



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[☆]



M.V. Stoitsov^{a,b}, N. Schunck^{c,*}, M. Kortelainen^{a,b,d}, N. Michel^a, H. Nam^b, E. Olsen^a, J. Sarich^e, S. Wild^e

^a Department of Physics and Astronomy, University of Tennessee, Knoxville, TN 37996, USA

^b Oak Ridge National Laboratory, P.O. Box 2008, Oak Ridge, TN 37831, USA

^c Physics Division, Lawrence Livermore National Laboratory Livermore, CA 94551, USA

^d Department of Physics, P.O. Box 35 (YFL), FI-40014 University of Jyväskylä, Finland

^e Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439, USA

ARTICLE INFO

Article history:

Received 5 October 2012

Received in revised form

21 December 2012

Accepted 20 January 2013

Available online 29 January 2013

Keywords:

Nuclear density functional theory

Self-consistent mean-field

Hartree–Fock–Bogoliubov

Skyrme functionals

Finite temperature HFB

Particle number projection

Transformed harmonic oscillator

Axial symmetry

ABSTRACT

We describe the new version 2.00d of the code HFBTHO that solves the nuclear Skyrme–Hartree–Fock (HF) or Skyrme–Hartree–Fock–Bogoliubov (HFB) problem by using the cylindrical transformed deformed harmonic oscillator basis. In the new version, we have implemented the following features: (i) the modified Broyden method for non-linear problems, (ii) optional breaking of reflection symmetry, (iii) calculation of axial multipole moments, (iv) finite temperature formalism for the HFB method, (v) linear constraint method based on the approximation of the Random Phase Approximation (RPA) matrix for multi-constraint calculations, (vi) blocking of quasi-particles in the Equal Filling Approximation (EFA), (vii) framework for generalized energy density with arbitrary density-dependences, and (viii) shared memory parallelism via OpenMP pragmas.

Program summary

Program title: HFBTHO v2.00d

Catalog identifier: ADUI_v2_0

Program summary URL: http://cpc.cs.qub.ac.uk/summaries/ADUI_v2_0.html

Program obtainable from: CPC Program Library, Queen's University, Belfast, N. Ireland

Licensing provisions: GNU General Public License version 3

No. of lines in distributed program, including test data, etc.: 167228

No. of bytes in distributed program, including test data, etc.: 2672156

Distribution format: tar.gz

Programming language: FORTRAN-95.

Computer: Intel Pentium-III, Intel Xeon, AMD-Athlon, AMD-Opteron, Cray XT5, Cray XE6.

Operating system: UNIX, LINUX, WindowsXP.

RAM: 200 Mwords

Word size: 8 bits

Classification: 17.22.

Does the new version supercede the previous version?: Yes

Catalog identifier of previous version: ADUI_v1_0

Journal reference of previous version: Comput. Phys. Comm. 167 (2005) 43

Nature of problem:

The solution of self-consistent mean-field equations for weakly-bound paired nuclei requires a correct description of the asymptotic properties of nuclear quasi-particle wave functions. In the present implementation, this is achieved by using the single-particle wave functions of the transformed harmonic oscillator, which allows for an accurate description of deformation effects and pairing correlations in nuclei arbitrarily close to the particle drip lines.

[☆] This paper and its associated computer program are available via the Computer Physics Communication homepage on ScienceDirect (<http://www.sciencedirect.com/science/journal/00104655>).

* Corresponding author.

E-mail addresses: schunck1@llnl.gov, schuncknf@gmail.com (N. Schunck).

Solution method:

The program uses the axial Transformed Harmonic Oscillator (THO) single- particle basis to expand quasi-particle wave functions. It iteratively diagonalizes the Hartree–Fock–Bogoliubov Hamiltonian based on generalized Skyrme-like energy densities and zero-range pairing interactions until a self-consistent solution is found. A previous version of the program was presented in: M.V. Stoitsov, J. Dobaczewski, W. Nazarewicz, P. Ring, Comput. Phys. Commun. 167 (2005) 43–63.

Reasons for new version:

Version 2.00d of HFBTHO provides a number of new options such as the optional breaking of reflection symmetry, the calculation of axial multipole moments, the finite temperature formalism for the HFB method, optimized multi-constraint calculations, the treatment of odd–even and odd–odd nuclei in the blocking approximation, and the framework for generalized energy density with arbitrary density-dependences. It is also the first version of HFBTHO to contain threading capabilities.

Summary of revisions:

1. The modified Broyden method has been implemented,
2. Optional breaking of reflection symmetry has been implemented,
3. The calculation of all axial multipole moments up to $\lambda = 8$ has been implemented,
4. The finite temperature formalism for the HFB method has been implemented,
5. The linear constraint method based on the approximation of the Random Phase Approximation (RPA) matrix for multi-constraint calculations has been implemented,
6. The blocking of quasi-particles in the Equal Filling Approximation (EFA) has been implemented,
7. The framework for generalized energy density functionals with arbitrary density-dependence has been implemented,
8. Shared memory parallelism via OpenMP pragmas has been implemented.

Restrictions:

Axial- and time-reversal symmetries are assumed.

Unusual features:

The user must have access to

- (i) the LAPACK subroutines DSYEVD, DSYTRF and DSYTRI, and their dependences, which compute eigenvalues and eigenfunctions of real symmetric matrices,
- (ii) the LAPACK subroutines DGETRI and DGETRF, which invert arbitrary real matrices, and
- (iii) the BLAS routines DCOPY, DSCAL, DGEMM and DGEMV for double-precision linear algebra (or provide another set of subroutines that can perform such tasks). The BLAS and LAPACK subroutines can be obtained from the Netlib Repository at the University of Tennessee, Knoxville: <http://netlib2.cs.utk.edu/>.

Running time:

Highly variable, as it depends on the nucleus, size of the basis, requested accuracy, requested configuration, compiler and libraries, and hardware architecture. An order of magnitude would be a few seconds for ground-state configurations in small bases $N_{\max} \approx 8 - 12$, to a few minutes in very deformed configuration of a heavy nucleus with a large basis $N_{\max} > 20$.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

The method to solve the Skyrme–Hartree–Fock–Bogoliubov equations in the transformed harmonic oscillator basis was presented in [1]. The present paper is a long write-up of the new version of the code HFBTHO. This extended version contains a number of new capabilities such as the breaking of reflection symmetry, the calculation of axial multipole moments, multi-constraint calculations and the readjustment of the corresponding Lagrange parameters using the cranking approximation of the RPA matrix, the blocking prescription in odd–even and odd–odd nuclei, the finite-temperature formalism, and generalized Skyrme-like energy functionals.

In addition to releasing a new version of the solver for general applications in nuclear science, the goal of this paper is to establish a number of precise benchmarks for nuclear structure calculations with Skyrme functionals. To this end, we devote an entire section to comparing various calculations performed with the spherical HOSPHE version 2.00 [2,3], axially-deformed HFBTHO v2.00d, and symmetry-unrestricted HFODD version 2.56 [4–6] nuclear density functional theory (DFT) solvers. Also, in order to facilitate the development of future versions of HFBTHO as well as to enable deeper

integration with the next releases of HFODD, backward compatibility of input and output files has been broken between version 1.66 of [1] and the current version 2.00d. Unless indicated otherwise, details about the methods presented in [1] still apply.

In Section 2, we review the new capabilities of the code. In Section 3, we present a number of numerical benchmarks between HFBTHO and the aforementioned DFT solvers. Such benchmarks are very important in view of the future development of these programs.

2. Modifications introduced in version 2.00d

We present in this section the major new features added to the code between version 1.66 and 2.00d. Minor improvements and bug fixes are not discussed here, the full history of changes can be found in the source code.

2.1. Modified Broyden method

In HFBTHO v2.00d, the matrix elements of the HFB matrix are updated at each iteration using the modified Broyden method,

instead of the traditional linear mixing of version 1.66. Details of the implementation, results of convergence tests, and comparisons with alternative implementations can be found in [7].

2.2. Axial multipole moments

In HFBTHO v2.00d, the expectation value of axial multipole moments $\hat{Q}_l \equiv \hat{Q}_{l0} = r^l Y_{l0}(\theta, \varphi)$ on the HFB ground-state is computed for all moments up to $l_{\max} = 8$. We recall that in spherical coordinates, the multipole moment \hat{Q}_l of order l reads

$$\hat{Q}_l(r, \theta, \varphi) = r^l \sqrt{\frac{2l+1}{4\pi}} P_l(\cos \theta), \quad (1)$$

where P_l is the Legendre polynomial of order l [8]. Spherical and cylindrical coordinate systems are related through $r^2 = \rho^2 + z^2$ and $r \cos \theta = z$. Recurrence relations on Legendre polynomials give an analytical expression for $\hat{Q}_l(r, z, \varphi)$ for $l = 0, \dots, 8$ [8]. Multipole moments can also be used as constraints. In this case, the matrix elements of \hat{Q}_l in the HO basis need to be computed. They are evaluated numerically on the Gauss-Laguerre and Gauss-Hermite nodes of integration used throughout the code [1].

2.3. Finite-temperature HFB method

The code HFBTHO v2.00d solves the finite temperature HFB (FT-HFB) equations. The numerical implementation is similar to that of HFODD v2.49t in [5]. Let us recall that the FT-HFB equations take the same form as the HFB equations at $T = 0$, only the one-body density matrix and pairing tensor now depend on the Fermi-Dirac occupation f_μ of quasi-particle states μ . Assuming axial- and time-reversal symmetry, all density matrices are real and read

$$\begin{aligned} \rho &= UfU^T + V(1-f)V^T \\ \kappa &= UfV^T + V(1-f)U^T, \end{aligned} \quad (2)$$

with U, V the matrices of the Bogoliubov transformation. In HFBTHO, these matrices are block-diagonal. As in HFODD, the Fermi level λ is not treated explicitly as the Lagrange parameter for the multipole operator \hat{Q}_{l0} alongside other multipole moments \hat{Q}_{lm} . Instead, it is determined directly at each iteration from the conservation of particle number and is based on the BCS formula

$$N(\lambda) = \sum_{\mu} [v_{\mu}(\lambda)^2 + (u_{\mu}(\lambda) - v_{\mu}(\lambda))f_{\mu}(\lambda)]. \quad (3)$$

The BCS occupations are given by the traditional formulas

$$v_{\mu}^2 = \frac{1}{2} \left[1 - \frac{\varepsilon_{\mu} - \lambda}{E_{\mu}^{\text{BCS}}} \right], \quad u_{\mu}^2 = 1 - v_{\mu}^2, \quad (4)$$

with $E_{\mu}^{\text{BCS}} = \sqrt{(\varepsilon_{\mu} - \lambda)^2 + \Delta_{\mu}^2}$ and ε_{μ} and Δ_{μ} are the equivalent single-particle energies and pairing gaps, see appendix B in [9]. The Fermi-Dirac occupation factors are given by

$$f_{\mu}(\lambda) = \frac{1}{1 + e^{\beta E_{\mu}^{\text{BCS}}}}. \quad (5)$$

When using the Newton-like method to solve the equation $N(\lambda) = N, Z$ for each type of particle at $T > 0$, one must now include the contribution $\partial f_{\mu} / \partial \lambda$ in the derivative of the function $N(\lambda)$.

2.4. Linear constraints and the RPA method

Multi-constraint calculations are possible in HFBTHO v2.00d. The code implements the linear constraint method, where the

quantity to be minimized is

$$E' = E - \sum_a \lambda_a (\langle \hat{Q}_{la} \rangle - Q_{la}), \quad (6)$$

where \hat{Q}_{la} is the multipole moment operator for the constraint a and λ_a is the related Lagrange parameter. Lagrange parameters are readjusted at each iteration according to the procedure presented in [10] and also used in the latest release of HFODD [5]. The philosophy of the method is to associate the variation of the Lagrange parameters with a first-order perturbation of the generalized density matrix.

As a reminder, we start with the variations $\delta \mathcal{R}$ of the generalized density matrix, which induce variations of the HFB matrix $\delta \mathcal{H}$ and of the Lagrange parameters $\delta \lambda = (\delta \lambda_1, \dots, \delta \lambda_N)$, (up to first order). Neglecting the variations of the HFB matrix with respect to the generalized density matrix is equivalent to working at the so-called cranking approximation, and it reduces the HFB equation with the perturbed quantities to

$$[\delta \mathcal{R}, \mathcal{H}^{(0)}] - \frac{1}{2} \sum_a \delta \lambda_a [\mathcal{R}^{(0)}, \mathcal{Q}_{la}] = 0, \quad (7)$$

with $\mathcal{R}^{(0)}$ and $\mathcal{H}^{(0)}$, respectively, the unperturbed generalized density matrix and HFB Hamiltonian, $\delta \lambda_a$ the perturbation of the Lagrange parameter for the constraint a , and \mathcal{Q}_{la} the matrix of the constraint in the doubled s.p. basis. This equation gives the desired relation between $\delta \mathcal{R}$ and $\delta \lambda$. The Lagrange parameter can then be readjusted at each iteration by interpreting the deviation $\delta Q_a = \langle \hat{Q}_{la} \rangle - Q_{la}$ from the requested value Q_{la} as caused by a variation of the generalized density matrix $\delta \mathcal{R}$,

$$\delta Q_a = \frac{1}{2} \text{Tr} (\mathcal{Q}_{la} \delta \mathcal{R}). \quad (8)$$

Knowing the deviation δQ_a , we obtain $\delta \mathcal{R}$, and thereby deduce the $\delta \lambda_a$ needed to reproduce the requested value. Calculations are performed in the q.p. basis, since the unperturbed generalized density and HFB matrix take a very simple form. The computational cost of the method thus comes essentially from transforming all relevant matrices into this basis. In HFBTHO, this operation can be performed separately for each Ω -block. The method can also be extended to finite-temperature in a straightforward manner by using the Fermi-Dirac occupation factors. Details of this extension are presented elsewhere [4].

2.5. Quasi-particle blocking

Odd-even and odd-odd nuclei can now be computed with HFBTHO v2.00d using the blocking of quasi-particle states [11]. Because time-reversal symmetry is built into the code, the equal filling approximation (EFA) has to be used [12]. However, it was shown in [11] that the EFA is an excellent approximation to exact blocking. The identification of the blocking candidate is done using the same technique as in HFODD [13]: the mean-field Hamiltonian h is diagonalized at each iteration and provides a set of equivalent single-particle states. Based on the Nilsson quantum numbers of the requested blocked level provided in the input file, the code identifies the index of the q.p. to be blocked by looking at the overlap between the q.p. wavefunction (both lower and upper component separately) and the s.p. wavefunction. The maximum overlap specifies the index of the blocked q.p.

2.6. Generalized energy density functionals

The kernel of the HFBTHO solver has been rewritten to enable the use of generalized Skyrme functionals that are not necessarily

derived from an effective pseudo-potential such as the Skyrme force. Generalized Skyrme functionals are defined here as being the most general scalar, iso-scalar, time-even functional \mathcal{H} of the one-body local density matrix $\rho(\mathbf{r})$ up to second-order in spatial derivatives of ρ [14,15]. Assuming time-reversal symmetry, such functionals thus take the form

$$\begin{aligned} \mathcal{H}_t[\rho] = & C_t^{\rho\rho}[\rho]\rho_t^2 + C_t^{\rho\tau}[\rho]\rho_t\tau_t + C_t^J[\rho]\mathbf{J}_t^2 \\ & + C_t^{\rho\Delta\rho}[\rho]\rho_t\Delta\rho_t + C_t^{\rho\nabla J}[\rho]\rho_t\nabla\cdot\mathbf{J}_t, \end{aligned} \quad (9)$$

where t stands for the isoscalar ($t = 0$) or isovector ($t = 1$) channel, and τ_t and \mathbf{J}_t are the kinetic energy and spin current density in each channel. The terms $C_t^{uu'}[\rho]$ are (possibly arbitrary) functions of the local isoscalar density $\rho_0(\mathbf{r})$. Note that all commonly used Skyrme forces or functionals fall into this category because of the phenomenological density-dependent term. Although most Skyrme functionals have been fitted “as a force”, the recent parameterizations UNEDF0 and UNEDF1 have looked at the problem more from a functional perspective [16,17]. Microscopically-derived EDF obtained, for example, from the density matrix expansion of effective nuclear potentials, are less trivial examples of these generalized functionals, since the density-dependence of the coupling constants can be significant [15].

In the current version, the code only implements 2nd-order generalized Skyrme functionals and it is left to the user to code more advanced functionals. The subroutine `calculate_U_parameters()` in module UNEDF provides a general template for such an implementation. Required are the form of the energy functional and at least its first partial derivatives with respect to the isoscalar ρ_0 and isovector ρ_1 density matrices. Second-order partial derivatives are also necessary to compute nuclear matter properties.

2.7. Shared memory parallelism with OpenMP

To facilitate large-scale applications of the HFBTHO solver on leadership class computers, the original source file has been split into a DFT solver kernel and a calling program. In version 2.00d, we have also parallelized a number of time-intensive routines using OpenMP pragmas. The routine `hfbdiag` diagonalizes the Ω -blocks of the HFB matrix: these diagonalizations are now done in parallel. The routine `coulomb` computes the direct Coulomb potential $V_c(\mathbf{r}, \mathbf{r}')$ at the first iteration: this step is carried out in parallel but saves time at the first iteration only. The routine `gamdel` reconstructs the HFB matrix in configuration space for each Ω -block by computing on-the-fly the various one-dimensional integrals that define the matrix elements: shared memory parallelism is implemented for the outermost loop corresponding to the Ω -blocks.

Fig. 1 shows the performance improvement when using multi-threading. The test was performed on an 8-core Intel Xeon E5-2670 at 2.6 GHz using the Intel Fortran compiler 13.0 and the MKL library 10.3.11 for ^{120}Sn in a full spherical basis of $N_{\text{max}} = 20$ shells, with the SLy4 interaction and a cutoff of $E_{\text{cut}} = 60$ MeV for the q.p. In version 1.66, the number of Gauss–Hermite and Gauss–Laguerre points are hard-coded in the program and are set to $N_{\text{GH}} = N_{\text{GL}} = 22$, and the number of Gauss–Legendre points is set to $N_{\text{GL}} = 30$. We used the same numbers in our test with the version 2.00d.

We note that version 2.00d is slightly slower (per iteration) than version 1.66 if only one thread is used. This additional overhead comes from the calculation of the densities and fields required for generalized Skyrme functionals, combined with the use of the Broyden method, which uses additional linear algebra at each iteration. In general, it is difficult to compare directly the overall

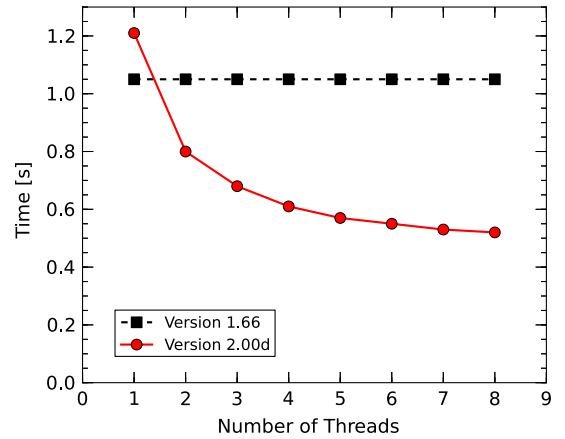


Fig. 1. Time per iteration of a HFB calculation in a full spherical basis of $N_{\text{max}} = 20$ shells as a function of the number of threads, see text for additional details.

performance of the two versions of the code. In version 1.66, calculations at $N_{\text{max}} > 14$ are warm-started automatically with a preliminary calculation at $N_{\text{max}} = 14$. On the other hand, version 2.00d implements the Broyden method, which reduces the number of iterations significantly, see [7]. We emphasize that HFBTHO makes use of BLAS and LAPACK routines and benefits from a threaded implementation of these libraries. Nested parallelism must be supported by the compiler.

3. Benchmarks and accuracy

There exist several comparisons of HFBTHO with other DFT solvers in the literature in both even–even [18,19] and odd nuclei [20,11]. In some cases, these benchmarks compare different approaches to solving the HFB equations, in others the emphasis is put on validation of the solver. Here, we want to gather in one place a comprehensive set of validation and performance evaluations that can be used as a reference in later developments of the code.

3.1. Benchmarks in spherical nuclei: ^{208}Pb and ^{120}Sn

In spherical nuclei, HFBTHO was benchmarked against the spherical DFT solver HOSPHE (version 2.00 of [2]) and the symmetry-unrestricted DFT solver HFODD (version 2.56 of [4]). We study the Hartree–Fock approximation in ^{208}Pb and the Hartree–Fock–Bogoliubov approximation with density-dependent delta pairing forces in ^{120}Sn .

3.1.1. Hartree–Fock computation of ^{208}Pb

In Table 1, we present the results of the benchmarks between the three solvers for the spherical HF point in ^{208}Pb for the SLy5 Skyrme functional of [21]. Calculations were performed in $N_{\text{max}} = 16$ full spherical oscillator shells with a constant oscillator length of $b = 2.0$ fm. In a spherical basis, the oscillator length is related to the oscillator frequency by

$$b = \sqrt{\frac{\hbar}{m\omega}}. \quad (10)$$

In the codes HFODD and HOSPHE, the oscillator frequency is set via a multiplicative factor f such that $\omega = f\omega_0$, with $\omega_0 = 41/A^{1/3}$. The oscillator length is related to f through

$$f = \frac{1}{b^2} \frac{\hbar^2 c^2}{mc^2} \frac{41}{A^{1/3}}, \quad (11)$$

with $mc^2 = 938.90590$ MeV, $\hbar c = 197.328910$ MeV fm. An oscillator length of $b = 2.0$ fm thus corresponds to $f = 1.49831558$

Table 1

Benchmark of the three solvers HOSPHE, HFBTHO and HFODD for a spherical Hartree–Fock calculation in ^{208}Pb with the SLy5 Skyrme functional in a full spherical basis of $N_{\text{max}} = 16$ shells with oscillator length $b = 2.0$ fm. See introduction of Section 3 for additional numerical information.

	HOSPHE	HFBTHO	HFODD
Without Coulomb			
E_{tot} (MeV)	–2445.930216	–2445.930216	–2445.930215
$E_{\text{kin}}^{(n)}$ (MeV)	2614.806852	2614.806852	2614.806852
$E_{\text{kin}}^{(p)}$ (MeV)	1438.160641	1438.160641	1438.160641
E_{Skyrme} (MeV)	–6498.897708	–6498.897708	–6498.897706
E_{SO} (MeV)	–109.091691	–109.091691	–109.091691
$r_{\text{rms}}^{(n)}$ (fm)	5.519846	5.519846	5.519846
$r_{\text{rms}}^{(p)}$ (fm)	5.249812	5.250015	5.250015
With Coulomb			
E_{tot} (MeV)	–1632.591419	–1632.591495	–1632.591454
$E_{\text{kin}}^{(n)}$ (MeV)	2535.409641	2535.409735	2535.409639
$E_{\text{kin}}^{(p)}$ (MeV)	1340.663301	1340.663408	1340.663312
E_{Skyrme} (MeV)	–6306.660514	–6306.660740	–6306.660527
E_{SO} (MeV)	–98.293331	–98.293340	–98.293331
$E_{\text{Cou}}^{(\text{dir})}$ (MeV)	829.308809	829.308760	829.308776
$E_{\text{Cou}}^{(\text{exc})}$ (MeV)	–31.312656	–31.312658	–31.312656
$r_{\text{rms}}^{(n)}$ (fm)	5.608237	5.608237	5.608237
$r_{\text{rms}}^{(p)}$ (fm)	5.448516	5.448711	5.448711

in ^{208}Pb . The number of Gauss–Legendre points for the integration of the Coulomb potential was $N_{\text{Leg}} = 80$, the number of Gauss–Hermite and Gauss–Laguerre integration points was $N_{\text{GH}} = N_{\text{GL}} = 40$, and the Coulomb length scale was $L = 50$ fm, see also Section 3.6 below for a detailed discussion. The Skyrme energy is defined from HFBTHO outputs as

$$E_{\text{Skyrme}} = E^{\text{vol}} + E^{\text{surface}} + E^{\text{SO}} + E^{\text{tensor}}. \quad (12)$$

Without Coulomb potentials included, we note that the difference with HFODD is not greater than 2 eV on energies (E_{Skyrme}), and the radii agree up to at least the 6th digit. Comparisons with HOSPHE show the difference in energies is less than 1 eV, while the proton radius differs by 0.0002 fm. This unexpected deviation may be caused by corrections related to the finite proton size which are first added to the proton radius and then afterwards subtracted. Let us note that the kinetic energy contribution to the total energy is probably the most sensitive to the details of the numerical implementation. With the Coulomb potential included (both direct and exchange), the discrepancy in the total energy is of the order of 100 eV (see also Section 3.6 below).

3.1.2. Hartree–Fock–Bogoliubov computation of ^{120}Sn with the Lipkin–Nogami prescription

In Table 2, we present the results of the benchmarks between the three solvers for the spherical HFB point in ^{120}Sn for the UNEDF0 Skyrme functional of [16]. Calculations were performed with the same basis and integration characteristics as in the previous section. The pairing channel was parameterized by a density-dependent delta-pairing force with mixed volume and surface features, of the general type

$$V_{\text{pair}}^{(n,p)}(\mathbf{r}) = V_0^{(n,p)} \left(1 - \frac{1}{2} \frac{\rho_0(\mathbf{r})}{\rho_c} \right) \delta(\mathbf{r} - \mathbf{r}'), \quad (13)$$

with $V_0^{(n,p)}$ the pairing strength for neutrons (n) and protons (p), $\rho_0(\mathbf{r})$ the isoscalar local density, and ρ_c the saturation density, fixed at $\rho_c = 0.16 \text{ fm}^{-3}$. Let us recall that in the case of the UNEDF parameterizations, the pairing strengths should *not* be adjusted by the user since they were fitted together with the Skyrme

Table 2

Benchmark of the three solvers HOSPHE, HFBTHO and HFODD for a spherical Hartree–Fock–Bogoliubov calculation in ^{120}Sn with the UNEDF0 functional (thus including the Lipkin–Nogami) prescription with a spherical basis of $N_{\text{max}} = 16$ shells with oscillator scale $b = 2.0$ fm ($f = 1.49831558$ in ^{120}Sn). See introduction of Section 3 for additional numerical information.

	HOSPHE	HFBTHO	HFODD
Without Coulomb			
E_{tot} (MeV)	–1374.087678	–1374.087663	–1374.087631
$E_{\text{kin}}^{(n)}$ (MeV)	1384.055506	1384.055495	1384.055733
$E_{\text{kin}}^{(p)}$ (MeV)	885.705977	885.705972	885.705875
E_{Skyrme} (MeV)	–3628.923716	–3628.923682	–3628.924197
E_{SO} (MeV)	–58.837670	–58.837667	–58.837805
$r_{\text{rms}}^{(n)}$ (fm)	4.678179	4.678179	4.678179
$r_{\text{rms}}^{(p)}$ (fm)	4.455211	4.455761	4.455760
E_{pair} (MeV)	–12.646024	–12.646023	–12.645709
$\Delta^{(n)}$ (MeV)	0.910071	0.910071	0.910072
$\Delta^{(p)}$ (MeV)	0.531364	0.531364	0.531338
$\lambda^{(n)}$ (MeV)	–7.333339	–7.333339	–7.333340
$\lambda^{(p)}$ (MeV)	–21.406069	–21.406069	–21.406063
$\lambda_2^{(n)}$ (MeV)	0.081422	0.081422	0.081422
$\lambda_2^{(p)}$ (MeV)	0.633305	0.633305	0.633309
With Coulomb			
E_{tot} (MeV)	–1021.265363	–1021.265407	–1021.265377
$E_{\text{kin}}^{(n)}$ (MeV)	1345.226437	1345.226497	1345.226444
$E_{\text{kin}}^{(p)}$ (MeV)	837.571445	837.571518	837.571444
E_{Skyrme} (MeV)	–3538.591673	–3538.591798	–3538.591661
E_{SO} (MeV)	–48.652094	–48.652102	–48.652100
$E_{\text{Cou}}^{(\text{dir})}$ (MeV)	367.071215	367.071165	367.071184
$E_{\text{Cou}}^{(\text{exc})}$ (MeV)	–19.140103	–19.140104	–19.140103
$r_{\text{rms}}^{(n)}$ (fm)	4.733892	4.733892	4.733892
$r_{\text{rms}}^{(p)}$ (fm)	4.585076	4.585610	4.585609
E_{pair} (MeV)	–11.125231	–11.125231	–11.125231
$\Delta^{(n)}$ (MeV)	0.864875	0.864875	0.864875
$\Delta^{(p)}$ (MeV)	0.481236	0.481237	0.481236
$\lambda^{(n)}$ (MeV)	–7.989573	–7.989572	–7.989573
$\lambda^{(p)}$ (MeV)	–8.286703	–8.286703	–8.286704
$\lambda_2^{(n)}$ (MeV)	0.100481	0.100481	0.100481
$\lambda_2^{(p)}$ (MeV)	0.675087	0.675087	0.675086

coupling constants. Recommended values are, respectively, $V_0^{(n)} = -170.374 \text{ MeV}$ and $V_0^{(p)} = -199.202 \text{ MeV}$. Because of the zero-range of the pairing force, a cutoff in the q.p. space has to be introduced, and we chose $E_{\text{cut}} = 60 \text{ MeV}$ in this example. When compatibility with HFODD is required, this cutoff is sharp, namely all q.p. with $E > E_{\text{cut}}$ are discarded from the calculation of the density.

3.2. Benchmarks in even–even deformed nuclei: ^{240}Pu

Next, we present the benchmark of HFBTHO in deformed even–even nuclei against HFODD. Accurate HFB calculations in deformed nuclei require the use of a suitably deformed, or stretched, HO basis. Such a basis is characterized by its oscillator frequencies, $\omega_x \neq \omega_y \neq \omega_z$ in Cartesian coordinates, and $\omega_{\perp} \neq \omega_z$ in cylindrical coordinates, as well as by the total number of states retained. The goal of this section is to compare basis truncation schemes between HFBTHO and HFODD in a realistic case.

Stretched basis in HFBTHO—It is determined by applying the general prescription given in [22] to the particular case of an axially-deformed prolate basis. Let us recall that the starting point

is an ellipsoid characterized by radii R_x , R_y and R_z . Introducing the spherical radius R_0 and the (β, γ) Bohr quadrupole deformation parameters, we have

$$\begin{aligned} R_x &= R_0 \exp \left\{ \sqrt{\frac{5}{4\pi}} \beta \cos \left(\gamma - \frac{2\pi}{3} \right) \right\}, \\ R_y &= R_0 \exp \left\{ \sqrt{\frac{5}{4\pi}} \beta \cos \left(\gamma + \frac{2\pi}{3} \right) \right\}, \\ R_z &= R_0 \exp \left\{ \sqrt{\frac{5}{4\pi}} \beta \cos(\gamma) \right\}. \end{aligned} \quad (14)$$

The deformation of the basis is characterized, equivalently, by the two parameters p and q such that

$$q = \frac{b_z^2}{b_x^2}, \quad p = \frac{b_y^2}{b_x^2}, \quad (15)$$

with b_μ the oscillator length for coordinate μ . It is then assumed that

$$q = \frac{R_z}{R_x}, \quad p = \frac{R_y}{R_x}. \quad (16)$$

In the special case of an axially-deformed basis ($\beta > 0$, $\gamma = 0^\circ$), we find

$$q = e^{-3\sqrt{\frac{5}{16\pi}}\beta}, \quad p = 1. \quad (17)$$

From the volume conservation condition ($b_\perp^2 b_z = b_0^3$), Eq. (17) leads to

$$b_\perp = b_0 q^{-1/6}, \quad b_z = b_0 q^{+1/3}. \quad (18)$$

Given a “spherical” oscillator length b_0 and the deformation β of the basis, the formula (18) uniquely defines the HO lengths of the stretched basis.

Stretched basis in HFODD—The starting point is a general nuclear shape parameterized by a surface Σ characterized by the deformation parameters $\alpha_{\lambda\mu}$ through

$$R(\theta, \varphi) = R_0 c(\alpha) \left[1 + \sum_{\lambda=2}^{\lambda_{\max}} \sum_{\mu=-\lambda}^{\lambda} \alpha_{\lambda\mu} Y_{\lambda\mu}(\theta, \varphi) \right], \quad (19)$$

where $R_0 = r_0 A^{1/3}$, $c(\alpha)$ is computed to ensure volume conservation, $Y_{\lambda\mu}(\theta, \varphi)$ are the spherical harmonics, and the $\alpha_{\lambda\mu}$ are the deformation parameters. The surface defined by (19) encloses a volume V and the radius of this “ellipsoid” along the direction μ ($=x, y, z$) is determined according to

$$R_\mu \equiv \sqrt{\langle x_\mu^2 \rangle} = \frac{1}{V} \int x_\mu^2 d^3\mathbf{r}. \quad (20)$$

Frequencies of the HO along each Cartesian direction satisfy $\omega_0^3 = \omega_x \omega_y \omega_z$ with

$$\begin{aligned} \omega_x &= \omega_0 (R_{xz} R_{yz}), & \omega_y &= \omega_0 (R_{xz} / R_{xy})^{-1/3}, \\ \omega_z &= \omega_0 (R_{xy} R_{xz})^{1/3}, \end{aligned} \quad (21)$$

with the geometrical ratios $R_{\mu\nu} = R_\mu / R_\nu$.

Discussion — It is straightforward to see that in the particular case of a prolate ellipsoid ($\beta > 0$, $\gamma = 0^\circ$, $R_{xy} = 1/p$, $R_{xz} = R_{yz} = 1/q$), both HFODD and HFBTHO prescriptions to choose the oscillator frequencies are in principle identical. In practice, however, the determination of the radii from Eq. (20) in HFODD produce small numerical deviations compared to the analytic formula (14). This will induce systematic differences between the HO frequencies computed in the two codes, which will in turn alter the selection of the basis states. Fig. 2 quantifies this statement in an extreme case.

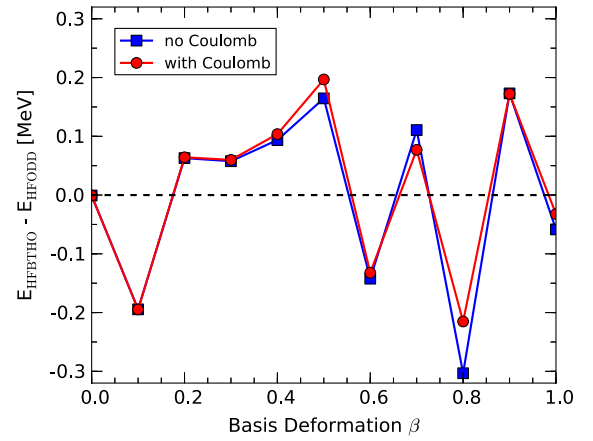


Fig. 2. Difference between HFBTHO and HFODD total energy for a very deformed configuration in ^{240}Pu (see details in text) as a function of the axial deformation of the basis β .

Fig. 2 shows the numerical difference between the two codes for the total energy in ^{240}Pu computed for $N_{\max} = 16$ and $N_{\text{states}} = 500$ as a function of the deformation β of the basis. Calculations are done with a spherical oscillator length $b_0 = 2.3$ fm ($f = 1.18829312$ for ^{240}Pu), the Sly4 parameterization of the Skyrme functional, identical pairing strengths of $V_{\text{pair}} = -300$ MeV for both protons and neutrons, and quadrature precisions of $N_{\text{GL}} = N_{\text{GH}} = 40$ and $N_{\text{Leg}} = 80$. The configuration chosen was obtained by putting a constraint on the quadrupole moment $\langle \hat{Q}_{20} \rangle = 150$ b and hexadecapole moment $\langle \hat{Q}_{40} \rangle = 30$ b²; expectation values of \hat{Q}_{60} and \hat{Q}_{80} vary with the deformation of the basis.

For a configuration with such large deformations, a basis with only 500 states and $N_{\max} = 16$ is not sufficient to reach convergence. In particular, important intruder orbitals are missing. As a result, all physical observables depend quite significantly on basis parameters such as its deformation or frequencies. It is therefore a good test-bench for numerical comparisons and is an illustration of the worst-case scenario.

Two sets of results, with and without Coulomb potentials included, are presented. In contrast to the much simpler cases of Section 3.1, the difference between the two codes reaches up to 300 keV, even without Coulomb terms. This discrepancy is entirely attributable to the slightly different HO frequencies/lengths, the impact of which is magnified by the large deformation of the requested configuration combined with the relatively small size of the HO basis. As an example, for $\beta = 0.7$, the oscillator lengths are $b_\perp = 2.0803$ and $b_z = 2.8114$ fm in HFODD, to be compared with $b_\perp = 2.0596$ and $b_z = 2.8682$ fm in HFBTHO. The Coulomb term does not qualitatively change this picture. Most importantly, if HO lengths are manually enforced to be numerically identical in the two codes, or in the case of a spherical basis, the agreement between the two sets of calculations without Coulomb goes back to the 1 eV level as in the previous sections.

3.3. Benchmark in deformed odd nuclei: ^{159}Ba

The new version of HFBTHO enables blocking calculations in odd-even or odd-odd nuclei. Since by construction HFBTHO conserves time-reversal symmetry, the blocking prescription is implemented in the equal filling approximation, and the time-odd fields of the Skyrme functional are identically zero. In [11], a detailed comparison of blocking calculations between the HFBTHO and HFODD solvers was presented for the case of ^{121}Sn , with a spherical HO basis and identical HO oscillator scales. The goal of this section is to present a benchmark result for an odd-even nu-

Table 3

Results of blocking calculations in HFBTHO and HFODD in ^{159}Ba in a stretched HO basis with $\beta = 0.2$ (see text for more details).

q.p.	Total energy (MeV)	
	HFBTHO	HFODD
[512]5/2	–1236.98 2565	–1236.98 5082
[633]7/2	–1237.3 17554	–1237.3 16510
[503]7/2	–1236.608 320	–1236.608 662
[510]1/2	–1236.44 5682	–1236.44 8800
[521]1/2	–1235.72 4052	–1235.72 6439
[523]5/2	–1235.44 5642	–1235.44 9199
[660]1/2	–1236.293 614	–1236.293 434
[514]7/2	–1235.799 601	–1235.799 806
[651]3/2	–1235.433 781	–1235.434 053

cleus in a deformed basis. As in the previous section, we do *not* manually enforce identical oscillator scales. Instead, we use the same basis selection rules in their respective implementations.

Calculations were performed in the nucleus ^{159}Ba using ^{158}Ba as the even–even core, with the SLy4 Skyrme functional, a mixed surface–volume pairing force with $V_0 = -300$ MeV for both protons and neutrons, and a q.p. cutoff of $E_{\text{cut}} = 60$ MeV. The HO basis was characterized by the oscillator length $b = 2.2$ fm ($f = 1.13221574$ in ^{159}Ba), $N_{\text{max}} = 16$ shells, a deformation of $\beta = 0.2$, and $N_{\text{States}} = 500$. The number of Gauss–Laguerre and Gauss–Hermite quadrature points was $N_{\text{GL}} = N_{\text{GH}} = 40$. In HFODD calculations, time–odd fields were zeroed. Results are presented in Table 3.

Numerical agreement is of the order of 1 keV, with maximum deviations of up to 3.6 keV. Such an agreement is in line with the results shown in the previous two sections. Since the nucleus is not as heavy as ^{240}Pu and the requested configuration is much less deformed than the one considered in 3.2, basis truncation effects are mitigated, and the small discrepancy between the calculated HO oscillator scales does not have as drastic an effect as in the previous section. Again, we note that if identical HO scales are manually enforced, the numerical agreement is of the order of a few eV as shown in [11].

3.4. Transformed harmonic oscillator basis: ^{90}Ni

One of the characteristic features of HFBTHO is the implementation of the transformed harmonic oscillator (THO) basis. We recall that the THO basis functions are generated by applying a local scale transformation (LST) $f(\mathcal{R})$ to the HO single-particle basis functions. The LST transforms every point (ρ, z) by

$$\begin{aligned} \rho &\rightarrow \rho' = \rho \frac{f(\mathcal{R})}{\mathcal{R}}, \\ z &\rightarrow z' = z \frac{f(\mathcal{R})}{\mathcal{R}}, \end{aligned} \quad (22)$$

with the scale $\mathcal{R} = \mathcal{R}(\rho, z)$ defined locally as

$$\mathcal{R} = \sqrt{\frac{\rho^2}{b_{\perp}^2} + \frac{z^2}{b_z^2}}. \quad (23)$$

The LST function f is chosen in such a way as to enforce the proper asymptotic conditions (exponential decay) for the density, according to the general procedure outlined in [23,24]. We refer to [1] for the details of the implementation of the THO method in HFBTHO.

The purpose of this section is to complete our collection of benchmarks by comparing the results obtained in the THO basis produced with HFBTHO with an independent implementation of the method written by one of us (N. Michel) and used in particular in [26,25]. This program assumes spherical symmetry and has been developed independently: comparing the two implementations is a particularly stringent test. To do it, we used HFBTHO to generate

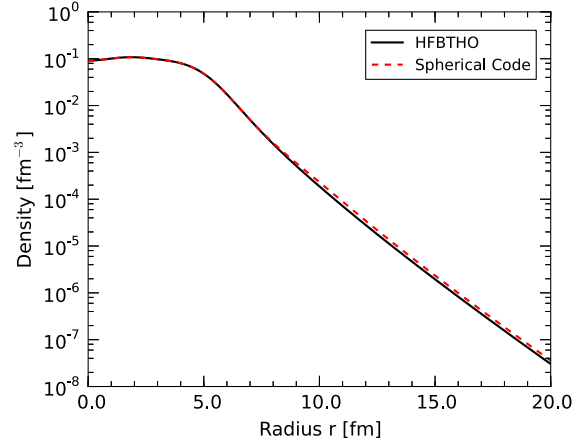


Fig. 3. Radial profile of the neutron density in ^{90}Ni . Black plain line: results from HFBTHO; red dashed line: results from the spherical code.

Table 4

Results of THO calculations in HFBTHO and the spherical code of [25] in ^{90}Ni (see text for more details).

	HFBTHO	N. Michel's Code
E_{Skyrme} (MeV)	–2349.54 7912	–2349.51 1233
E_{SO} (MeV)	–61.590 852	–61.590 694
$E_{\text{kin}}^{(n)}$ (MeV)	1190.96 2911	1190.98 5716
$E_{\text{pair}}^{(n)}$ (MeV)	–58.59 3263	–58.66 4803
$\Delta^{(n)}$ (MeV)	1.91 5093	1.91 6030
$\lambda^{(n)}$ (MeV)	–0.195 904	–0.196 557
$r_{\text{rms}}^{(n)}$ (fm)	4.7173 31	4.7173 75

the LST function $f(\mathcal{R})$ and its partial derivatives on a spatial mesh \mathcal{R}_k with $0 \leq \mathcal{R} \leq 40$ fm by steps of 0.02 fm. These functions were then read numerically by the spherical code.

The test was carried out in the neutron-rich nucleus ^{90}Ni , for the SLy4 Skyrme functional and a pure surface pairing force characterized by $V_0^{(n)} = V_0^{(p)} = -519.9$ MeV with a pairing cut-off of $E_{\text{cut}} = 60$ MeV. Both the HO basis used to generate the THO basis and the THO basis itself were spherical and contained $N_{\text{max}} = 20$ full shells. The oscillator length was fixed at $b = 2.0$ fm ($f = 1.13326033$ in ^{90}Ni). The number of Gauss–Laguerre and Gauss–Hermite quadrature points was $N_{\text{GL}} = N_{\text{GH}} = 40$. Both the direct and exchange Coulomb terms were neglected in these tests. We present in Table 4 various quantities that are good indicators of potential numerical discrepancies.

Overall, the agreement between the two implementations is very good. Indeed, we recall that the matrix elements of the Hamiltonian in the THO basis depend not only on the LST but also on its derivatives $\partial f / \partial \mathcal{R}$. Using a numerically generated LST in the spherical code is, therefore, bound to lead to systematic deviations. We show in Fig. 3 the radial profile of the corrected neutron density after the LST in both HFBTHO and the spherical code. The tiny deviations beyond $r = 8$ fm are the consequence of quantizing the LST in HFBTHO, and using this numerical function in the spherical code instead of a native LST.

3.5. Benchmark at finite temperature: ^{50}Cr

The new version of HFBTHO implements the finite-temperature HFB equations. Table 5 shows the comparison between HFBTHO and HFODD for a simple finite-temperature calculation in ^{50}Cr . The characteristics of the test run (included with the submitted program) were the following: the calculation was performed in a full spherical basis of $N_{\text{max}} = 12$ shells, with an oscillator length of

Table 5

Results of finite-temperature HFB calculations in HFBTHO and HFODD in ^{50}Cr in a spherical HO basis of $N_{\text{max}} = 12$ shells (see text for more details).

	HFBTHO	HFODD
E_{tot} (MeV)	−423.459451	−423.459431
$E_{\text{kin}}^{(n)}$ (MeV)	461.530478	461.530458
$E_{\text{kin}}^{(p)}$ (MeV)	402.309730	402.309693
E_{Skyrme} (MeV)	−1386.929749	−1386.929684
E_{SO} (MeV)	−36.600077	−36.600071
$r_{\text{rms}}^{(n)}$ (fm)	3.591964	3.591964
$r_{\text{rms}}^{(p)}$ (fm)	3.594830	3.594830
$\lambda^{(n)}$ (MeV)	−11.806442	−11.806442
$\lambda^{(p)}$ (MeV)	−6.886468	−6.886468
$S^{(n)}$ (MeV)	6.915625	6.915625
$S^{(p)}$ (MeV)	6.995155	6.995158

$b_0 = 1.7622146$ fm (equivalent to $f = 1.2$), for the SLY4 interaction in the particle–hole channel and the standard surface–volume pairing force of Eq. (13) with $V_0^{(n)} = V_0^{(p)} = -300.0$ MeV and a cutoff of $E_{\text{cut}} = 60.0$ MeV. The temperature was set at $T = 1.5$ MeV. We note that there is a bug in HFODD version 2.49t: the value of the entropy should be multiplied by a factor 2.

3.6. Precision of the Coulomb term

In this section, we discuss in greater detail the precision of the direct term of the Coulomb potential to the total energy. In HFBTHO, the direct term is computed by the Gaussian substitution method. A less accurate method based on the Laplacian substitution method is also available [27].

3.6.1. The Gaussian substitution method

The direct term $V_{\text{Cou}}^{(\text{dir})}(\mathbf{r})$ of the Coulomb potential reads

$$V_{\text{Cou}}^{(\text{dir})}(\mathbf{r}) = e^2 \int d^3\mathbf{r}' \frac{\rho_p(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|}, \quad (24)$$

with ρ_p the proton density. In HFBTHO, the direct term is computed by introducing the following expansion,

$$\begin{aligned} \frac{1}{|\mathbf{r} - \mathbf{r}'|} &= \frac{2}{\sqrt{\pi}} \int_0^\infty e^{-(\mathbf{r}-\mathbf{r}')^2/\mu^2} \frac{d\mu}{\mu^2} \\ &= \frac{2}{\sqrt{\pi}} \int_0^\infty e^{-(\mathbf{r}-\mathbf{r}')^2 a^2} da. \end{aligned} \quad (25)$$

Denoting

$$I_a(\mathbf{r}) = \int d^3\mathbf{r}' e^{-(\mathbf{r}-\mathbf{r}')^2 a^2} \rho_p(\mathbf{r}'), \quad (26)$$

we can write

$$V_{\text{Cou}}^{(\text{dir})}(\mathbf{r}) = e^2 \frac{2}{\sqrt{\pi}} \int_0^\infty I_a(\mathbf{r}) da. \quad (27)$$

The integral over the range a can be performed by Gauss–Legendre quadrature if we introduce the variable $0 \leq \xi < 1$ such that

$$a = \frac{1}{L} \frac{\xi}{\sqrt{1-\xi^2}}, \quad (28)$$

with $L > 0$ an arbitrary length scale. This leads to

$$V_{\text{Cou}}^{(\text{dir})}(\mathbf{r}) = e^2 \frac{2}{\sqrt{\pi}} \frac{1}{L} \int_0^1 \frac{I_{a(\xi)}(\mathbf{r})}{(1-\xi^2)^{3/2}} d\xi. \quad (29)$$

The Coulomb direct energy is then given by

$$E_{\text{Cou}} = \frac{1}{2} \int d^3\mathbf{r} V_{\text{Cou}}^{(\text{dir})}(\mathbf{r}) \rho_p(\mathbf{r}) = \int_0^1 E_{\text{Cou}}(\xi) d\xi, \quad (30)$$

with the integrand

$$E_{\text{Cou}}(\xi) = e^2 \frac{1}{\sqrt{\pi}} \frac{1}{L} \int d^3\mathbf{r} \frac{I_{a(\xi)}(\mathbf{r}) \rho_p(\mathbf{r})}{(1-\xi^2)^{3/2}}. \quad (31)$$

Let us note that the choice (28) for the change of variables is only a particular case of

$$a = \frac{1}{L} \frac{\xi}{(1-\xi^\alpha)^{1/\alpha}}, \quad (32)$$

with α any positive real number. In principle, α could be tuned to maximize the convergence of the Coulomb energy with respect to both the length scale L and/or the number of points in the Legendre quadrature, see below. In practice, the choice $\alpha = 2$ gives the best compromise between accuracy and speed.

3.6.2. Practical implementation in HFBTHO

In practice, the integral (29) is computed numerically by introducing N_{Leg} quadrature abscissae ξ_ℓ and weights w_ℓ ,

$$V_{\text{Cou}}^{(\text{dir})}(\mathbf{r}) = e^2 \frac{2}{\sqrt{\pi}} \frac{1}{L} \sum_{\ell=1}^{N_{\text{Leg}}} w_\ell \frac{I_{a(\xi_\ell)}(\mathbf{r})}{(1-\xi_\ell^2)^{3/2}}. \quad (33)$$

In HFBTHO, all integrals over $\mathbf{r} = (\rho, z, \varphi)$ are computed by Gauss quadrature, with N_{GL} Gauss–Laguerre points for the coordinate $\eta = b_\perp^2 \rho^2$ and N_{GH} Gauss–Hermite points for the coordinate $\xi = b_{zz}$ (see notations in [1]). Following the work of Vautherin [27], the code uses a general method known in electronic structure theory as the pseudospectral representation of the HFB equations [28]. While the HFB equations are solved in the HO basis, i.e. in Fock (or spectral) space, the HF and pairing fields, as well as all expectations values of observables, are computed directly on the quadrature grid, implying constant transformation from/to Fock to/from coordinate space.

Following this philosophy, the calculation of the Coulomb field and energy is somewhat accelerated by introducing the following matrix at the first iteration,

$$V_{ki} = e^2 \frac{2}{\sqrt{\pi}} \frac{1}{L} \omega_i \sum_{\ell=1}^{N_{\text{Leg}}} w_\ell \frac{e^{-(\mathbf{r}_k - \mathbf{r}_i')^2 a(\xi_\ell)^2}}{(1-\xi_\ell^2)^{3/2}}, \quad (34)$$

with L the length scale mentioned in the previous section, k, i compound indexes running from 1 to $N_{\text{GL}} \times N_{\text{GH}}$, and ω_i the product of the weights for both types of quadrature, $\omega_i \equiv w_{\text{GL}} w_{\text{GH}}$. With this notation, the Coulomb field on the grid is obtained at each iteration by vector multiplication $V_k = \sum_i V_{ki} \rho_i$, with ρ_i the vector containing the proton density on the grid. The energy is then obtained by another vector multiplication $E_{\text{Cou}} = \sum_k \omega_k V_k \rho_k$.

It is important to bear in mind that the matrix V_{ik} is a quantized form of the true potential $1/|\mathbf{r} - \mathbf{r}'|$ on the quadrature grid. When the original potential has a singularity, dealing with such quantized representations generate a systematic error that can become arbitrarily large near the singularity. This phenomenon is known as “aliasing” in electronic structure theory [28]. In principle, the error should decrease as the grid becomes larger and larger (closer to the exact integration). In any practical calculation, however, it will always be non-zero.

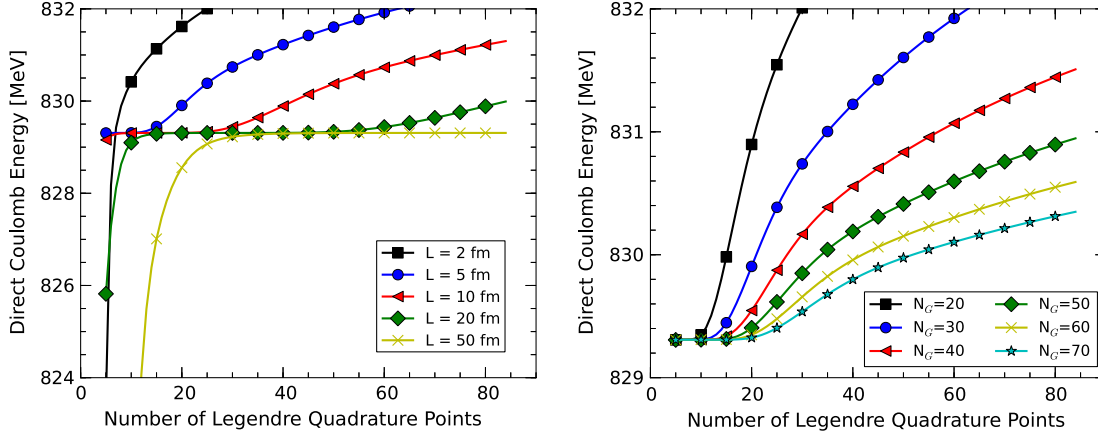


Fig. 4. Left: Direct Coulomb energy as a function of the number of Gauss–Legendre integration points for different values of the length scale L (see text). Right: Same for $L = 5$ fm and different values of the Gauss–Hermite and Gauss–Laguerre quadrature points. All calculations done in ^{208}Pb for $N_{\text{max}} = 16$ shells and the Sly5 Skyrme functional.

3.6.3. Numerical accuracy

Mathematically, expressions (27) and (29) are strictly equivalent. In particular, they do not depend on the length scale L . However, the use of finite quadrature for both the Gauss–Legendre integration of the $1/|\mathbf{r} - \mathbf{r}'|$ function and the spatial integration over coordinates \mathbf{r} and \mathbf{r}' introduce an alias, as mentioned above. In our case, the practical consequence of having an aliased integration is that the Coulomb energy will depend, possibly in a significant way, on the length scale L and the number of Gauss–Legendre quadrature points N_{leg} , but also on the number of Gauss–Hermite and Gauss–Laguerre points N_{GH} and N_{GL} .

In the left panel of Fig. 4, we show the direct Coulomb energy in ^{208}Pb as a function of the number of Gauss–Legendre quadrature points for different length scales L . Calculations were done in a full spherical HO basis with $N_{\text{max}} = 16$ oscillator shells and the Sly5 interaction with $N_{\text{GH}} = N_{\text{GL}} = 30$. The dependence on L is clearly marked. In particular, there is no asymptotic convergence to the true value of the Coulomb potential as the number of Legendre integration points increases. Instead, one observes a plateau condition, the range of which increases with L . In the right panel of Fig. 4, we fix the length scale to $L = 5$ fm, and increase the precision of both Gauss–Hermite and Gauss–Laguerre integrations (by convenience, we choose $N_{\text{GH}} = N_{\text{GL}} \equiv N_G$). This clearly mitigates the dependence of the Coulomb energy on the Legendre quadrature.

It thus appears that the error is not really related to the Legendre integration of Eq. (33) itself. Instead, it seems to be a consequence of using a finite quadrature for spatial integrations, i.e. of dealing with a spurious alias. This effect can be visualized in the behavior of the integrand (31). In Fig. 5, we show the integrand as a function of the variable of integration ξ for two types of quadrature meshes ($N_G = 20$ or $N_G = 70$) and three different numbers of Legendre integration points ($N_{\text{leg}} = 10, 20, 30$). All calculations were done with a length scale $L = 5$ fm. We recall that the integrand should tend to 0 as $\xi \rightarrow 1$ (equivalent to $a \rightarrow +\infty$). While the precision of the quadrature mesh does not really play a role for most of the interval of variation of ξ , we observe that for $\xi \rightarrow 1$, the function begins to bend up for coarse quadrature grids (see right panel). This behavior is clearly nonphysical and is the manifestation of the alias. It can be mitigated by increasing the precision of the quadrature grid, as shown by the dashed lines.

In HFBTHO, we have set $L = 50$ fm and $N_{\text{leg}} = 80$ as default values. For calculations of ground-state properties, it is sufficient to use the default values $N_{\text{GH}} = N_{\text{GL}} = 40$. For calculations of very deformed configurations such as in fission, it is recommended

to increase the precision of Gauss integrations. In future releases of the code, we will implement the calculation of both the direct and exchange Coulomb field in Fock space using Moshinsky transformations, which will eliminate all aliasing errors.

4. Input data file

The input data file format has been entirely changed from version 1.66 to the current version 2.00d. The number of additional features in the new version was the reason to adopt a more flexible format for inputs.

4.1. Sample input file

The new format uses Fortran namelist structure. An example is shown below,

```
&HFBTHO_GENERAL
  number_of_shells = 10, oscillator_length = -1.0,
  basis_deformation = 0.0,
  proton_number = 24, neutron_number = 26,
  type_of_calculation = 1 /
&HFBTHO_ITERATIONS
  number_iterations = 100, accuracy = 1.E-5,
  restart_file = -1 /
&HFBTHO_FUNCTIONAL
  functional = 'SLY4', add_initial_pairing = F,
  type_of_coulomb = 2 /
&HFBTHO_PAIRING
  user_pairing = F, vpair_n = -300.0,
  vpair_p = -300.0, pairing_cutoff = 60.0,
  pairing_feature = 0.5 /
&HFBTHO_CONSTRAINTS
  lambda_values = 1, 2, 3, 4, 5, 6, 7, 8,
  lambda_active = 0, 0, 0, 0, 0, 0, 0, 0,
  expectation_values = 0.0, 0.0, 0.0, 0.0, 0.0,
  0.0, 0.0, 0.0 /
&HFBTHO_BLOCKING
  proton_blocking = 0, 0, 0, 0, 0,
  neutron_blocking = 0, 0, 0, 0, 0 /
&HFBTHO_PROJECTION
  switch_to_THO = 0, projection_is_on = 0,
  gauge_points = 1, delta_Z = 0, delta_N = 0 /
&HFBTHO_TEMPERATURE
  set_temperature = F, temperature = 0.0 /
&HFBTHO_DEBUG
```

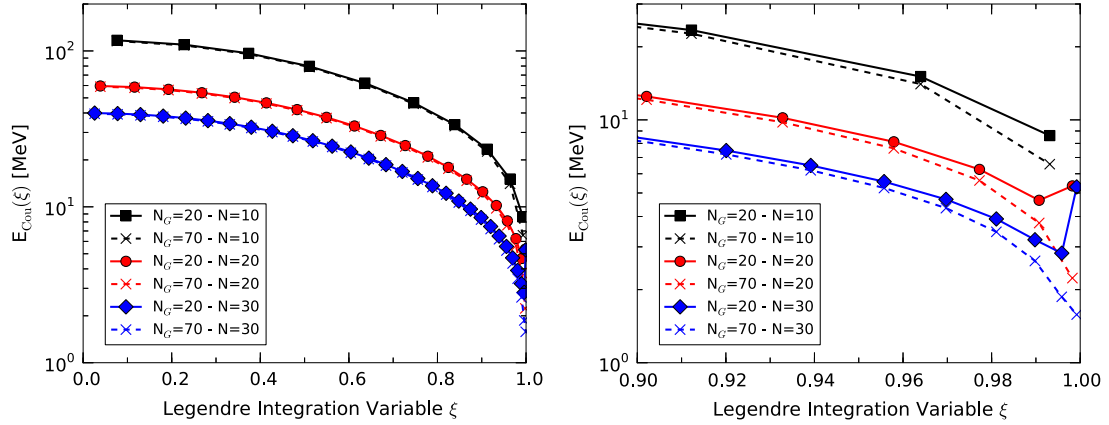


Fig. 5. Left: Integrand $E_{\text{Cou}}(\xi)$ of the Coulomb potential at convergence as function of the Gauss-Legendre integration variable ξ (see text). Right: Close-up on the $\xi \in [0.9, 1.0]$ interval.

```
number_Gauss = 40, number_Laguerre = 40,
number_Legendre = 80,
compatibility_HFODD = F, number_states = 500,
force_parity = T,
print_time = 0 /
```

4.2. Description of input data

We now define the classes of input used in version 2.00d.

Keyword: HFBTHO_GENERAL

- **number_of_shells= 10:** The principal number of oscillator shells N . If the basis is spherical (see below), it is made of the $N_{\text{states}} = (N + 1)(N + 2)(N + 3)/6$ states corresponding to N full shells. If the basis is deformed, the code searches for the lowest N_{states} , with possible intruder contributions from up to the $N_{\text{max}} = 90$ HO shell. Default: 10.
- **oscillator_length= -1.0:** The oscillator length in fm, denoted b_0 in this manuscript, corresponding to the spherical basis. It is related to the HO frequency by $b_0 = \sqrt{\hbar/m\omega_0}$. If the basis is deformed, the code uses the constant volume condition to define the b_z and b_{\perp} oscillator lengths such that $b_0^3 = b_z b_{\perp}^2$. For negative values of b_0 , the code automatically sets b_0 by using $\hbar\omega_0 = 1.2 \times 41/A^{1/3}$. Default: -1.0.
- **basis_deformation= 0.0:** The axial deformation β_2 of the basis. Only axial quadrupole deformations are possible. Negative values correspond to an oblate basis and are allowed. Default: 0.0.
- **proton_number= 24:** Number of protons for this run. Only even values are allowed, see item **proton_blocking** under keyword HFBTHO_BLOCKING for dealing with odd-proton nuclei. Default: 24.
- **neutron_number= 26:** Number of neutrons for this run. Only even values are allowed, see item **neutron_blocking** under keyword HFBTHO_BLOCKING for dealing with odd-neutron nuclei. Default: 26.
- **type_of_calculation= 1:** Defines the type of calculation to be performed for this run. If equal to 1, standard HFB calculations will be performed. If equal to -1, the code will do HFB + LN, where approximate particle-number projection is handled by the Lipkin-Nogami prescription in the seniority pairing approximation following [5]. Default: 1.

Keyword: HFBTHO_ITERATIONS

- **number_iterations= 100:** The maximum number of iterations in the self-consistent loop. Default: 100.

- **accuracy= 1.E-5:** Iterations are stopped when the norm of the HFB matrix between two iterations, $\max \|\mathcal{H}^{(n)} - \mathcal{H}^{(n-1)}\|$, is lower than accuracy, or the number of iterations has exceeded **number_iterations**. Default: 1.E-5.
- **restart_file = -1:** This key can take the values $\pm 1, \pm 2, \pm 3$. If it is negative, calculations will be restarted from an existing solution stored in a HFBTHO compatible binary file. The name of this file will always take the form **[shape] [neutron_number]_ [proton_number] . [extension]**, where **[shape]** is 's' for the value ± 1 , 'p' for the value ± 2 and 'o' for the value ± 3 , and **[extension]** is either 'hel' for regular HO calculations or 'tel' for THO calculations. If the value of the key is positive, calculations will be started from scratch by solving the Schrödinger equation for a Woods-Saxon potential with (possibly) an axial deformation β_2 defined by the value of the constraint on Q_2 , see below. Default: -1.

Keyword: HFBTHO_FUNCTIONAL

- **functional = 'SLY4':** This key with 4 letters indicates the Skyrme functional to be used. Possible values are: 'SIII', 'SKM*', 'SKP', 'SLY4', 'SLY5', 'SLY6', 'SLY7', 'SKI3', 'SKO', 'SKX', 'HFB9', 'UNE0', 'UNE1'. Default: 'SLY4'.
- **add_initial_pairing = F:** In restart mode (see **restart_file**), this Boolean variable decides if a small number will be added to all pairing matrix elements. This option can be useful to ensure that pairing correlations remain non-zero even when restarting from a nucleus where they have collapsed, such as a doubly-magic nucleus. Default: F.
- **type_of_coulomb = 2:** Chooses how the Coulomb potential is treated. If 0, both the direct and exchange terms are neglected. If 1, only the direct Coulomb potential is included in the calculation. If 2, both the direct and exchange Coulomb potentials are included, the exchange term being treated in the Slater approximation. Default: 2.

Keyword: HFBTHO_PAIRING

- **user_pairing = T:** When this keyword is set to T, some characteristics of the pairing interaction can be set by the user. It is always assumed that the pairing force reads

$$V_{\text{pair}}^{n,p}(\mathbf{r}) = V_0^{n,p} \left(1 - \alpha \frac{\rho(\mathbf{r})}{\rho_c} \right) \delta(\mathbf{r} - \mathbf{r}'). \quad (35)$$

Parameters that can be adjusted are the value of the pairing strength for protons and neutrons $V_0^{n,p}$ (which can be different), the cutoff in energies defining the q.p. entering the calculation of the densities, and the type of pairing force defined by the parameter α . When this keyword is set to F, a pre-defined pairing force is used for each Skyrme functional. Default: F.

- `vpair_n` = -300.0: The value of the pairing strength (in MeV) for neutrons V_0^n in Eq. (35). Default: depends on the Skyrme force.
- `vpair_p` = -300.0: The value of the pairing strength (in MeV) for protons V_0^p in Eq. (35). Default: depends on the Skyrme force.
- `pairing_cutoff` = 60.0: The energy cutoff (in MeV) in q.p. space: all q.p. with energy lower than the cutoff are taken into account in the calculation of the densities. Default: 60.0 MeV.
- `pairing_feature` = 0.5: The factor α in Eq. (35). This parameter enables one to tune the properties of the pairing force: If equal to 0, the pairing force has pure volume character and does not depend on the isoscalar density; if set to 1, the pairing force is only active at the surface, since in the interior, $\rho(\mathbf{r}) \approx \rho_c$; if set to 0.5, the pairing force has mixed volume–surface characteristics. Only values between 0 and 1 are possible. Default: 0.5.

Keyword: HFBTHO_CONSTRAINTS

- `lambda_values` = 1, 2, 3, 4, 5, 6, 7, 8: This series of 8 integers define the multipolarity of the multipole moment constraints. It is informational only and is not meant to be changed.
- `lambda_active` = 0, 0, 0, 0, 0, 0, 0, 0: This line defines which of the multipole moments operator \hat{Q}_l , for $l = 1, \dots, 8$, will be used as constraints. When 0, the corresponding multipole is not used as constraint. When 1 it is used, and the resulting constrained HFB calculation is initialized from the diagonalization of the Woods–Saxon potential with the basis deformations. The user can also set this key to -1, which triggers the kickoff mode: the code first performs up to 10 iterations with the constraints specified by the keyword `expectation_values` below, then releases all constraints so as to reach the nearest unconstrained solution. Default: (/ 0, 0, 0, 0, 0, 0, 0, 0 /) (unconstrained calculations).
- `expectation_values` = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0: This line complements the preceding one by specifying the value of the constraint for each multipolarity l . Internally, the units for the multipole moment of order l are $10^l \times \text{fm}^l$. Example: In order to obtain a constraint value of $Q_3 = 5 \text{ b}^{3/2} = 5000 \text{ fm}^3$, the third number must be set to 5.0. Default: (/ 0, 0, 0, 0, 0, 0, 0, 0 /).

Keyword: HFBTHO_BLOCKING

- `proton_blocking` = 0, 0, 0, 0, 0: This group of 5 integers defines the blocking configuration for protons. It takes the form $2\Omega, \pi, N, n_z, n_r$, where $[N, n_z, n_r]\Omega^\pi$ is the traditional Nilsson label. Recall that with time-reversal symmetry, states $+\Omega$ and $-\Omega$ are degenerate, and HFBTHO only considers states with positive values of Ω by default: the sign of 2Ω given above is not related to the actual value of Ω , but to the nucleus in which the blocking is performed. Specifically,
 - If $2\Omega = 0$, the entire group is disregarded (no blocking).
 - If $2\Omega > 0$, blocking is carried out in the nucleus with $Z + 1$ protons, where Z is the value given by the flag `proton_number`. In practice, it means the resulting HFB solution corresponds to the $(Z + 1, N)$ nucleus.
 - If $2\Omega < 0$, blocking is carried out in the nucleus with $Z - 1$ protons, where Z is the value given by the flag `proton_number`. In practice, it means the HFB solution corresponds to the $(Z - 1, N)$ nucleus.

Additionally, the user may request all blocking configurations within 2 MeV of the Fermi level in the even–even core to be computed. This automatization is activated by setting the parity π to 0 instead of ± 1 . For example, the line 1, 0, 0, 0, 0 would compute all blocking configurations in the $(Z + 1, N)$

nucleus, while the line -7, -1, 3, 0, 3 would yield the configuration $[303]7/2^-$ in the $(Z - 1, N)$ nucleus. Refer to the examples included with the program for a practical application. Default: (/ 0, 0, 0, 0, 0 /).

- `neutron_blocking` = 0, 0, 0, 0, 0: This group of 5 integers defines the blocking configuration for neutrons. It obeys the same rules as for the protons. Default: (/ 0, 0, 0, 0, 0 /).

Keyword: HFBTHO_PROJECTION

- `switch_to_THO` = 0: This switch controls the use of the transformed harmonic oscillator basis. If equal to 0, then the traditional HO basis is used; if equal to -1, then the code first performs a calculation in the HO basis before automatically restarting the calculation in the THO basis after the local scale transformation has been determined; if 1, the code runs the calculation in the THO basis only. Note that the use of the THO option requires a large enough basis, typically with at least $N_{\text{max}} = 20$. Default: 0.
- `projection_is_on` = 0: Particle number projection (after variation) is activated by switching this integer to 1. Default: 0.
- `gauge_points` = 1: The implementation of particle number projection is based on the discretization of the integration interval over the gauge angle. The number of gauge points is given here. Default: 1.
- `delta_Z` = 0: If particle projection is on, HFB results will be projected on $Z + \delta Z$, where Z is the actual number of protons in the nucleus and δZ is specified here. Default: 0.
- `delta_N` = 0: If particle projection is on, HFB results will be projected on $N + \delta N$, where N is the actual number of neutrons in the nucleus and δN is specified here. Default: 0.

Keyword: HFBTHO_TEMPERATURE

- `set_temperature` = F: For finite-temperature HFB calculations, `set_temperature` must be set to T. Default: F.
- `temperature` = 0.0: In finite-temperature HFB calculations, the value of the nuclear temperature is given here, in MeV. If `set_temperature` = F, but the nuclear temperature is positive, the code overwrites the flag `set_temperature`. Default: 0.0.

Keyword: HFBTHO_DEBUG

- `number_Gauss` = 40: Number of Gauss–Hermite integration points for integrations along the z-axis (elongation axis). Default: 40 (conserved parity), 80 (broken parity).
- `number_Laguerre` = 40: Number of Gauss–Laguerre integration points for integrations along the perpendicular axis. Default: 40.
- `number_Legendre` = 80: Number of Gauss–Legendre integration points for the calculation of the direct Coulomb potential, see section 3.8 of [1] and Section 3.6 in this manuscript. If this number is negative, the Laplacian substitution method is used instead of the Gaussian substitution method, see [27]. Default: 80.
- `compatibility_HFODD` = F: This Boolean flag enforces the same HO basis as in HFODD. In practice, it is only meaningful in deformed nuclei. Default: F.
- `number_states` = 500: When compatibility with HFODD conventions is enforced, this parameter gives the total number of states in the basis. Default: Inactive.
- `force_parity` = T: This Boolean flag enforces the conservation or breaking of parity depending on the multipolarity of the multipole moments used as constraints. Default: T.
- `print_time` = 0: If 1, the time taken by some of the major routines will be printed in the output. Default: 0.

5. Program HFBTHO v2.00d

The program HFBTHO comes in the form of two files:

- `hfbtho_200d.f90`—Main file containing the self-contained HFBTHO solver. This file contains several Fortran modules, see below.
- `main_200d.f90`—Calling program.

The programming language of most of the code is Fortran 95, while legacy code is still written, in part or totally, in Fortran 90 and Fortran 77. The code HFBTHO requires an implementation of the BLAS and LAPACK libraries to function correctly. Shared memory parallelism is available.

5.1. Fortran source files

The main file `hfbtho_200d.f90` contains the following Fortran modules:

- `HFBTHO_VERSION`: informational module only containing the change log;
- `HFBTHO_utilities`: definition of integer and real number types;
- `linear_algebra`: collection of various routines dealing with interpolation;
- `UNEDF`: module computing the Skyrme-like energy density and the corresponding Hartree–Fock fields at a given density ρ ;
- `HFBTHO`: module storing all public variables used throughout the code;
- `HFBTHO_gauss`: collection of routines and functions dealing with the integration meshes (contains several Fortran 77 legacy routines);
- `HFBTHO_THO`: module in charge of the THO transformation;
- `EllipticIntegral`: module that provides the elliptic integral of the second kind;
- `bessik`: module that provides the modified Bessel function of integer order.

The rest of the routines are not stacked into a module.

5.2. Compilation

The program is shipped with a Makefile that is preset for a number of Fortran compilers. The user should choose the compiler and set the path for the BLAS and LAPACK libraries. To compile, type: “make”.

5.3. Code execution

Assuming an executable named `main` and a Linux system, execution is started by typing

```
“./main < /dev/null > &main.out”.
```

The program will attempt to read the file named `hfbtho_NAMELIST.dat` in the current directory. The user is in charge of assuring this file is present and readable. The code will automatically generate a binary file of the form `[shape]_ [neutron_number]_ [proton_number]_ [extension]` where:

- `[shape]` is one of the letters ‘s’, ‘p’, ‘o’, which refers to spherical, prolate or oblate shape respectively. The choice of this letter is left to the user through the keyword `restart_mode`. This format remains for backward compatibility;
- `[neutron_number]` is a 3-integer number giving the neutron number (left-padding with zero if necessary);
- `[proton_number]` is a 3-integer number giving the proton number (left-padding with zero if necessary);
- `[extension]` is either ‘hel’ (normal HO run) or ‘tel’ (THO run).

Acknowledgments

Discussions with R. Parrish are very warmly acknowledged. This work was supported in part by the Academy of Finland under the Centre of Excellence Programme 2012–2017 (Nuclear and Accelerator Based Physics Programme at JYFL) and FIDIPRO programme, the US Department of Energy grant Nos. DE-FC02-09ER41583 (UNEDF SciDAC program), DE-SC0008499 (NUCLEI SciDAC Collaboration), DE-FC02-07ER41457, DE-FG02-96ER40963 (University of Tennessee), and DE-AC02006CH11357 (Argonne National Laboratory). It was partly performed under the auspices of the US Department of Energy by the Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 (code release number: LLNL-CODE-573953, document release number: LLNL-JRNL-587360). Funding was also provided by the United States Department of Energy Office of Science, Nuclear Physics Program pursuant to Contract DE-AC52-07NA27344 Clause B-9999, Clause H-9999 and the American Recovery and Reinvestment Act, Pub. L. 111-5. An award of computer time was provided by the Innovative and Novel Computational Impact on Theory and Experiment (INCITE) program. This research used resources of the Oak Ridge Leadership Computing Facility located in the Oak Ridge National Laboratory, which is supported by the Office of Science of the Department of Energy under Contract DE-AC05-00OR22725. It was also supported by the National Energy Research Scientific Computing Center supported by the Office of Science of the US Department of Energy under Contract No. DE-AC02-05CH11231. We also acknowledge “Fusion”, a 320-node cluster operated by the Laboratory Computing Resource Center at Argonne National Laboratory, and the CSC—IT Center for Science Ltd, Finland for the allocation of computational resources.

References

- [1] M.V. Stoitsov, J. Dobaczewski, W. Nazarewicz, P. Ring, *Comput. Phys. Comm.* 167 (2005) 43.
- [2] B.G. Carlsson, J. Toivanen, J. Dobaczewski, P. Veselý, Y. Gao, 2013 (in preparation).
- [3] B.G. Carlsson, J. Dobaczewski, J. Toivanen, P. Veselý, *Comput. Phys. Comm.* 181 (2010) 1641.
- [4] N. Schunck, J. Dobaczewski, R. Parrish, W. Satuła, 2013 (in preparation).
- [5] N. Schunck, J. Dobaczewski, J. McDonnell, W. Satuła, J.A. Sheikh, A. Staszczak, M.V. Stoitsov, P. Toivanen, *Comput. Phys. Comm.* 183 (2012) 166.
- [6] J. Dobaczewski, W. Satuła, B.G. Carlsson, J. Engel, P. Olbratowski, P. Powłowski, M. Sadziak, J. Sarich, N. Schunck, A. Staszczak, M.V. Stoitsov, M. Zalewski, H. Zduniczuk, HFODD (v2.40h) User's Guide, 2009. [arXiv:0909.3626](https://arxiv.org/abs/0909.3626).
- [7] A. Baran, A. Bulgac, M. McNeil Forbes, G. Hagen, W. Nazarewicz, N. Schunck, M.V. Stoitsov, *Phys. Rev. C* 78 (2008) 014318.
- [8] M. Abramowitz, I.A. Stegun, *Handbook of Mathematical Functions*, Dover Publications, 1964.
- [9] J. Dobaczewski, H. Flocard, J. Treiner, *Nuclear Phys. A* 422 (1984) 103.
- [10] W. Younes, D. Gogny, *Phys. Rev. C* 80 (2009) 054313.
- [11] N. Schunck, J. Dobaczewski, J. Moré, J. McDonnell, W. Nazarewicz, J. Sarich, M.V. Stoitsov, *Phys. Rev. C* 81 (2010) 024316.
- [12] S. Perez-Martin, L.M. Robledo, *Phys. Rev. C* 78 (2008) 014304.
- [13] J. Dobaczewski, W. Satuła, B.G. Carlsson, J. Engel, P. Olbratowski, P. Powłowski, M. Sadziak, J. Sarich, N. Schunck, A. Staszczak, M.V. Stoitsov, M. Zalewski, H. Zduniczuk, *Comput. Phys. Comm.* 180 (2009) 2361.
- [14] J. Dobaczewski, J. Dudek, *Phys. Rev. C* 52 (1995) 1827.
- [15] M.V. Stoitsov, M. Kortelainen, S.K. Bogner, T. Duguet, R.J. Furnstahl, B. Gebremariam, N. Schunck, *Phys. Rev. C* 82 (2010) 054307.
- [16] M. Kortelainen, T. Lesinski, J. Moré, W. Nazarewicz, J. Sarich, N. Schunck, M.V. Stoitsov, S. Wild, *Phys. Rev. C* 82 (2010) 024313.
- [17] M. Kortelainen, J. McDonnell, W. Nazarewicz, P.-G. Reinhard, J. Sarich, N. Schunck, M.V. Stoitsov, S. Wild, *Phys. Rev. C* 85 (2012) 024304.
- [18] J. Dobaczewski, M.V. Stoitsov, W. Nazarewicz, *AIP Conf. Proc.* 726 (2004) 52.
- [19] J.C. Pei, M.V. Stoitsov, G.I. Fann, W. Nazarewicz, N. Schunck, F.R. Xu, *Phys. Rev. C* 78 (2008) 064306.
- [20] N. Schunck, M.V. Stoitsov, W. Nazarewicz, N. Nikolov, *AIP Conf. Proc.* 1128 (2009) 40.
- [21] E. Chabanat, P. Bonche, P. Haensel, J. Meyer, R. Schaeffer, *Nuclear Phys. A* 635 (1998) 231.

- [22] M. Girod, B. Grammaticos, Phys. Rev. C 27 (1983) 2317.
- [23] M.V. Stoitsov, W. Nazarewicz, S. Pittel, Phys. Rev. C 58 (1998) 2092.
- [24] M.V. Stoitsov, J. Dobaczewski, W. Nazarewicz, S. Pittel, D.J. Dean, Phys. Rev. C 68 (2003) 054312.
- [25] N. Michel, K. Matsuyanagi, M. Stoitsov, Phys. Rev. C 78 (2008) 044319.
- [26] M. Stoitsov, N. Michel, K. Matsuyanagi, Phys. Rev. C 77 (2008) 054301.
- [27] D. Vautherin, Phys. Rev. C 7 (1973) 296.
- [28] R. Friesner, Chem. Phys. Lett. 116 (1985) 39.