\{z[\varphi_\sigma(\phi_n)]\}\, w_\sigma(\phi_n)\Delta\phi. \qquad (52)$$

Typically, the number of integration points does not exceed 10^4, and further increases do not improve the numerical result. The computation of the integrand along the contour is very fast and represents a very small fraction of the entire calculation time, which is dominated by the calculation of the Green's function at the reference point.

C. Convergence behavior

The norm of the residual vectors $\boldsymbol{r}_n$ at the reference and of the shifted systems $\boldsymbol{r}_n^\sigma$ quantify the error of the approximate solution $\boldsymbol{x}_n$ and $\boldsymbol{x}_n^\sigma$ in each iteration. Since we need to calculate the Green's functions $G(z,\boldsymbol{r};\boldsymbol{r}')$ for all z's on the contour defined in Sec. III B, the convergence behavior on the contour needs to be studied.

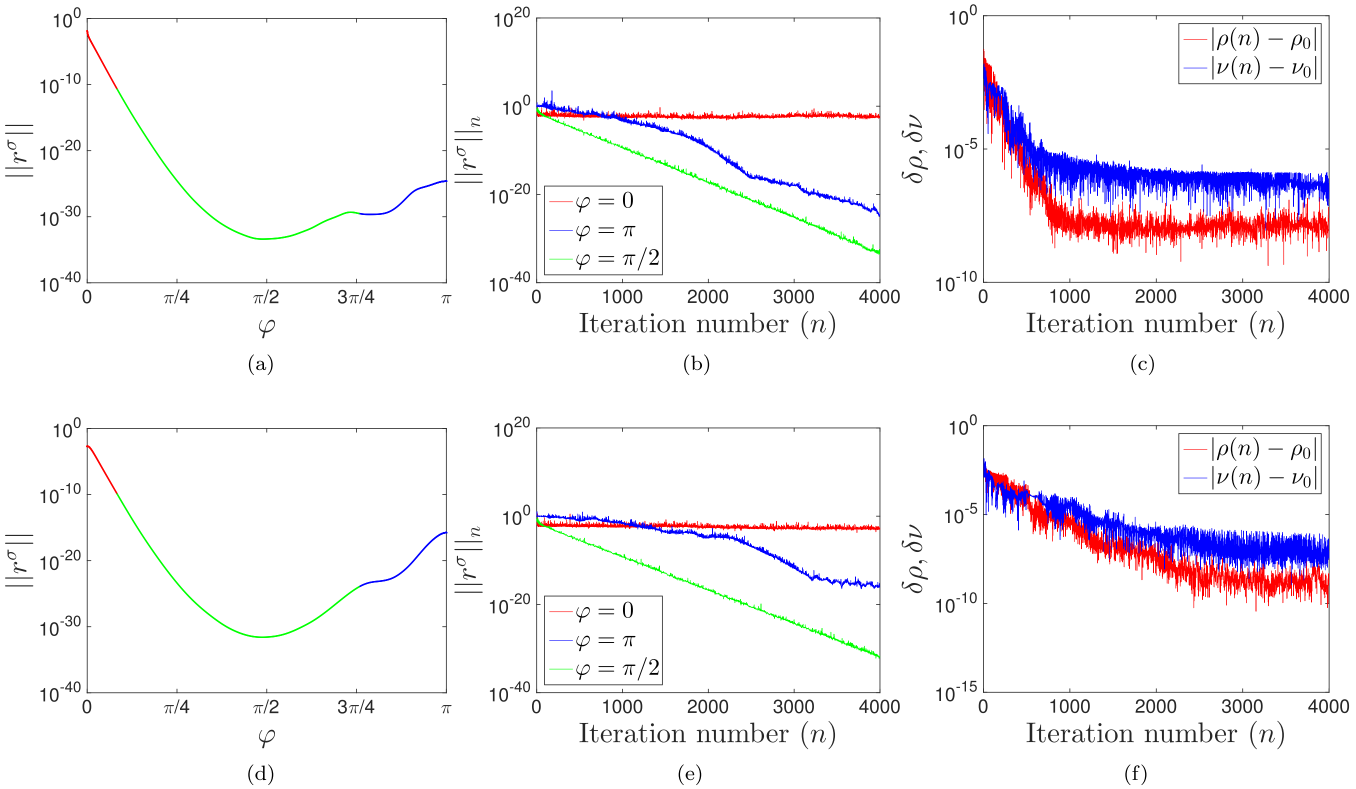
As a simple but nontrivial test case, we consider only neutrons, and choose a phenomenological Wood-Saxon (W-S) model [5,52]. The central part of the single-particle Hamiltonian Eq. (14) is given by $U(\boldsymbol{r}) = V_0 f(\boldsymbol{r})$, where $f(\boldsymbol{r})$ has the symmetrized W-S form:

$$f(\boldsymbol{r}) = \frac{1-\exp\left(\frac{-2R_0}{a}\right)}{\left[1+\exp\left(\frac{r-R_0}{a}\right)\right]\left[1+\exp\left(\frac{-r-R_0}{a}\right)\right]} \qquad (53)$$

and $V_0 = -50$ MeV, $r = \sqrt{x^2+y^2+(z/\beta)^2}$, $R_0 = 1.12A_0^{1/3} - 0.86A_0^{-1/3}$ fm, $A_0 = 100$, $a = 0.54$ fm. By varying the parameter β, we calculate the densities in a W-S potential in both spherical and axially deformed cases. The spin-orbit potential $\boldsymbol{W}(\boldsymbol{r})$ is chosen as $\boldsymbol{W}(\boldsymbol{r}) = \lambda\nabla U(\boldsymbol{r})$, where $\lambda = -0.5\ \mathrm{fm}^2$, and the effective mass is set to be the bare mass of neutron $mc^2 = 939.565$ MeV. The self-consistent pairing field $\Delta(\boldsymbol{r})$ in Eq. (15) should be determined by the anomalous density $\nu(\boldsymbol{r})$, but here we still use a phenomenological W-S potential shape $\Delta(\boldsymbol{r}) = 12/A_0^{1/2} f(\boldsymbol{r})$. In the test, the chemical potential is fixed at $\mu = -7$ MeV. The numerical tests are performed in a cubic box of size $L_x = L_y = L_z = 20$ fm for the spherical W-S model ($\beta = 1$), and a rectangular box of size $L_x = L_y = 20, L_z = 25$ fm for the deformed W-S model ($\beta = 1.5$). In both cases, the lattice constant $dx = dy = dz = 1.25$ fm. The number of lattice points is $N_x = L_x/dx$ and respectively for y and z directions. The energy cutoff is set to $E_{\rm cut} = 100$ MeV.

We find that the convergence properties of the algorithm are strongly correlated with the density of particles for a given point. For points with higher particle densities, the method needs more iterations to converge. This behavior can be easily understood if we recall that the starting point for iterations is $\boldsymbol{x}_0 = 0$; see Sec. III A. For the points where the density is higher, the convergence is noticeably slower. In the following tests, we consider the convergence properties of the method for the points requiring the highest number of iterations. These points are located close to the center of the simulation box.

Results for the spherically symmetric case ($\beta = 1$) are presented in the top row of Fig. 2. Figure 2(a) shows the distribution of $\|\boldsymbol{r}_n^\sigma\|$ on the semiellipse $z(\varphi), 0 \leqslant \varphi \leqslant \pi$ on



(a) (b) (c)

(d) (e) (f)

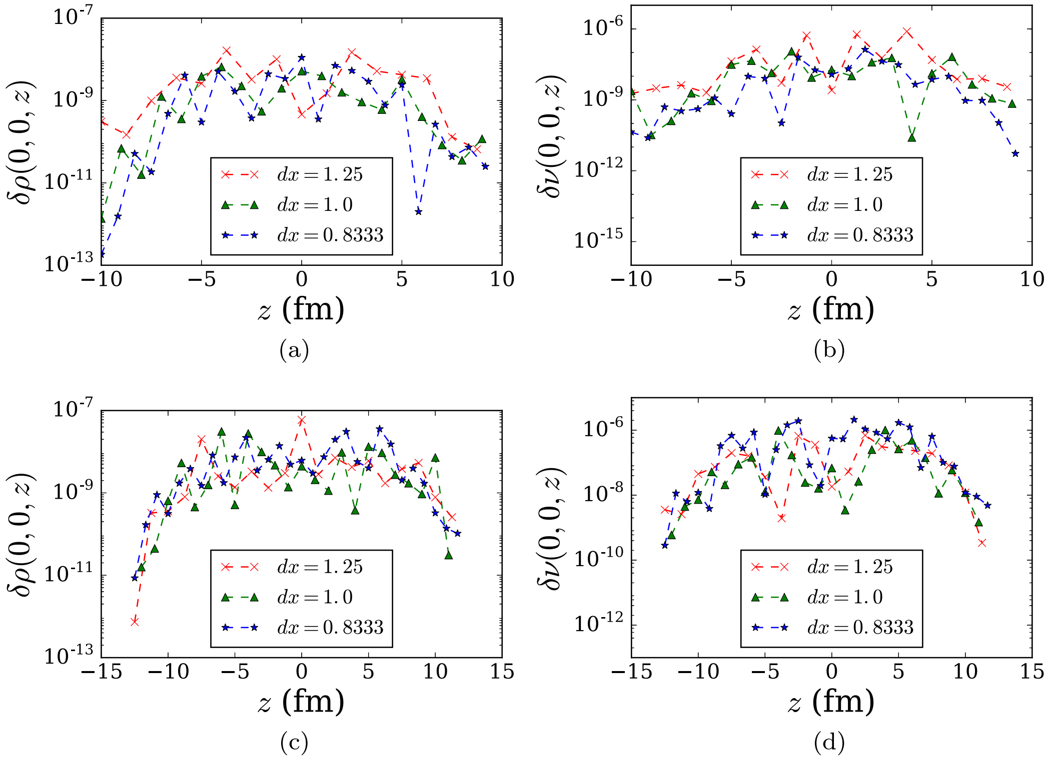
FIG. 2. The convergence behavior of the shifted COCG method for the HFB equation with W-S potential. The top and bottom rows show results for spherically symmetric and deformed W-S potentials respectively. The spherically symmetric problem was solved on lattice 16^3, while the deformed problem used a lattice $16^2 \times 20$. In both cases, the lattice spacing is $dx = 1.25$ fm. Panels (a) and (d) show the distribution of the norms of the residual vectors $\|r_n^{\sigma}\|$ on the semiellipse contour $z(\varphi)$ for iteration number $n = 4000$. Three colors (red, green, blue) correspond to different parts of the contour, as depicted on Fig. 1. Panels (b) and (e) show the convergence behavior of $\|r_n^{\sigma}\|$ for three points $z(0)$ (red), $z(\pi/2)$ (green), and $z(\pi)$ (blue) as a function of iteration number. Panels (c) and (f) show the difference between the value obtained from COCG method in the nth iteration and the exact value for the normal (red) and anomalous (blue) densities.

the upper half complex plane, for iteration number $n = 4000$, and fixed position $\boldsymbol{r}'$ located close to center of the box. In the lower half plane ($\pi \leqslant \varphi \leqslant 2\pi$), the behavior is identical due to the reflection symmetry of the integration contour. The shifted system converges most quickly in the middle part of the contour (represented by the color green), more slowly near the left end of contour $z(\pi) = 0$ (represented by the color blue), and very slowly near the right end of the contour $z(0) = E_{\text{cut}}$ (represented by the color red). A closer look at the convergence behavior at the representative points in these three parts, respectively $z(\pi/2)$, $z(\pi)$, and $z(0)$, is shown in Fig. 2(b). Besides the rapid convergence of the middle point $z(\pi/2)$, the residual at the origin $z(\pi) = 0$ also has a stable decrease with iterations. However, the iterations for the right end $z(0) = E_{\text{cut}}$ fail to converge, and $\|r_n^{\sigma}\|$ keeps oscillating around 10^{-2}. Consider the convergence analysis of the conjugate gradient (CG) method [53], in which case the convergence ratio depends on the 2-norm condition number κ of the matrix A:

$$\frac{\|r_n\|}{\|r_0\|} \leqslant 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^n, \tag{54}$$

where the condition number $\kappa(A) = |\lambda(A)|_{\max}/|\lambda(A)|_{\min}$ is the ratio of maximum and minimal absolute value of the eigenvalues of matrix A. From Eq. (54) it is clear that a larger κ leads to a slower convergence. The order of magnitude of the condition number for matrix $A = zI - H$ can be estimated as $\frac{\max|z-E_k|}{\min|z-E_k|}$, where E_k are eigenvalues of the HFB matrix from interval $[0, E_{\text{cut}}]$. Definitely, the condition number decreases as the imaginary part of z increases, thus confirming the finding that the convergence is fastest in middle part of the semiellipse. The largest values of κ occur at points close to the real axis. We know that the spectrum of a HFB matrix is discrete if $|E_k| < |\mu|$ and continuous otherwise [23,54]. Around $z = 0$ there is a gap in the spectrum and therefore for $z \to 0$ $\kappa = O(E_{\text{cut}}/\Delta)$, where Δ is average value of the pairing gap in the system. When $z \to E_{\text{cut}}$, $\min|z - E_k|$ can be arbitrarily small (only set by lattice resolution), and in the limit of zero lattice spacing dx the condition number diverges. This quasisingular matrix equation is hard to solve within a reasonable iteration number. However, these continuous states near the energy cutoff have very small occupation probabilities ($<10^{-5}$) and make negligible contributions to the local densities. While it is possible that preconditioning the COCG method can help [55–58], our efforts in this direction did not yield any advantages.

Figure 2(c) tracks the error of the particle density ρ and anomalous density ν obtained by the calculated Green's



(a) (b) (c) (d)

FIG. 3. Comparisons were made between the local densities calculated by shifted COCG iteration and the direct diagonalization approaches along z axis for fixed $x = 0$ and $y = 0$ coordinates with three different lattice constants dx. In the case with spherical symmetric potential (top row) the system size is $L_x = L_y = L_z = 20$ fm, and in the deformed case (bottom row) the system size is $L_x = L_y = 20$ and $L_z = 25$ fm. The left column shows differences for the normal density $\delta\rho = |\rho_{\mathrm{cocg}} - \rho_{\mathrm{diag.}}|$, and the right column shows them for the anomalous density $\delta\nu = |\nu_{\mathrm{cocg}} - \nu_{\mathrm{diag.}}|$.

function in each iteration. The exact value of these two densities ρ_0 ($\sim 10^{-1}$) and ν_0 ($\sim 10^{-3}$) are calculated by direct diagonalization of the same HFB matrix via SCALAPACK. It shows that the particle density ρ can converge to an accuracy of 10^{-8} within 1000 iterations, and this accuracy will not improve as the iteration goes past that. The anomalous density ν will converge to an accuracy of 10^{-6} within the same iteration count, which is quite acceptable, since the contribution to the energetics of a nucleus from the pairing field is much smaller in absolute terms. The pairing field is almost two orders of magnitude smaller than the single-particle potential. This behavior of the anomalous density is due to the fact that the high-energy quasiparticle continuum states contribute with significant weights as the anomalous density diverges when the energy cutoff is increased [46,48,54]. The same tests for deformed W-S model are presented in Figs. 2(d)– 2(f). They show a similar pattern of the residual distribution on the contour. The local densities can reach the same accuracy but with more iterations, because of the slower convergence of points near $E = 0$. In both the spherical and deformed cases, the convergence behavior at $E = 0$ can represent the convergence behavior of the final local densities, and thus is chosen to be the breaking condition for the iterations.

Until now our discussion of the convergence behavior has been for a fixed spatial point in the lattice system. To have a comprehensive view of the accuracy of this method, we also compare the difference between the local densities ρ and ν calculated with the shifted COCG method and with the direct diagonalization method on *all* lattice spatial points. Figure 3 displays the differences for spherical and deformed W-S model along the z axis for fixed $x = 0$ and $y = 0$ coordinates. Because the local densities on this spatial line have the largest magnitude throughout the box, we will also get the largest errors on this line. To study the influence of the lattice constants on the accuracy, besides the previous $dx = 1.25$, we also performed calculations with smaller lattice constants $dx = 1.0$ and $dx = 0.8333$. From the figures one can see the global maximum error of ρ and ν are respectively 10^{-8} and 10^{-6}, which matches the convergence behavior of Fig. 2. The kinetic and spin-current densities are calculated with similar accuracy as achieved for the calculation of normal density. Moreover, the accuracy is little affected by the value of the lattice constant, a fact which can be expected, as argued in Sec. I; see also Refs. [21,22].

D. Exploiting symmetries

Densities on each spatial point in the system are calculated independently. The number of points to be processed is equal to the number of lattice points $N_xN_yN_z$. This number can be easily reduced if the system exhibits symmetries, like reflection symmetries, axial symmetry, or spherical symmetry. For axially or spherically symmetric systems, one can argue that it is more profitable to exploit this symmetry directly on the level of HFB equations, i.e., assume the correct symmetry for the wave functions and solve the constrained HFB problem. However, if the solution of HFB problem will be used as initial point for an unconstrained 3D simulation, then this approach is not accurate enough. Typically, the solutions obtained from a solver which explicitly uses spherical or axial symmetries once discretized on a 3D spatial lattice are no longer orthogonal to each other and do not represent self-consistent eigenstates (with the necessary numerical accuracy for a stable numerical integration in time). Self-consistent iterations in full 3D space are still required to get a high-quality state. We demonstrate the utilization of axial symmetry for reducing the computational cost, while the underlying HFB matrix is defined in the full 3D coordinate space.

Consider the scalar local densities represented in cylindrical coordinates (r,ϕ,z):

$$\begin{aligned}\rho(\boldsymbol{r}) &= \rho(r,z),\\ \nu(\boldsymbol{r}) &= \nu(r,z),\\ \tau(\boldsymbol{r}) &= \tau(r,z).\end{aligned}$$

We only need to calculate the densities on the points with different values of (r,z)'s in the system. This reduces the number of points to be calculated to approximately $1/8$ of the total number of points in the full 3D lattice system, as only the points with $0 \leqslant x \leqslant y \leqslant L/2\,(L = L_x = L_y)$ need to be explicitly considered. Thus the performance will improve by a factor of 8. A reflection symmetry along the z axis can add another factor of 2, thus an overall speedup of 16.

We re-expand the vector density $\boldsymbol{J}(\boldsymbol{r})$ along the three unit directions of the cylindrical coordinate system as J_r, J_ϕ, J_z, which can be related to the Cartesian components J_x, J_y, J_z by the transformation

$$\begin{aligned}J_r(r,z) &= J_x\cos\phi + J_y\sin\phi,\\ J_\phi(r,z) &= J_y\cos\phi - J_x\sin\phi,\\ J_z(r,z) &= J_z,\end{aligned}\tag{55}$$

and reversely

$$\begin{aligned}J_x &= J_r\cos\phi - J_\phi\sin\phi,\\ J_y &= J_r\sin\phi + J_\phi\cos\phi,\\ J_z &= J_z.\end{aligned}\tag{56}$$

The components J_r, J_ϕ, J_z have axial symmetry, i.e., all of them depend only on r and z, while the Cartesian components J_x, J_y, and J_z depend on all three spatial coordinates. In our approach, we first calculate the Cartesian components J_x, J_y, J_z for the set of points with different (r,z), we transform them into the cylindrical components J_r, J_ϕ, J_z via Eqs. (55), and using the axial symmetry we cover all the equivalent points in Cartesian space via reverse transformation using Eqs. (56).

Numerically, we found that the axial symmetry is not strictly realized due to the finite size of the lattice constant. In the tests discussed above for a spherical or deformed W-S model, we start from a HFB Hamiltonian with strict axial symmetry and calculate the local densities on all points of the lattice system without enforcing any symmetry constraints. In theory, for example, the scalar densities (ρ, ν, τ) on the point of Cartesian coordinates (in fm) (0,5,0) should have the same value as on the point (3,4,0), but numerically they are different due to the cubic symmetry of the simulation box. The relative difference between the scalar densities on these pairs of pseudoequivalent points varies from 10^{-2} to 10^{-4} as the lattice constants decreases from $dx = 1.25$ fm to $dx = 0.8333$ fm. In order to fix this issue, we include all points in these pairs into the calculation, with a small increase of the total number of lattice points needed to be calculated.

This lattice effect becomes more pronounced when one calculates the cylindrical components of the vector density $\boldsymbol{J}(\boldsymbol{r})$. The components J_r and J_z can be treated in the same manner as the scalar densities, but J_ϕ, which is expected theoretically to vanish, suffers from a numerical noise, which oscillates around zero. The average magnitude of this noise decreases from 10^{-5} to 10^{-7} as dx decreases from 1.25 fm to 0.8333 fm and for simplicity we just force them to be zero, which will bring some discrepancy in the calculation of $\boldsymbol{J}(\boldsymbol{r})$ when comparing the COCG and the direct diagonalization approaches. We will discuss this discrepancy in realistic calculations for finite nuclei in Sec. IV and show that it vanishes, as naturally expected, as the lattice constant goes to 0.

E. Computational cost

The computational cost of the shifted COCG iteration is set by the cost of solving the reference system, as it involves matrix-vector (MV) multiplication between the matrix $A = z_0 - H$ and the vector $\boldsymbol{p}_n$ of size $2N$, where $N = 4N_xN_yN_z$. The nonlocality in the HFB matrix (see the Appendix for explicit form) comes from the Laplacian or gradient operators. When discretized, the gradient or the Laplacian of a function (vector) can be obtained either via a finite difference formula or through the Fourier transform. In both cases the operation is represented by a sparse matrix. Exploiting the sparsity accelerates the MV operations significantly. We compute derivatives using Fourier transforms, as it is a more accurate method than using finite difference formulas [22]. Moreover, we advocate the direct use of fast Fourier transforms (FFT) due to a manageable complexity $O(N\log N)$. The computation of derivatives with FFT is expected to be faster than accurate multipoint finite difference algorithms [59]. For shifted systems, the collinear theorem insures that there is no need to evaluate MV products. The vector-scalar arithmetic in Eqs. (38), (41), and (42) for $\sigma \neq 0$ makes a considerable contribution to the total computational cost if all $2Nm$ elements of $\boldsymbol{x}_n^\sigma$, $\boldsymbol{p}_n^\sigma$, $\boldsymbol{r}_n^\sigma$ are calculated, where m is the number of points on the contour. However, in order to obtain the local densities on a fixed spatial point $\boldsymbol{r}'$, we do not need to know the Green's function $G(z,\boldsymbol{r};\boldsymbol{r}')$ for all spatial points $\boldsymbol{r}$ in the system. This is because integrand function $f(z)$ requires either $G(z,\boldsymbol{r};\boldsymbol{r}')|_{\boldsymbol{r}=\boldsymbol{r}'}$ or overlap of Green's function with $\nabla\delta(\boldsymbol{r}-\boldsymbol{r}')$ or $\Delta\delta(\boldsymbol{r}-\boldsymbol{r}')$. Derivatives of $\delta(\boldsymbol{r}-\boldsymbol{r}')$ have only $l = N_x + N_y + N_z - 2$ nonzero elements (which are located on three lines in x,y,z directions that cross at $\boldsymbol{r}'$). The vector-scalar arithmetic can be executed only for elements required by the integrand functions. Finally, the size of $\boldsymbol{x}_n^\sigma$, $\boldsymbol{p}_n^\sigma$, $\boldsymbol{r}_n^\sigma$ is reduced to the order $2lm$, which is a small number when compared with N for the reference system. The cost related to solving the shifted systems turns out to be negligible. Therefore, the theoretical upper bound of the computational cost within one full self-consistent iteration of shifted COCG method at N coordinates is determined by the FFT operation $[O(N \log N)]$ and the maximum number of COCG Krylov iterations M required to converge each point in the reference system. The number of Krylov iterations to converge is problem dependent and theoretically it scales as $O(N)$, but in all of our realistic calculation we find that $M \approx 4000$ is sufficient to get a high-accuracy solution for nuclear problems. Final numerical complexity scales as $O(MN^2 \log N)$ where in practice $M \ll N$, but in the worst case M can be of the same order as N. On the other hand, the direct diagonalization scales like $O(N^3)$. It is not *a priori* clear that the COCG approach is preferred. It is at this point that leveraging the strong scalability feature of the COCG approach wins. Let us use a distinct parallel process for each point in the reference system reducing the calculation to the cost $O(MN \log N)$. Thus, within the errors of the converged results we present, the COCG approach gains a clear advantage for larger dimensional systems.

We have also investigated other algorithms that can be applied to nonsymmetric shifted linear systems, the shifted BiCG-Stab(l) [60] and the shifted GMRES [61]. In these algorithms, we do not need to play the tricks described in Sec. III A for the Hermitian matrix H and the size of matrix A will not be increased by a factor of two. But, these algorithms require two sequential matrix-vector (MV) products in each iteration; thus in total there is no profit per iteration. Among the tested methods, the COCG method exhibits so far the best convergence properties.

Finally, we emphasize that the presented method can efficiently utilize heterogeneous computers. In our implementation we perform calculations both using CPUs as well as highly efficient multithreaded GPUs. In our experience, the GPU implementation of shifted COCG is more than 50× faster than its CPU counterpart. Table I compares the charged core hours between our GPU code implementing the shifted COCG method and the CPU code using the direct diagonalization method (SCALAPACK) for two problems with different lattice dimensions. On supercomputers, this is the most relevant quantity to be compared as it essentially represents the cost that a user has to pay for calculations. For shifted COCG method, the axial symmetry described in Sec. III D is implemented. These timing tests were performed at OLCF Titan [62] and NERSC Edison [32] supercomputers. In both problems, shifted COCG is faster than diagonalization, and this advantage becomes more pronounced in the problem with larger dimension (no. 2).

IV. BENCHMARK EXAMPLES

In this section, the Green's function and the shifted COCG method, which we denote XCOCG, is benchmarked by solving the self-consistent HFB equation in 3D coordinate space and compared with the codes used in Refs. [9,44,45], which we denote XDIAG. The XDIAG code extracts wave functions of HFB Hamiltonian (13) in the discrete variable representation (DVR) basis [21] via a direct diagonalization.[1] The diagonalization procedure is executed parallel using SCALAPACK library [31]. Next, densities are formed from the wave functions using formulas (16)–(20). These operations form a single self-consistent iteration. We mix the intermediate solutions during the iteration process using a linear or a Broyden mixing algorithm [63]. The XCOCG code is a modified version of the XDIAG code where parallel diagonalization procedure and computation of densities is replaced by the COCG method. Densities are extracted according formulas provided in Sec. II B. The COCG part is ported to GPUs. Both codes provide the same results up to the accuracy specified in Sec. III C.

TABLE I. Timing comparison between the shifted COCG method and diagonalization method using SCALAPACK for solving two problems with different dimensions Ns in each self-consistent iteration. The first case is for finite nuclei ^{240}Pu in the lattice of dimension 20 × 20 × 40 (the extra factor of 4 is included for spin orbit and pairing). The second case is for the study of a nucleus immersed in a neutron superfluid sea of dimension 50 × 50 × 40, where spin orbit is ignored (the extra factor of 2 is for pairing only). In both problems, $t_{\rm cocg}$ and $t_{\rm diag}$ denote the charged core hours of the GPU code using shifted COCG method and the CPU code using SCALAPACK in one self-consistent iteration. The capital letter in parentheses denotes the computing facility where the timing test is performed: "T" represents Titan and "E" represents Edison [32,62] and the CPU hours were determined in units of CPU hours according to the corresponding policy for these supercomputers. On Titan an hour using a single computing node, which has 16 CPUs and one GPU, is charged as 30 CPU h. On Edison an hour using a CPU is charged as 2 CPU h. For problem 2 the estimated charge in case of using Titan for direct diagonalization is about 87 500 CPU h.

No.	N	$t_{\rm cocg}$ (CPU h)	$t_{\rm diag}$ (CPU h)
1	20 × 20 × 40 × 4	649.23 (T)	1,547.71 (T)
2	50 × 50 × 40 × 2	9,318.4 (T)	46,694.4 (E)

We performed tests on the nuclei with axial symmetry, including spherical and axially deformed nuclei. In all of these realistic tests, the Skyrme NEDF SLy4 [43] is used in the p-h channel, and the SLDA treatment [46,47] for the pairing interaction is used in the p-p channel with bare coupling strength $g_0(\boldsymbol{r}) = g_0 = -233$ MeV (volume pairing) and energy cutoff $E_{\rm cut} = 100$ MeV. These are typical values used in realistic calculations. We conclude this section by presenting states generated by the shifted COCG method used for studies of interaction between quantized vortices and nuclei in neutron start crust [11]. These calculations are not feasible for XDIAG code in a reasonable time.

---

[1] Direct methods yield the exact solution if the precision is unlimited whereas indirect methods may be even more accurate and are often more efficient depending on system details. Also, iterative methods more tend to naturally damp out roundoff errors that accumulate and become difficult with large N problems evaluated directly.

TABLE II. Results of HFB + SLy4 calculations for ^{208}Pb using XCOCG in a cubic box of size 32^3 fm^3 with lattice constants $dx = 1.25, 1.0, 0.8$ fm. Results obtained with the spherical 1D code HFBRAD are presented for comparison. All energies are in MeV.

	$dx = 1.25$ fm	$dx = 1.0$ fm	$dx = 0.8$ fm	HFBRAD
E_{kin}	3868.984	3866.098	3866.187	3866.163
E_{ρ^2}	− 22401.965	− 22383.542	− 22384.385	− 22384.462
E_{ρ^γ}	14548.826	14536.219	14536.840	14536.890
$E_{\rho\Delta\rho}$	315.563	315.236	315.286	315.288
$E_{\rho\tau}$	1332.134	1330.147	1330.208	1330.216
$E_{\rho\nabla J}$	− 96.592	− 96.451	− 96.446	− 96.446
E_{Coul}	796.848	796.600	796.607	796.645
E_{tot}	− 1636.202	− 1635.693	− 1635.703	− 1635.707

A. Spherical limit: Doubly magic nuclei ^{208}Pb, ^{40}Ca and semimagic nucleus ^{62}Ni

A common approach to testing the accuracy of a 3D coordinate solver is to compare the 3D results for a spherical nucleus with the results obtained by a 1D spherical code that also represents the single-particle wave functions in coordinate space. We can choose extremely fine lattice constants for the 1D solver and its results can be considered to be very accurate. Instead of a benchmark with XDIAG, we compute the double magic nucleus ^{208}Pb using our XCOCG code in a cubic box of size 32^3 fm^3 with lattice constants $dx = 1.25, 1.0, 0.8$ fm, and compare them with the results obtained with the HFBRAD code [26] code (the lattice constant $dx = 0.05$ fm). In Table II we compare various contributions to the total energy E_{tot}, computed as volume integrals from corresponding terms in NEDF for ^{208}Pb in these situations. From $dx = 1.25$ to 0.8 fm, the difference of total energy between XCOCG and HFBRAD decreases from 0.5 MeV to 4 keV. In particular, from $dx = 1.0$ to 0.8 fm, the values of the energy terms have a steady convergence to those solved by HFBRAD with the maximum difference $\leqslant$100 keV. This numerical accuracy and convergence pattern is similar to the results in Ref. [22], which uses Lagrange-mesh representation [64] in the calculation of spatial derivatives. The Lagrange-mesh method is equivalent to the DVR method using FFT on a 3D spatial lattice.

The benchmarking with XDIAG starts with the case of spherical nuclei. We solve the HFB problem for the doubly magic nucleus ^{40}Ca and semimagic nucleus ^{62}Ni to demonstrate the accuracy of our solver in the cases with and without pairing. In both cases, we use a cubic lattice of size 20^3 fm^3 with different lattice constants $\mathrm{d}x = 1.25, 1.0, 0.8333$ fm. In Table III we compare various contributions to the total energy of ^{40}Ca calculated by XCOCG and XDIAG methods respectively. The main source of differences between the results of the two solvers is due to the neglect of the azimuthal component of the spin-orbit density $\boldsymbol{J}_\phi(\boldsymbol{r})$ when we utilize the axial symmetry of the system. This lattice effect vanishes as the lattice constant decreases. In particular, when $dx \leqslant 1$ fm, the difference between the total energies is less than 10 keV. Notice also that the XCOCG energies are always lower than the XDIAG energies and we attribute this to the fact that $\boldsymbol{J}_\phi(\boldsymbol{r}) \equiv 0$ in XCOCG. The same kind of calculations are performed for the semimagic nucleus ^{62}Ni with nonzero neutron pairing; see Table IV. Compared to ^{40}Ca, the difference of total energy is larger in $dx = 1.25$ case (0.175 MeV). This is due to the much larger magnitude of the spin-orbit contribution, which brings larger error in $\boldsymbol{J}$ caused by the finite lattice effects. Similarly to ^{40}Ca, as the lattice constant becomes finer, this lattice effect vanishes and the difference of total energy drops to values below 10 keV when $dx \leqslant 1$ fm.

B. Axially deformed nucleus: ^{102}Zr

The advantage of solving the HFB equation in a coordinate space basis is that it can correctly describe the asymptotic behavior of quasiparticle wave functions of nuclei with large deformations and weak binding energies [24]. Following Refs. [28–30], we choose the neutron-rich Zr isotope ^{102}Zr, which has a large prolate deformation, as the testing ground and we choose a rectangular box of size $22.5 \times 22.5 \times 30$ fm^3 to fit its large deformation. As in Sec. IV A, we compare the XCOCG and XDIAG results with different lattice sizes $dx = 1.25$ and 0.9375 fm. These are shown in Table V. The quadrupole moment Q_{20} of the nucleus is also listed in each case, where

$$Q_{20} = \langle \hat{Q} \rangle = \int (2z^2 - x^2 - y^2)\rho(\boldsymbol{r})\, d^3\boldsymbol{r} \qquad (57)$$

and $\rho(\boldsymbol{r}) = \rho_n(\boldsymbol{r}) + \rho_p(\boldsymbol{r})$.

TABLE III. Results of spherical HFB + SLy4 calculations for ^{40}Ca using the XCOCG and XDIAG approaches with a cubic 3D lattice of size $L_x = 20$ fm with different mesh size dx. All energies are in MeV. In the column of ΔE, "<0.001" means $|\Delta E|$ is less than 1 keV and can be negligible.

	$dx = 1.25$ fm			$dx = 1.0$ fm			$dx = 0.8333$ fm		
	XDIAG	XCOCG	ΔE	XDIAG	XCOCG	ΔE	XDIAG	XCOCG	ΔE
E_{kin}	623.983	624.115	0.132	624.755	624.781	0.026	624.836	624.837	<0.001
E_{ρ^2}	− 3714.569	− 3714.702	− 0.133	− 3722.988	− 3722.977	0.011	− 3723.483	− 3723.483	<0.001
E_{ρ^γ}	2396.941	2396.967	0.026	2403.198	2403.190	− 0.008	2403.555	2403.554	<0.001
$E_{\rho\Delta\rho}$	106.284	106.368	0.084	106.957	106.957	<0.001	106.992	106.992	<0.001
$E_{\rho\tau}$	173.227	173.122	− 0.105	173.725	173.712	− 0.013	173.746	173.746	<0.001
$E_{\rho\nabla J}$	− 1.233	− 1.267	− 0.033	− 1.279	− 1.287	− 0.008	− 1.282	− 1.282	<0.001
E_{Coul}	71.483	71.484	<0.001	71.532	71.532	<0.001	71.535	71.535	<0.001
E_{tot}	− 343.885	− 343.914	− 0.029	− 344.100	− 344.093	0.007	− 344.101	− 344.101	<0.001

TABLE IV. Same as Table III, but for ^{62}Ni.

	$dx = 1.25$ fm			$dx = 1.0$ fm			$dx = 0.8333$ fm		
	XDIAG	XCOCG	ΔE	XDIAG	XCOCG	ΔE	XDIAG	XCOCG	ΔE
E_{kin}	1092.413	1092.737	0.324	1090.235	1090.293	0.058	1090.107	1090.122	0.015
E_{ρ^2}	− 6335.819	− 6336.367	− 0.547	− 6319.916	− 6320.133	− 0.218	− 6318.595	− 6318.650	− 0.055
E_{ρ^γ}	4134.013	4134.153	0.140	4122.721	4122.864	0.143	4121.738	4121.775	0.037
$E_{\rho\Delta\rho}$	150.018	150.256	0.238	150.031	150.055	0.024	149.967	149.971	0.004
$E_{\rho\tau}$	341.670	341.492	− 0.178	339.456	339.460	0.005	339.326	339.329	0.003
$E_{\rho\nabla J}$	− 52.747	− 52.819	− 0.072	− 52.371	− 52.380	− 0.010	− 52.367	− 52.365	0.001
E_{Coul}	129.928	129.917	− 0.011	129.761	129.762	<0.001	129.751	129.751	<0.001
E_{pair}	− 3.834	− 3.904	− 0.070	− 3.989	− 4.000	− 0.011	− 3.986	− 3.994	− 0.009
E_{tot}	− 544.358	− 544.534	− 0.175	− 544.073	− 544.080	− 0.007	− 544.058	− 544.061	− 0.003

The difference in the total energy is 0.3 MeV between XCOCG and XDIAG for $dx = 1.25$, which almost equals to the difference in the spin-orbit energy 0.27 MeV. This error in the spin-orbit energy contribution will affect the position of the HFB minimum, leading to a difference in the quadrupole moment Q_{20} in the ground state of 3 fm^2 and in the energy terms $E_{\rho^2}, E_{\rho^\gamma}$ of 1–2 MeV between XCOCG and XDIAG. For $dx = 0.9375$ fm, the errors are significantly smaller, $\leqslant$1 keV for the total energy and 0.16 fm^2 for the quadrupole moment.

C. Constrained HFB: saddle-point of ^{240}Pu fission

Induced fission of ^{240}Pu is a frequent benchmark for many implementations based on DFT methods [9,65–67]. In a constrained HFB calculation in Ref. [9] the nucleus is brought to a shape and an energy near the outer saddle point of the fission barrier (at zero temperature), used as the initial state of a time-dependent SLDA (TDSLDA) simulation. In this work, we use our XCOCG code to reproduce the configuration on this saddle point and compare with that obtained via the XDIAG code.

To obtain a nucleus with a given quadrupole moment $\langle\hat{Q}\rangle = Q_0$, we need to minimize the Routhian

$$E' = E_{\mathrm{total}} + c(\langle\hat{Q}\rangle - Q_0)^2, \tag{58}$$

which is equivalent to adding a Lagrange multiplier in the single-particle Hamiltonian: $h' = h + 2c(\langle\hat{Q}\rangle - Q_0)\hat{Q}$. In the self-consistent calculation, the constraint strength c will be updated in each iteration using the augmented Lagrangian method [68]. This saddle-point configuration of ^{240}Pu is prepared in a rectangular box of size $25 \times 25 \times 50$ fm^3, with lattice spacing $dx = 1.25$ fm. Following paper [9], the quadrupole moment constraint is set to $Q_0 = 16\,500$ fm^2 and an additional auxiliary external field is turned on for the formation of the neck in this configuration.

In Fig. 4 we show the density profile for the converged solution. Table VI compares the XCOCG and XDIAG results for the saddle-point configuration of ^{240}Pu. As we discussed earlier, the lattice effects in the calculation of $\boldsymbol{J}$ cause the change of the HFB minimum, which results in large differences in the energy terms E_{kin}, E_{ρ^2}, E_{ρ^γ}, and $E_{\rho\Delta\rho}$. In this case, because of the strong pairing, and extremely large deformation of the heavy nucleus, any minute change of HFB minimum will result in larger difference in various energy terms than in the cases of the nuclei we tested earlier. On the other hand, the larger amplitude of $\boldsymbol{J}$, which is reflected by the absolute value of $E_{\rho\nabla J}$, also makes the lattice effects more pronounced. In principle, these differences will reduce and eventually vanish as $dx \to 0$ as we demonstrated in the earlier sections. But the large dimension of this system and the considerable number of self-consistent iterations in the constrained HFB problem

TABLE V. Results of HFB + SLy4 calculations for ^{102}Zr using XCOCG and XDIAG in a rectangular box of size $22.5 \times 22.5 \times 30$ fm^3 with lattice constants $dx = 1.25$ and 0.9375 fm. All energies are in MeV and the quadrupole moment Q_{20} is in fm^2.

	$dx = 1.25$ fm			$dx = 0.9375$ fm		
	XDIAG	XCOCG	$\Delta E(\Delta Q)$	XDIAG	XCOCG	$\Delta E(\Delta Q)$
E_{kin}	1841.034	1841.444	0.410	1838.435	1838.438	0.003
E_{ρ^2}	− 10 381.846	− 10 383.607	− 1.760	− 10 364.579	− 10 364.463	0.115
E_{ρ^γ}	6715.556	6716.620	1.064	6702.965	6702.865	− 0.100
$E_{\rho\Delta\rho}$	209.650	210.055	0.405	208.432	208.427	− 0.005
$E_{\rho\tau}$	592.475	592.366	− 0.109	590.686	590.669	− 0.017
$E_{\rho\nabla J}$	− 63.446	− 63.716	− 0.270	− 62.429	− 62.422	0.008
E_{Coul}	230.311	230.288	− 0.023	230.359	230.358	− 0.001
E_{pair}	− 3.067	− 3.083	− 0.016	− 3.231	− 3.233	− 0.002
E_{tot}	− 859.333	− 859.632	− 0.299	− 859.361	− 859.360	0.001
Q_{20}	1077.44	1080.62	3.18	1047.81	1047.65	− 0.16

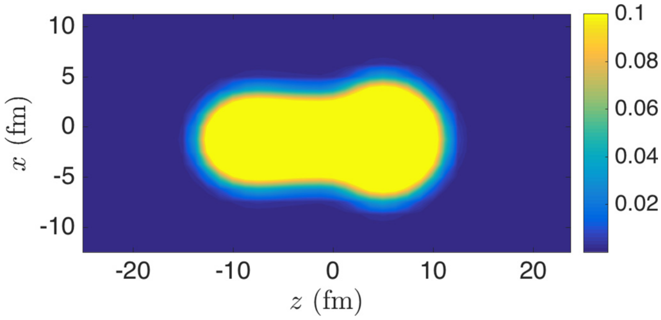


FIG. 4. Total density distribution $\rho(\boldsymbol{r}) = \rho_n(\boldsymbol{r}) + \rho_p(\boldsymbol{r})$ for the saddle-point configuration of ^{240}Pu in the $y = 0$ plane.

will make the calculation on a finer lattice grid more expensive with the current computational resources.

D. A nucleus immersed in a neutron superfluid sea

We used the shifted COCG method to generate initial states for studies of quantum vortex dynamics in a neutron star crust [11]. The stationary state is a superfluid neutron medium containing a quantum vortex and an immersed nucleus located in the vicinity of the topological defect or vortex. The calculations were performed with the FaNDF^0 nuclear density functional constructed by Fayans *et al.* [69,70], which is particularly well suited for these type of studies. The bare pairing coupling constant in Eq. (10) was chosen to reproduce the rescaled BCS $^1S_0$ pairing gap in neutron matter; for more details, see Ref. [11].

For these studies we used a simulation volume of size $75 \times 75 \times 60$ fm^3 with a lattice spacing $dx = 1.5$ fm. The energy cutoff was chosen to be 75 MeV. In the box, we place a tube (simulated by a flat-bottomed external potential) that we fill with superfluid neutrons of density $n = 0.014$ fm^{-3} or 0.031 fm^{-3}. The problem has been simplified by dropping the spin-orbit term, which is not expected to play a major role in vortex pinning. The simplification results in an HFB matrix (13) with a simpler block structure ($h_{\uparrow\downarrow} = h_{\downarrow\uparrow} = 0$) and a smaller dimension $2N_xN_yN_z$:

$$\begin{pmatrix} h_{\uparrow\uparrow} - \mu & \Delta \\ \Delta^* & -h^*_{\downarrow\downarrow} + \mu \end{pmatrix} \begin{pmatrix} u_{k\uparrow} \\ v_{k\downarrow} \end{pmatrix} = E_k \begin{pmatrix} u_{k\uparrow} \\ v_{k\downarrow} \end{pmatrix}. \tag{59}$$

After this reduction, the HFB matrix has the size $200\,000 \times 200\,000$ and it still represents a very demanding problem for the traditional approaches to determine the stationary states; see Sec. III E and Table I. We solved this problem successfully with moderate computational costs to achieve self-consistency using GPUs and the COCG approach described here on the Titan supercomputer [62]. Only after the final iteration do we use a diagonalization to generate the wave functions on the Edison supercomputer [32].

TABLE VI. Results of HFB + SLy4 calculations for the saddle point of ^{240}Pu using XCOCG and XDIAG in a rectangular box of size $25 \times 25 \times 50$ fm^3 with mesh size $dx = dy = dz = 1.25$ fm. All energies are in MeV.

	XDIAG	XCOCG	ΔE
E_{kin}	4418.132	4419.125	0.993
E_{ρ^2}	−25 104.582	−25 109.048	−4.467
E_{ρ^γ}	16 238.995	16 241.781	2.786
$E_{\rho\Delta\rho}$	408.182	409.032	0.850
$E_{\rho\tau}$	1465.062	1464.900	−0.161
$E_{\rho\nabla J}$	−111.516	−112.088	−0.572
E_{Coul}	901.083	901.062	−0.021
E_{pair}	−8.801	−8.818	−0.017
E_{tot}	−1793.439	−1794.054	−0.615

In Fig. 5 we present stationary configurations (with constraints) for two background neutron densities: $n = 0.014$ fm^{-3} and 0.031 fm^{-3}. The nuclear defect consists of $Z = 50$ protons. Two mutual configurations were considered: (1) a quantum vortex attached to the nucleus (pinned case) and (2) a nucleus outside the vortex core (unpinned case). The position of nucleus in the box was fixed by adding a constraint to the density functional for the center of mass of the protons, in a similar fashion as it was done for the quadrupole moment Eq. (58) and for the for saddle point in ^{240}Pu test case. The vortex was generated by imprinting the correct phase pattern for the pairing field Δ in the neutron channel $\Delta(\rho,z,\phi) = |\Delta(\rho,z)| \exp(i\phi)$, where $\rho = \sqrt{x^2 + y^2}$ is the distance from the center of the tube and $\phi = \tan^{-1}\frac{y}{x}$. In the pinned case, we took advantage of the axial symmetry of the problem and of the reflection symmetry with respect to one plane in the unpinned configuration.

From the energetics of these systems we determine that for both densities the configuration with the nucleus located outside the vortex core has a lower energy per particle than the pinned configuration by about 6 and 4 keV respectively for densities $n = 0.014$ fm^{-3} and 0.031 fm^{-3}. Thus one can expect that the effective vortex-nucleus interaction is repulsive in nature, which was further confirmed by studying the motion of the vortex in dynamical simulations [11].

V. FURTHER EXTENSION

A. Linear response

The free linear response of a many-fermion system can be evaluated using the same COCG approach. For simplicity we will illustrate this procedure here only for a normal system, as the extension to the superfluid case is straightforward. The free polarization operator is defined as [71,72]

$$\Pi_0(\omega,\boldsymbol{r},\boldsymbol{r}') = \sum_k \psi_k(\boldsymbol{r})\psi_k^*(\boldsymbol{r}') \times [G(\varepsilon_k + \omega,\boldsymbol{r},\boldsymbol{r}') + G(\varepsilon_k - \omega,\boldsymbol{r},\boldsymbol{r}')], \tag{60}$$

where $\psi_k(\boldsymbol{r})$ and $G(\boldsymbol{r},\boldsymbol{r}',z)$ were defined in Eqs. (2) and (4), and the summation is over occupied levels. In practice, the polarization operator is evaluated for a complex energy $\omega + i\gamma$, since the spreading width $\Gamma^\downarrow$ is either not accounted for in the random phase approximation, or in order to imitate to some extent the imaginary part of the optical potential.

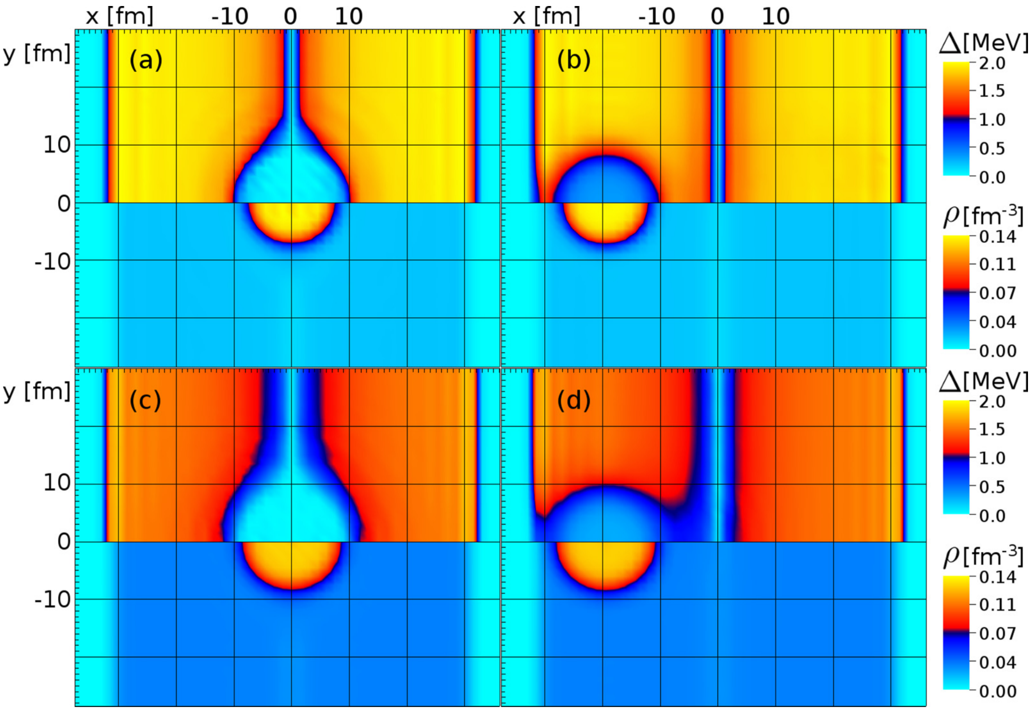


FIG. 5. The lowest energy states generated by shifted COCG method for neutron background density $n = 0.014$ fm^{-3} [panels (a) and (b)] and 0.031 fm^{-3} [panels (c) and (d)]. In each box of size 75 × 60 fm^2 we show the absolute value of the neutron pairing potential $\Delta(\boldsymbol{r})$ (upper half) and the total density distribution $\rho(\boldsymbol{r})$ (lower half). The black lines separating blue and red regions correspond to a value 1 MeV for the paring and of 0.07 fm^{-3} for the density respectively.

The spreading width accounts for the fragmentation of the particle-hole transition strength due to the coupling to more complex states. If the complex integration contour in Eq. (6) is chosen so that $|\mathrm{Im}\, z| < \gamma$ one can easily show that

$$\Pi_0(\omega+i\gamma,\boldsymbol{r},\boldsymbol{r}') = \frac{1}{2\pi i}\oint_C G(z,\boldsymbol{r},\boldsymbol{r}')[G(z+\omega+i\gamma,\boldsymbol{r},\boldsymbol{r}') + G(z-\omega-i\gamma,\boldsymbol{r},\boldsymbol{r}')]. \quad (61)$$

It remains to be established if this method for the evaluation of the free polarization operator $\Pi_0(\omega+i\gamma,\boldsymbol{r},\boldsymbol{r}')$ and the subsequent determination of the full response is competitive and under what conditions with the finite-amplitude method [73–78] and/or the time-dependent approach [44,45] for deform open-shell nuclei.

B. Extraction of eigenvalues and eigenvectors using the shifted COCG Krylov method

The shifted COCG Krylov method can be used also to determine the eigenvalues and the eigenvectors using the approach described in Refs. [39–41]. In this approach one has at first to evaluate the moments

$$\mu_k = \frac{1}{2\pi i}\oint_\Gamma dz \sum_n \frac{(z-\epsilon)^k}{z-\varepsilon_n}, \quad k = 0,1,\dots,2N-1, \quad (62)$$

where ϵ is located inside the contour Γ enclosing a segment on the real axis with a known number N of the eigenvalues and where ε_n are eigenvalues. Once these moments are computed the eigenvalues are obtained by solving a generalized eigenvalue problem for two matrices of size $N \times N$. The number of eigenvalues in a given interval is not known *a priori* and some eigenvalues could also be degenerate. In the presence of degeneracies one has to disentangle the corresponding eigenvectors. If the degeneracy is due to spherical or axial symmetry, one can introduce slightly different lattice constants dx, dy, and dz or a very weak external field and lift the degeneracies at a level that has no noticeable consequence on the physics studied. Subsequently, the solution of the Schrödinger equation for the corresponding eigenvector ψ_n in the case of a known nondegenerate eigenvalue $(H-\varepsilon_n)\psi_n = 0$ is a trivial linear algebra problem. In the case of Kramers degeneracies, one can easily separate the two degenerate eigenvectors. The unknown number N of eigenvalues in a given energy interval can be determined by evaluating the trace (integral over all coordinates and summation over all four components) of the Green's function of the Hamiltonian (13)

$$N = \mathrm{Tr}\left(\frac{1}{2\pi i}\oint_\Gamma dz \frac{1}{z-H}\right). \quad (63)$$

Thus the need to use diagonalization of very large matrices can be completely eschewed.

VI. CONCLUSIONS

In this paper, we describe a new approach for solving the HFB type of equations in a coordinate representation that is different from the traditional approaches based typically on direct diagonalizations. In the present approach, there is no need to evaluate the individual single-particle wave functions and their energies, but instead we calculate the Green's function of the HFB equations, from which we extract various densities after evaluating a contour integral. The Green's functions are obtained by solving a set of linear equations with scalar shifts using the iterative shifted COCG Krylov method. We demonstrate the high accuracy of the iterative shifted COCG approach by solving typical nuclear problems with and without complicated constraints, such as the fission outer saddle point of ^{240}Pu and a quantum vortex state in a neutron star crust. A notable advantage of this algorithm is its suitability for efficient parallelization and effective utilization of heterogeneous computing platforms. The method becomes computationally superior for large spatial lattice sizes that are otherwise computationally very expensive for standard approaches, such as a direct diagonalization.

ACKNOWLEDGMENTS

We thank G. F. Bertsch for discussions and I. Stetcu and P. Magierski for many helpful discussions and code sharing. This work was supported in part by U.S. DOE Office of Science Grant No. DE-FG02-97ER41014. This work was also supported in part by the Polish National Science Center (NCN) under Contract No. UMO-2014/13/D/ST3/01940. Calculations have been performed at the OLCF Titan, resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725; at NERSC Edison, resources of the National Energy Research Scientific computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231; and at Moonlight, resources of the Institutional Computing Program at Los Alamos National Laboratory.

APPENDIX: EQUATION OF GREEN'S FUNCTION FOR HFB EQUATION

For the HFB Hamiltonian H in Eq. (13), the single-particle Hamiltonian in Eq. (14) reads

$$h = -\boldsymbol{\nabla}\cdot\frac{\hbar^2}{2m^*(\boldsymbol{r})}\boldsymbol{\nabla} + U(\boldsymbol{r}) - i\boldsymbol{W}(\boldsymbol{r})\cdot(\boldsymbol{\nabla}\times\boldsymbol{\sigma}), \tag{A1}$$

where the kinetic energy term is represented by a real symmetric operator in numerical implementations T. When discretized, besides the local term $U(\boldsymbol{r})_{mn} = U_n\delta_{nm}$, the nonlocal term T and spin-orbit terms require more attention. For T, since

$$-\boldsymbol{\nabla}\cdot\frac{\hbar^2}{2m^*(\boldsymbol{r})}\boldsymbol{\nabla}v(\boldsymbol{r}) = -\frac{1}{2}\left[\frac{\hbar^2}{2m^*(\boldsymbol{r})}\nabla^2 v(\boldsymbol{r}) + \nabla^2\left(\frac{\hbar^2}{2m^*(\boldsymbol{r})}v(\boldsymbol{r})\right) - \left(\nabla^2\frac{\hbar^2}{2m^*(\boldsymbol{r})}\right)v(\boldsymbol{r})\right], \tag{A2}$$

we have

$$T_{nm} = \left(-\boldsymbol{\nabla}\cdot\frac{\hbar^2}{2m^*(\boldsymbol{r})}\boldsymbol{\nabla}\right)_{nm} = -\frac{1}{2}(\nabla^2)_{nm}\left(\frac{\hbar^2}{2m^*_n} + \frac{\hbar^2}{2m^*_m}\right) + \frac{1}{2}\left(\nabla^2\frac{\hbar^2}{2m^*}\right)_n\delta_{nm}, \tag{A3}$$

where the Laplacian operator ∇^2 is a symmetric matrix in the discrete variable representation (DVR) basis [21] and $m^*_{n/m} = m^*(\boldsymbol{r}_{n/m})$. A similar symmetrization is also performed for the spin-orbit term

$$\boldsymbol{W}(\boldsymbol{r})\cdot(\boldsymbol{\nabla}\times\boldsymbol{\sigma}) = \tfrac{1}{2}\{\boldsymbol{W}(\boldsymbol{r})\cdot(\boldsymbol{\nabla}\times\boldsymbol{\sigma}) + \boldsymbol{\nabla}\cdot[\boldsymbol{\sigma}\times\boldsymbol{W}(\boldsymbol{r})]\}, \tag{A4}$$

where the gradient operator $\boldsymbol{\nabla} = \{\partial_x,\partial_y,\partial_z\}$ is an antisymmetric operator in each spatial direction in the DVR representation. In the p-p channel, the complex pairing field $\Delta(\boldsymbol{r})$ is diagonal when discretized, $\Delta(\boldsymbol{r})_{mn} = \Delta_n\delta_{nm}$. Finally, after separating the real and imaginary parts of the HFB Hamiltonian $H = A + iB$, the equation of Green's function $G = G_x + iG_y$ for a HFB equation is

$$(zI - H')\begin{pmatrix} G_x \\ G_y \end{pmatrix} = \begin{pmatrix} \delta^{(4)} \\ 0 \end{pmatrix}, \quad H' = \begin{pmatrix} A & -B \\ B & A \end{pmatrix}, \tag{A5}$$

and where

$$A = \begin{pmatrix} T + \tilde{U} & \widetilde{W_x\partial_z} - \widetilde{W_z\partial_x} & 0 & \mathrm{Re}\Delta \\ \widetilde{W_z\partial_x} - \widetilde{W_x\partial_z} & T + \tilde{U} & -\mathrm{Re}\Delta & 0 \\ 0 & -\mathrm{Re}\Delta & -T - \tilde{U} & \widetilde{W_z\partial_x} - \widetilde{W_x\partial_z} \\ \mathrm{Re}\Delta & 0 & \widetilde{W_x\partial_z} - \widetilde{W_z\partial_x} & -T - \widetilde{U} \end{pmatrix}, \tag{A6}$$

$$B = \begin{pmatrix} \widetilde{W_y\partial_x} - \widetilde{W_x\partial_y} & \widetilde{W_z\partial_y} - \widetilde{W_y\partial_z} & 0 & \mathrm{Im}\Delta \\ \widetilde{W_z\partial_y} - \widetilde{W_y\partial_z} & \widetilde{W_x\partial_y} - \widetilde{W_y\partial_x} & -\mathrm{Im}\Delta & 0 \\ 0 & \mathrm{Im}\Delta & \widetilde{W_y\partial_x} - \widetilde{W_x\partial_y} & \widetilde{W_z\partial_y} - \widetilde{W_y\partial_z} \\ -\mathrm{Im}\Delta & 0 & \widetilde{W_z\partial_y} - \widetilde{W_y\partial_z} & \widetilde{W_x\partial_y} - \widetilde{W_y\partial_x} \end{pmatrix}, \tag{A7}$$

with $\tilde{U} = U - \mu$ and $\widetilde{W_i \partial_j}$ the anticommutator of W_i and ∂_j:

$$\widetilde{W_i \partial_j} = \tfrac{1}{2}(W_i \partial_j + \partial_j W_i), \quad i,j = x,y,z,i \neq j. \tag{A8}$$

One can show that $A = A^T$ and $B = -B^T$. In the shifted COCG method, we do not need to construct these $4N_{xyz} \times 4N_{xyz}$ matrices explicitly because we only need the MV product between these matrices and the vectors $\boldsymbol{x}_n$, $\boldsymbol{p}_n$ (see Sec. III A). Among the MV product operation, the product between the local part of H and the vectors can be regarded as a vector-vector product. The product between the vectors and the nonlocal part of H, due to the Laplacian and gradient operators, can be performed using the fast Fourier transform.

[1] M. Bender, P.-H. Heenen, and P.-G. Reinhard, Self-consistent mean-field models for nuclear structure, Rev. Mod. Phys. **75**, 121 (2003).

[2] E. Runge and E. K. U. Gross, Density-Functional Theory for Time-Dependent Systems, Phys. Rev. Lett. **52**, 997 (1984).

[3] M. A. L. Marques, C. A. Ullrich, F. Nogueira, A. Rubio, K. Burke, and E. U. K. Gross (eds.), *Time-Dependent Density Functional Theory*, Vol. 706 (Springer, Berlin, 2006).

[4] A. L. M. Marques, T. N. Maitra, M. S. F. Nogueira, E. K. U. Gross, and A. Rubio (eds.), *Fundamentals of Time-Dependent Density Functional Theory*, Vol. 837 (Springer, Berlin, 2012).

[5] P. Ring and P. Schuck, *The Nuclear Many-Body Problem* (Springer, Berlin, 2004).

[6] G. Bertsch, The nuclear density of states in the space of nuclear shapes, Phys. Lett. B **95**, 157 (1980).

[7] F. Barranco, G. F. Bertsch, R. A. Broglia, and E. Vigezzi, Large-amplitude motion in superfluid Fermi droplets, Nucl. Phys. A **512**, 253 (1990).

[8] G. F. Bertsch and A. Bulgac, Comment on "Spontaneous Fission: A Kinetic Approach", Phys. Rev. Lett. **79**, 3539 (1997).

[9] A. Bulgac, P. Magierski, K. J. Roche, and I. Stetcu, Induced fission of ^{240}Pu within a real-time microscopic framework, Phys. Rev. Lett. **116**, 122504 (2016).

[10] P. W. Anderson and N. Itoh, Pulsar glitches and restlessness as a hard superfluid phenomenon, Nature (London) **256**, 25 (1975).

[11] G. Wlazłowski, K. Sekizawa, P. Magierski, A. Bulgac, and M. McNeil Forbes, Vortex Pinning and Dynamics in the Neutron Star Crust, Phys. Rev. Lett. **117**, 232701 (2016).

[12] K. T. R. Davies, H. Flocard, S. Krieger, and M. S. Weiss, Application of the imaginary time step method to the solution of the static Hartree-Fock problem, Nucl. Phys. A **342**, 111 (1980).

[13] W. Ryssens, V. Hellemans, M. Bender, and P.-H. Heenen, Solution of the Skyrme–HF+BCS equation on a 3D mesh, II: A new version of the Ev8 code, Comput. Phys. Commun. **187**, 175 (2015).

[14] J. L. Egido, J. Lessing, V. Martin, and L. M. Robledo, On the solution of the Hartree-Fock-Bogoliubov equations by the conjugate gradient method, Nucl. Phys. A **594**, 70 (1995).

[15] L. M. Robledo and G. F. Bertsch, Application of the gradient method to Hartree-Fock-Bogoliubov theory, Phys. Rev. C **84**, 014312 (2011).

[16] G. H. Golub and H. A. van der Vorst, Eigenvalue computation in the 20th century, J. Comput. Appl. Math. **123**, 35 (2000).

[17] M. V. Stoitsov, N. Schunck, M. Kortelainen, N. Michel, H. Nam, E. Olsen, J. Sarich, and S. Wild, Axially deformed solution of the Skyrme-Hartree–Fock–Bogoliubov equations using the transformed harmonic oscillator basis (II) HFBTHO v2.00d: A new version of the program, Comput. Phys. Commun. **184**, 1592 (2013).

[18] N. Schunck, J. Dobaczewski, J. McDonnell, W. Satuła, J. A. Sheikh, A. Staszczak, M. Stoitsov, and P. Toivanen, Solution of the Skyrme–Hartree–Fock–bogolyubov equations in the cartesian deformed harmonic-oscillator basis: (VII) HFODD (v2.49t): A new version of the program, Comput. Phys. Commun. **183**, 166 (2012).

[19] J. C. Pei, G. I. Fann, R. J. Harrison, W. Nazarewicz, Y. Shi, and S. Thornton, Adaptive multi-resolution 3d Hartree-Fock-Bogoliubov solver for nuclear structure, Phys. Rev. C **90**, 024317 (2014).

[20] N. Michel, W. Nazarewicz, M. Płoszajczak, and K. Bennaceur, Gamow Shell Model Description of Neutron-Rich Nuclei, Phys. Rev. Lett. **89**, 042502 (2002).

[21] A. Bulgac and M. McNeil Forbes, Use of the discrete variable representation basis in nuclear physics, Phys. Rev. C **87**, 051301 (2013).

[22] W. Ryssens, P.-H. Heenen, and M. Bender, Numerical accuracy of mean-field calculations in coordinate space, Phys. Rev. C **92**, 064318 (2015).

[23] J. Dobaczewski, H. Flocard, and J. Treiner, Hartree-Fock-Bogolyubov description of nuclei near the neutron-drip line, Nucl. Phys. A **422**, 103 (1984).

[24] J. Dobaczewski, W. Nazarewicz, T. R. Werner, J. F. Berger, C. R. Chinn, and J. Dechargé, Mean-field description of ground-state properties of drip-line nuclei: Pairing and continuum effects, Phys. Rev. C **53**, 2809 (1996).

[25] Y. N. Zhang, J. C. Pei, and F. R. Xu, Hartree-Fock-Bogoliubov descriptions of deformed weakly bound nuclei in large coordinate spaces, Phys. Rev. C **88**, 054305 (2013).

[26] K. Bennaceur and J. Dobaczewski, Coordinate-space solution of the Skyrme–Hartree–Fock–Bogolyubov equations within spherical symmetry: The program HFBRAD (v1.00), Comput. Phys. Commun. **168**, 96 (2005).

[27] V. E. Oberacker, A. S. Umar, E. Terán, and A. Blazkiewicz, Hartree-Fock-Bogoliubov calculations in coordinate space: Neutron-rich sulfur, zirconium, cerium, and samarium isotopes, Phys. Rev. C **68**, 064302 (2003).

[28] E. Terán, V. E. Oberacker, and A. S. Umar, Axially symmetric Hartree-Fock-Bogoliubov calculations for nuclei near the drip lines, Phys. Rev. C **67**, 064314 (2003).

[29] A. Blazkiewicz, V. E. Oberacker, A. S. Umar, and M. Stoitsov, Coordinate space Hartree-Fock-Bogoliubov calculations for the zirconium isotope chain up to the two-neutron drip line, Phys. Rev. C **71**, 054321 (2005).

[30] J. C. Pei, M. V. Stoitsov, G. I. Fann, W. Nazarewicz, N. Schunck, and F. R. Xu, Deformed coordinate-space Hartree-

Fock-Bogoliubov approach to weakly bound nuclei and large deformations, Phys. Rev. C **78**, 064306 (2008).

[31] http://www.netlib.org/scalapack/.

[32] http://www.nersc.gov/users/computational-systems/edison/.

[33] http://www.nvidia.com/object/cuda_home_new.html.

[34] H. A. van der Vorst and J. B. M. Melissen, A Petrov-Galerkin type method for solving $Axk = b$, where A is symmetric complex, IEEE Trans. Magn. **26**, 706 (1990).

[35] S. Yamamoto, T. Sogabe, T. Hoshi, S.-L. Zhang, and T. Fujiwara, Shifted conjugate-orthogonal–conjugate-gradient method and its application to double orbital extended Hubbard model, J. Phys. Soc. Jpn. **77**, 114713 (2008).

[36] R. Takayama, T. Hoshi, T. Sogabe, S.-L. Zhang, and T. Fujiwara, Linear algebraic calculation of the Green's function for large-scale electronic structure theory, Phys. Rev. B **73**, 165108 (2006).

[37] T. Mizusaki, K. Kaneko, M. Honma, and T. Sakurai, Filter diagonalization of shell-model calculations, Phys. Rev. C **82**, 024310 (2010).

[38] N. Shimizu, Y. Utsuno, Y. Futamura, T. Sakurai, T. Mizusaki, and T. Otsuka, Stochastic estimation of nuclear level density in the nuclear shell model: An application to parity-dependent level density in 58Ni, Phys. Lett. B **753**, 13 (2016).

[39] T. Sakurai and H. Sugiura, A projection method for generalized eigenvalue problems using numerical integration, J. Comput. Appl. Math. **159**, 119 (2003).

[40] T. Sakurai, Y. Kodaki, H. Tadano, D. Takahashi, M. Sato, and U. Nagashima, A parallel method for large sparse generalized eigenvalue problems using a gridrpc system, Future Gen. Comp. Sys. **24**, 613 (2008).

[41] T. Ikegami, T. Sakurai, and U. Nagashima, A filter diagonalization for generalized eigenvalue problems based on the Sakurai–Sugiura projection method, J. Comput. Appl. Math. **233**, 1927 (2010).

[42] J. Dobaczewski, W. Nazarewicz, and M. V. Stoitsov, Nuclear ground-state properties from mean-field calculations, Eur. Phys. J. A **15**, 21 (2002).

[43] E. Chabanat, P. Bonche, P. Haensel, J. Meyer, and R. Schaeffer, A Skyrme parametrization from subnuclear to neutron star densities, part II: Nuclei far from stabilities, Nucl. Phys. A **635**, 231 (1998).

[44] I. Stetcu, A. Bulgac, P. Magierski, and K. J. Roche, Isovector giant dipole resonance from the 3D time-dependent density functional theory for superfluid nuclei, Phys. Rev. C **84**, 051309 (2011).

[45] I. Stetcu, C. A. Bertulani, A. Bulgac, P. Magierski, and K. J. Roche, Relativistic Coulomb Excitation Within the Time Dependent Superfluid Local Density Approximation, Phys. Rev. Lett. **114**, 012701 (2015).

[46] A. Bulgac and Y. Yu, Renormalization of the Hartree-Fock-Bogoliubov Equations in the Case of a Zero Range Pairing Interaction, Phys. Rev. Lett. **88**, 042504 (2002).

[47] Y. Yu and A. Bulgac, Energy Density Functional Approach to Superfluid Nuclei, Phys. Rev. Lett. **90**, 222501 (2003).

[48] P. J. Borycki, J. Dobaczewski, W. Nazarewicz, and M. V. Stoitsov, Pairing renormalization and regularization within the local density approximation, Phys. Rev. C **73**, 044319 (2006).

[49] G. F. Bertsch, C. A. Bertulani, W. Nazarewicz, N. Schunck, and M. V. Stoitsov, Odd-even mass differences from self-consistent mean field theory, Phys. Rev. C **79**, 034306 (2009).

[50] A. Bulgac, M. McNeil Forbes, and S. Jin, Nuclear energy density functionals: What do we really know? arXiv:1506.09195.

[51] M. Dutra, O. Lourenço, J. S. Sá Martins, A. Delfino, J. R. Stone, and P. D. Stevenson, Skyrme interaction and nuclear matter constraints, Phys. Rev. C **85**, 035201 (2012).

[52] A. Bohr and B. R. Mottelson, *Nuclear Structure* (World Scientific, Singapore, 1998), Vol. 1.

[53] L. N. Trefethen and D. Bau, *Numerical Linear Algebra* (Society for Industrial and Applied Mathematics, Philadelphia, PA, 1997).

[54] A. Bulgac, Hartree-Fock-Bogoliubov approximation for finite systems, Report No. FT-194-1980, CIP, Bucharest; arXiv:nucl-th/9907088.

[55] X. M. Gu, T. Z. Huang, L. Li, H. B. Li, T. Sogabe, and M. Clemens, Quasi-minimal residual variants of the COCG and COCR methods for complex symmetric linear systems in electromagnetic simulations, IEEE Trans. Microwave Theory Tech. **62**, 2859 (2014).

[56] Y. H. Li, Z. P. Nie, M. Meng, X. Q. Zhang, and X. Y. Sun, An efficient MAINV preconditioned COCG method for FEM analysis of millimeter wave filters, J. Infrared, Millimeter, Terahertz Waves **32**, 216 (2011).

[57] L. Li, T.-Z. Huang, and Z.-G. Ren, A preconditioned COCG method for solving complex symmetric linear systems arising from scattering problems, J. Electromagn. Waves. Appl. **22**, 2023 (2008).

[58] Y. Wang, K. Du, and W. Sun, Preconditioning iterative algorithm for the electromagnetic scattering from a large cavity, Num. Linear Algebra Appl. **16**, 345 (2009).

[59] M. Aichinger and E. Krotscheck, A fast configuration space method for solving local Kohn–Sham equations, Comput. Mater. Sci. **34**, 188 (2005).

[60] A. Frommer, Bicgstab(ℓ) for families of shifted linear systems, Computing **70**, 87 (2003).

[61] A. Frommer and U. Glässner, Restarted GMRES for shifted linear systems, SIAM J. Sci. Comput. **19**, 15 (1998).

[62] https://www.olcf.ornl.gov/computing-resources/titan-cray-xk7/.

[63] A. Baran, A. Bulgac, M. McNeil Forbes, G. Hagen, W. Nazarewicz, N. Schunck, and M. V. Stoitsov, Broyden's method in nuclear structure calculations, Phys. Rev. C **78**, 014318 (2008).

[64] D. Baye, The Lagrange-mesh method, Phys. Rep. **565**, 1 (2015).

[65] W. Younes and D. Gogny, Microscopic calculation of ^{240}Pu scission with a finite-range effective force, Phys. Rev. C **80**, 054313 (2009).

[66] N. Schunck, D. Duke, H. Carr, and A. Knoll, Description of induced nuclear fission with Skyrme energy functionals: Static potential energy surfaces and fission fragment properties, Phys. Rev. C **90**, 054305 (2014).

[67] D. Regnier, N. Dubray, N. Schunck, and M. Verrière, Fission fragment charge and mass distributions in ^{239}Pu(n,f) in the adiabatic nuclear energy density functional theory, Phys. Rev. C **93**, 054611 (2016).

[68] A. Staszczak, M. Stoitsov, A. Baran, and W. Nazarewicz, Augmented Lagrangian method for constrained nuclear density functional theory, Eur. Phys. J. A **46**, 85 (2010).

[69] S. A. Fayans, Towards a universal nuclear density functional, J. Exp. Theor. Phys. Lett. **68**, 169 (1998).

[70] S. A. Fayans, S. V. Tolokonnikov, E. L. Trykov, and D. Zawischa, Nuclear isotope shifts within the local energy-density functional approach, Nucl. Phys. A **676**, 49 (2000).

[71] S. Shlomo and G. Bertsch, Nuclear response in the continuum, Nucl. Phys. A **243**, 507 (1975).

[72] G. F. Bertsch and S. F. Tsai, A study of the nuclear response function, Phys. Rep. **18**, 125 (1975).

[73] T. Nakatsukasa, T. Inakura, and K. Yabana, Finite amplitude method for the solution of the random-phase approximation, Phys. Rev. C **76**, 024318 (2007).

[74] P. Avogadro and T. Nakatsukasa, Finite amplitude method for the quasiparticle random-phase approximation, Phys. Rev. C **84**, 014314 (2011).

[75] T. Nakatsukasa, Finite amplitude method in linear response TDDFT calculations, J. Phys.: Conf. Ser. **533**, 012054 (2014).

[76] N. Hinohara, M. Kortelainen, and W. Nazarewicz, Low-energy collective modes of deformed superfluid nuclei within the finite-amplitude method, Phys. Rev. C **87**, 064309 (2013).

[77] T. Oishi, M. Kortelainen, and N. Hinohara, Finite amplitude method applied to the giant dipole resonance in heavy rare-earth nuclei, Phys. Rev. C **93**, 034329 (2016).

[78] M. T. Mustonen, T. Shafer, Z. Zenginerler, and J. Engel, Finite-amplitude method for charge-changing transitions in axially deformed nuclei, Phys. Rev. C **90**, 024308 (2014).