POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Title

Tesi di Laurea Magistrale in
Xxxxxxx Engineering - Ingegneria Xxxxxxx

Author: **Name Surname**

Student ID: 000000
Advisor: Prof. Name Surname
Co-advisors: Name Surname, Name Surname
Academic Year: 20XX-XX

Abstract

Here goes the Abstract in English of your thesis followed by a list of keywords. The Abstract is a concise summary of the content of the thesis (single page of text) and a guide to the most important contributions included in your thesis. The Abstract is the very last thing you write. It should be a self-contained text and should be clear to someone who hasn't (yet) read the whole manuscript. The Abstract should contain the answers to the main scientific questions that have been addressed in your thesis. It needs to summarize the adopted motivations and the adopted methodological approach as well as the findings of your work and their relevance and impact. The Abstract is the part appearing in the record of your thesis inside POLITesi, the Digital Archive of PhD and Master Theses (Laurea Magistrale) of Politecnico di Milano. The Abstract will be followed by a list of four to six keywords. Keywords are a tool to help indexers and search engines to find relevant documents. To be relevant and effective, keywords must be chosen carefully. They should represent the content of your work and be specific to your field or sub-field. Keywords may be a single word or two to four words.

Keywords: here, the keywords, of your thesis

Abstract in lingua italiana

Qui va l'Abstract in lingua italiana della tesi seguito dalla lista di parole chiave.

Parole chiave: qui, vanno, le parole chiave, della tesi

Contents

1 | Energy functional

1.1. Hartree-Fock theory

An empirical description of nuclear structure can be carried out using phenomenological models, as reported in section (REF).
A more rigorous approach needs to take into account the fact that the mean field which the nucleons interact with, is generated by the nucleons themselves, due to some microscopic interaction. Starting from the many-body hamiltonian of the system, we will be able to extract a single particle hamiltonian, where the nucleon is subject to a mean field potential generated by the effective microscopic force.
The many-body hamiltonian of the system, made of A fermions, given by

$$\hat{H} = \hat{T} + \hat{V} = \sum_i -\frac{\hbar^2}{2m}\nabla_i^2 + \sum_{i<j} v_{ij}^{(2)} + \sum_{i<j<k} v_{ijk}^{(3)} + \dots \tag{1.1}$$

With the Schrödinger equation of the many-body system given by

$$\hat{H}\Psi = E\Psi. \tag{1.2}$$

1.1.1. Variational principle

Since $\hat{H}$ is a many body operator, finding its eigenstates would be a rather challenging task. To our aid, comes the variational principle, from which we can show that equation 1.2 is equivalent to

$$\delta E[\Psi] = \delta \frac{\langle\Psi|\, E\,|\Psi\rangle}{\langle\Psi|\Psi\rangle} = 0. \tag{1.3}$$

The variation 1.3 can be obtained from an arbitrary variation of Ψ, which can be done indipendently on $\langle\Psi|$ and $|\Psi\rangle$, since Ψ is complex, yielding

$$\langle\delta\Psi|\, \hat{H} - E\,|\Psi\rangle + \langle\Psi|\, \hat{H} - E\,|\delta\Psi\rangle = 0 \tag{1.4}$$

since the variation is arbitrary, we can multiply by a phase factor $|\delta\Psi\rangle \mapsto i\,|\delta\Psi\rangle$ and get

$$-i\,\langle\delta\Psi|\,\hat{H}-E\,|\Psi\rangle + i\,\langle\Psi|\,\hat{H}-E\,|\delta\Psi\rangle = 0. \tag{1.5}$$

Combining equations 1.4 and 1.5, we get

$$\langle\delta\Psi|\,\hat{H}-E\,|\Psi\rangle = 0. \tag{1.6}$$

Again, since the variation is arbitrary, equation 1.6 satisfies equation 1.2.

Ground state

Since we always restrict ourselves to a certain subspace of the full Hilbert space, we can only find an approximate solution to the eigenvalue problem. Expanding this solution on the complete set of exact eigenstates of $\hat{H}$, we have

$$|\Psi\rangle = \sum_n a_n\,|\Psi_n\rangle \tag{1.7}$$

the total energy amounts to

$$E[\Psi] = \frac{\sum_{nn'}\langle a_{n'}\Psi_{n'}|\,\hat{H}\,|a_n\Psi_n\rangle}{\sum_{nn'}\langle a_{n'}\Psi_{n'}|a_n\Psi_n\rangle} = \frac{\sum_n E_n|a_n|^2}{\sum_n|a_n|^2} \geq \frac{\sum_n E_0|a_n|^2}{\sum_n|a_n|^2} \geq E_0. \tag{1.8}$$

where the orthonormality $\langle\Psi_{n'}\,|\Psi_n\rangle| = \delta_{nn'}$ has been used. Equation 1.8 tells us that the minimum of the variational subspace we are considering is bound from below by the true ground state energy.

1.1.2. Hartree-Fock equations

The Hartree-Fock method is the application of the variational principle 1.6 to a many-body wavefunction represented by a Slater determinant, which reads

$$\Psi = \frac{1}{\sqrt{A!}}\sum_{\{p\}}(-1)^p\varphi_{p(1)}(\boldsymbol{r}_1)\ldots\varphi_{p(A)}(\boldsymbol{r}_A) \tag{1.9}$$

where φ_i are single-particle states. The slater determinant sums over all possible permutations of the A fermions on the single particle states, with a $-$ sign according to the parity of the permutation. The Slater determinant is just the assumption of independent single-particle fermion states, so that the Pauli exclusion principle is not violated.

To ensure the orthonormality of the single-particle states, we need to add a Lagrange multiplier to the variation of E, which ends up reading

$$\delta\left(E[\Psi] - \sum_i \lambda_i \int \varphi_i^* \varphi_i d\boldsymbol{r}\right) = 0 \tag{1.10}$$

We can explicit $E[\Psi] = \langle\Psi|\,\hat{T} + \hat{V}\,|\Psi\rangle$, which can be expressed using the single-particle states

$$\langle\Psi|\,\hat{T}\,|\Psi\rangle = \sum_i^A -\frac{\hbar^2}{2m}\int \varphi_i^*(\boldsymbol{r})\nabla^2\varphi_i(\boldsymbol{r})d\boldsymbol{r} \tag{1.11}$$

$$\langle\Psi|\,\hat{V}\,|\Psi\rangle = \frac{1}{2}\sum_{ij}\int \varphi_i^*(\boldsymbol{r})\varphi_j^*(\boldsymbol{r}')v_{ij}(\boldsymbol{r},\boldsymbol{r}')\varphi_i(\boldsymbol{r})\varphi_j(\boldsymbol{r}')d\boldsymbol{r}d\boldsymbol{r}' \tag{1.12}$$

$$-\frac{1}{2}\sum_{ij}\int \varphi_i^*(\boldsymbol{r})\varphi_j^*(\boldsymbol{r}')v_{ij}(\boldsymbol{r},\boldsymbol{r}')\varphi_i(\boldsymbol{r}')\varphi_j(\boldsymbol{r})d\boldsymbol{r}d\boldsymbol{r}' \tag{1.13}$$

If we use $\varphi_i^*(\boldsymbol{r})$ as the quantity to be varied for equation 1.10, we get the *Hartree-Fock equations*

$$-\frac{\hbar^2}{2m}\nabla^2\varphi_i \tag{1.14}$$

$$+\sum_j^A \int \varphi_j^*(\mathbf{r}')v_{ij}(\mathbf{r},\mathbf{r}')\,\varphi_j(\mathbf{r}')\,\varphi_i(\mathbf{r})\,d\mathbf{r}' \tag{1.15}$$

$$-\sum_j^A \int \varphi_j^*(\mathbf{r}')\,v_{ij}(\mathbf{r},\mathbf{r}')\,\varphi_j(\mathbf{r})\,\varphi_i(\mathbf{r}')\,d\mathbf{r}' = \varepsilon_i\varphi_i \tag{1.16}$$

here the Lagrange multipliers λ_i have been replaced by ε_i, since they can be interpreted as the energy of the single-particle states.
Now, a couple of observations are in order.

Exchange interaction The interaction term 1.15, called Hartree term, arises from considering independent particles, and is also routinely found in classical physics. The other one in 1.16, called Fock term, or exchange term, takes the form of an integral operator and is given by the quantum mechanical nature of the problem.
For what concerns the solution of equation 1.14, the Fock term is very problematic, and is avoided using finite-range interaction like the Gogny force [28], or contact forces like the Skyrme one [32], which is used in the present work, since they render the exchange term as a local one.

Self-consistent solution Even if the interaction terms are local, the equation is still highly non-linear, since the mean field potential will be a function of the eigenfunctions

themselves. The consequece is that the solution will be found *self-consistently*, that is, by solving for the set of eigenfunctions $\{\varphi_i\}$, using them to build the new mean field, and solving again, repeating the process until convergence.

1.1.3. Symmetries

Since the objective of this work is to solve the Hartree-Fock equations without spatial symmetry assumptions, it is useful to first understand how symmetries propagate along the self-consistent calculation.
We start by defining the creation and annihilation operators of the single particle hamiltonian eigenstates, $a_i^\dagger, a_i$, which abide the usual anticommutation relations of fermions

$$\{a_i, a_j^\dagger\} = \delta_{ij} \tag{1.17}$$

If we expand on a different, orthonormal complete basis $\{\chi_l\}$, we can write the corresponding creation and annihilation operators $c_l^\dagger, c_l$ as

$$\varphi_k = \sum_l D_{lk}\chi_l \tag{1.18}$$

$$a_k^\dagger = \sum_l D_{lk}c_l^\dagger \tag{1.19}$$

$$a_k = \sum_l D_{lk}^\dagger c_l \tag{1.20}$$

Since orthonormality is guaranteed for both sets, taking

$$\delta_{jk} = \langle\varphi_j|\varphi_k\rangle = \sum_{ll'} D_{l'j}^\dagger D_{lk} \langle\chi_l|\chi_l'\rangle \implies DD^\dagger = 1. \tag{1.21}$$

We can define the density matrix as

$$\rho_{ll'} = \langle\Psi| c_{l'}^\dagger c_l |\Psi\rangle \tag{1.22}$$

which in the hartree fock basis, using relations 1.18, is diagonal and its trace is equal to the particle number A.
Writing the many body hamiltonian 1.2 in the arbitrary basis of second quantization operators $c_l^\dagger, c_l$, we get

$$\hat{H} = \sum_{l_1 l_2} t_{l_1 l_2} c_{l_1}^\dagger c_{l_2} + \frac{1}{4}\sum_{l_1 l_2 l_3 l_4} \overline{v}_{l_1 l_2 l_3 l_4} c_{l_1}^\dagger c_{l_2}^\dagger c_{l_3} c_{l_4} \tag{1.23}$$

where $t_{l_1 l_2}$ and $\overline{v}_{l_1 l_2 l_3 l_4}$ are defined as

$$t_{l_1 l_2} = \langle - | \, c_{l_1} c_{l_2} t c^\dagger_{l_1} c^\dagger_{l_2} \, | - \rangle = \langle l_1 l_2 | \, t \, | l_1 l_2 \rangle \tag{1.24}$$

$$\overline{v}_{l_1 l_2 l_3 l_4} = \langle l_1 l_2 l_3 l_4 | \, v \, | l_1 l_2 l_3 l_4 \rangle - \langle l_1 l_2 l_4 l_3 | \, v \, | l_1 l_2 l_4 l_3 \rangle \tag{1.25}$$

The minimization 1.10 can be restated as the variation of $\langle \Psi | \, \hat{H} \, | \Psi \rangle$, with respect to the density matrix $\rho_{ll'}$, which yields the single particle hamiltonian h

$$h_{ll'} = \frac{\partial E[\rho]}{\partial \rho_{ll'}} = t + \sum_{kk'} \overline{v}_{lk'l'k} \rho_{kk'} = t + \Gamma_{ll'}. \tag{1.26}$$

Being h diagonal in the Hartree-Fock basis, the self-consistent solution is the one for which

$$[h, \rho] = 0 \tag{1.27}$$

holds.

Symmetry propagation

Suppose to start a Hartree-Fock calculation with an initial guess $|\Psi\rangle^{(0)}$ for which the corresponding density matrix is symmetric under the action of a many-body symmetry operator S which commutes with the Hamiltonian

$$[S, \hat{H}] = 0. \tag{1.28}$$

It can be shown [27] that

$$S\Gamma[\rho]S^\dagger = \Gamma[S\rho S^\dagger]. \tag{1.29}$$

The single particle Hamiltonian h will then display the same property

$$Sh[\rho]S^\dagger = h[S\rho S^\dagger] \tag{1.30}$$

meaning that h will be symmetric under the action of S, as well as the next iteration's density matrix $\rho^{(1)}$. The symmetry S gets propagated self-consistently until the minimum is found.

This has profound numerical implications, since the real minimum of deformed nuclei can be found only starting from starting guesses with similar broken symmetries. It can be the case that numerical noise allows to explore the full energy surface, but if one has to take into consideration the numerical cost of a bad guess, then it's still advantageous to

start from a good one.

1.1.4. Density Functional Theory

It shall be evident shortly, in section 1.2, that a more general approach to microscopic models has to be taken, in order to give a satisfactory description of the nuclear world. The framework that we'll briefly outline here is called Density Functional Theory (DFT). DFT was introduced by P. Hohenberg and W. Kohn in 1964 [16], by proving two theorems. The **first** HK theorem states that the energy of a fermion system, subject to an external potential V_{ext} can be expressed solely as a functional of the particle density ρ of the system.

$$E[\rho] = F[\rho] + \int V_{\text{ext}}\rho(\boldsymbol{r})d\mathbf{r} \tag{1.31}$$

While the **second** one states that the ground state of the system is found by minimizing its variation with respect to ρ.
HK theorems are fundamental but not constructive [9], since they do not provide a form for the functional F, which is intrinsic to the physics of the fermions at hand. COMPLETARE

1.2. Skyrme

Now that the theoretical framework is clear, we can investigate a plausible nucleonic interaction, which in the present work, takes the form of the Skyrme force.

1.2.1. Skyrme force

It was first proposed by Tony Skyrme in 1958 [32] as a zero range force between nucleons, and has been used successfully as the building block of theoretical nuclear structure. It comprises a two body attractive term that reads

$$\begin{aligned} v^{(2)}(\mathbf{r}_1,\mathbf{r}_2) = {} & t_0\left(1+x_0P_\sigma\right)\delta(\mathbf{r}) \\ & + \frac{1}{2}t_1\left(1+x_1P_\sigma\right)\left[\mathbf{P}'^2\delta(\mathbf{r}) + \delta(\mathbf{r})\mathbf{P}^2\right] \\ & + t_2P_\sigma\mathbf{P}'\cdot\delta(\mathbf{r})\mathbf{P} \\ & + iW_0\boldsymbol{\sigma}\cdot\left[\mathbf{P}'\times\delta(\mathbf{r})\mathbf{P}\right] \end{aligned}$$

And a three body interaction, that is

$$v^{(3)}(\boldsymbol{r}_1,\boldsymbol{r}_2,\boldsymbol{r}_3) = t_3\delta(\boldsymbol{r}_1-\boldsymbol{r}_2)\delta(\boldsymbol{r}_2-\boldsymbol{r}_3)$$

which mimics the repulsive three-body force; without it, a collapse of the nuclear density would occur.
It's trivial to show that the three-body term is equivalent to a two-body, density-dependent interaction: [37]

$$v^{(3)}(\boldsymbol{r}_1, \boldsymbol{r}_2) = \frac{1}{6} t_3 (1 + P_\sigma) \delta(\boldsymbol{r}) \rho(\boldsymbol{R}) \tag{1.32}$$

The different operators here are defined as

$$\mathbf{r} = \mathbf{r}_1 - \mathbf{r}_2 \quad \mathbf{R} = \frac{\mathbf{r}_1 + \mathbf{r}_2}{2} \tag{1.33}$$

which are respectively the relative position of two particles and their center of mass coordinate, assuming equal masses.

$$\mathbf{P} = \frac{-i(\nabla_1 - \nabla_2)}{2} \tag{1.34}$$

which is the so called relative wave-number operator.

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}_1 + \boldsymbol{\sigma}_2 \tag{1.35}$$

being the total spin of the two interacting particles, and lastly

$$\mathbf{P}_\sigma = \frac{(1 + \boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2)}{2} \tag{1.36}$$

which represents the spin-exchange operator. Primed operators refer to the adjoint acting on the left.
The zero-range characteristic takes the form of a Dirac delta $\delta(\mathbf{r})$, which allows the writing of the Fock term detailed in (REF) as a purely local one.
Taking the expectation value of the many body hamiltonian, in the Hilbert space of Slater determinants, yields an energy density which can be expressed as a function of $\rho_q, \tau_q, \boldsymbol{J}_q$ [37].

$$\langle H \rangle = \langle \Psi | \, H \, | \Psi \rangle = \int \mathcal{H}(\boldsymbol{r}) d\mathbf{r} \tag{1.37}$$

Modern parametrization

The Skyrme force has evolved from the original one to accomodate new nuclei, done through the addition of a few parameters, yielding the following form of the interaction

[5]

$$
\begin{aligned}
v^{(2)}(\mathbf{r}_1,\mathbf{r}_2) = {} & t_0\left(1+x_0P_\sigma\right)\delta(\mathbf{r}) \\
& + \frac{1}{2}t_1\left(1+x_1P_\sigma\right)\left[\mathbf{P}'^2\delta(\mathbf{r}) + \delta(\mathbf{r})\mathbf{P}^2\right] \\
& + t_2\left(1+x_2P_\sigma\right)\mathbf{P}'\cdot\delta(\mathbf{r})\mathbf{P} \\
& + \frac{1}{6}t_3\left(1+x_3P_\sigma\right)\left[\rho(\mathbf{R})\right]^\sigma\delta(\mathbf{r}) \\
& + iW_0\boldsymbol{\sigma}\cdot\left[\mathbf{P}'\times\delta(\mathbf{r})\mathbf{P}\right] \\
& + \frac{1}{6}t_3\left(1+x_3P_\sigma\right)\left[\rho(\mathbf{R})\right]^\sigma\delta(\mathbf{r})
\end{aligned}
$$

Here, the boundary between Hartree-Fock and DFT starts to thin out, as the exponent σ of the density makes that piece of the force a true three-body interaction only for the value $\sigma = 1$ [11].
On top of that, additional, empirical tweaking of the resulting energy density needed to reach satisfactory physical accuracy, such as the case for the spin-orbit couplings [26], prompts for the following, well established proceeding: use the Skyrme interaction as a starting guide for building the energy functional to employ Density Functional Theory.

1.2.2. Skyrme functional

The energy functional we want to minimize is rather complex, due to the rich phenomenology of nuclear interactions and numerical nuances.
The complete energy functional is

$$E_{\mathrm{HF}} = E_{\mathrm{Skyrme}} + E_{\mathrm{Coul}} + E_{\mathrm{Kin}} \tag{1.38}$$

We'll begin by looking at the Skyrme and kinetic parts, while later on give a treatment for the Coulomb one.

$$\langle H\rangle = \langle\Psi|\,H\,|\Psi\rangle = \int(\mathcal{E}_{\mathrm{Skyrme}} + \mathcal{E}_{\mathrm{Kin}})d\mathbf{r} = \int\mathcal{E}d\mathbf{r} \tag{1.39}$$

Since the Skyrme interaction is time-reversal invariant (as well as the kinetic operator), this means that the total Hamiltonian must be time-even. Different additive contributions must be time-even as well, even if the densities from which they are calculated are not. Unless we reduce to the even-even nucleus case, where total angular momentum is defined at $J = 0$, giving vanishing time-odd densities. This allows us to write the functional in a

simpler form as [35]

$$\mathcal{E}_{\text{Kin}} = \frac{\hbar^2}{2m}\tau \tag{1.40}$$

$$\mathcal{E}_{\text{Skyrme}} = \sum_{t=0,1}\left\{C_t^{\rho}[\rho_0]\rho_t^2 + C_t^{\Delta\rho}\rho_t\nabla^2\rho_t + C_t^{\nabla\cdot J}\rho_t\nabla\cdot\mathbf{J}_t + C_t^{\tau}\rho_t\tau_t\right\} \tag{1.41}$$

Here, $t = 0, 1$ refers to the isoscalar and isovector components of the densities, e.g.

$$\rho_0 = \rho_p + \rho_n$$

$$\rho_1 = \rho_p - \rho_n$$

Where

$$C_0^{\rho} = +\frac{3}{8}t_0 + \frac{3}{48}t_3\rho_0^{\sigma} \tag{1.42}$$

$$C_1^{\rho} = -\frac{1}{8}t_0(1+2x_0) - \frac{1}{48}t_3(1+x_3)\rho_0^{\sigma} \tag{1.43}$$

$$C_0^{\tau} = +\frac{3}{16}t_1 + \frac{1}{16}t_2(5+4x_2) \tag{1.44}$$

$$C_1^{\tau} = -\frac{1}{16}t_1(1+2x_1) + \frac{1}{16}t_2(1+2x_2) \tag{1.45}$$

$$C_0^{\Delta\rho} = -\frac{9}{64}t_1 + \frac{1}{64}t_2(5+4x_2) \tag{1.46}$$

$$C_1^{\Delta\rho} = +\frac{3}{64}t_1(1+2x_1) + \frac{1}{64}t_2(1+2x_2) \tag{1.47}$$

$$C_0^{\nabla\cdot J} = -\frac{3}{4}W_0 \tag{1.48}$$

$$C_1^{\nabla\cdot J} = -\frac{1}{4}W_0 \tag{1.49}$$

As outlined in previous chapters (REF), we can now derive the Kohn-Sham equations, by constraining orthonormality and enforcing the variation of the functional to be zero. What we end up with is

$$\left[-\nabla\left(\frac{\hbar^2}{2m_q^*(\mathbf{r})}\nabla\right) + U_q(\mathbf{r}) + \delta_{\text{q,proton}}U_C(\mathbf{r}) - i\mathbf{B}_q(\mathbf{r})\cdot(\nabla\times\boldsymbol{\sigma})\right]\varphi_\alpha = \varepsilon_\alpha\varphi_\alpha \tag{1.50}$$

The index $q = n, p$ refers respectively to the neutron and proton quantities.
Where the different terms are given by

$$\frac{\hbar^2}{2m_q^*(\mathbf{r})} = \frac{\delta \mathcal{E}}{\delta \tau_q} \tag{1.51}$$

$$U_q(\mathbf{r}) = \frac{\delta \mathcal{E}}{\delta \rho_q} \tag{1.52}$$

$$\mathbf{B}_q(\mathbf{r}) = \frac{\delta \mathcal{E}}{\delta \mathbf{J_q}} \tag{1.53}$$

The coulomb field U_C, which is present only in the single particle equation for protons, doesn't come from the skyrme interaction, rather from the Coulomb part of the whole functional. It will be properly derived in section 1.3.
Following the rules for functional derivatives, outlined in the appendix (REF) we get

$$\frac{\hbar^2}{2m_q^*(\mathbf{r})} = + \frac{\hbar^2}{2m} \tag{1.54}$$

$$+ \frac{1}{8}[t_1(2+x_1) + t_2(2+x_2)]\rho(\mathbf{r}) \tag{1.55}$$

$$- \frac{1}{8}[t_1(1+2x_1) + t_2(1+2x_2)]\rho_q(\mathbf{r}) \tag{1.56}$$

$$\tag{1.57}$$

$$U_q(\mathbf{r}) = + \frac{1}{8}[t_1(2+x_1) + t_2(2+x_2)]\rho \tag{1.58}$$

$$+ \frac{1}{8}[t_2(1+2x_2) - t_1(1+2x_1)]\rho_q \tag{1.59}$$

$$+ \frac{1}{8}[t_1(2+x_1) + t_2(2+x_2)]\tau \tag{1.60}$$

$$+ \frac{1}{8}[t_2(1+2x_2) - t_1(1+2x_1)]\tau_q \tag{1.61}$$

$$+ \frac{1}{16}[t_2(2+x_2) - 3t_1(2+x_1)]\nabla^2\rho \tag{1.62}$$

$$+ \frac{1}{16}[3t_1(2x_1+1) + t_2(2x_2+1)]\nabla^2\rho_q \tag{1.63}$$

$$\tag{1.64}$$

$$\mathbf{B}_q(\mathbf{r}) = + \frac{1}{2}W_0[\nabla\rho + \nabla\rho_q] \tag{1.65}$$

$$- \frac{1}{8}(t_1x_1 + t_2x_2)\mathbf{J} + \frac{1}{8}(t_1 - t_2)\mathbf{J}_q \tag{1.66}$$

Unless otherwise specified, unindexed densities denote isoscalar quantities (sum of neutron's and proton's).

1.3. Coulomb interaction

Unlike the Skyrme interaction, the Coulomb force is not local, giving rise to an unwanted integral operator in the Hamiltonian. A well known and widely used device is the Slater approximation [14], which gives a local exchange interaction.
In this approximation, the Coulomb energy reads

$$E_{\text{Coul}} = \int \mathcal{E}_{\text{Coul}}(\mathbf{r}) d\boldsymbol{r}$$

$$\mathcal{E}_{\text{Coul}}(\boldsymbol{r}) = \frac{e^2}{2}\left[\int \frac{\rho_p(\mathbf{r})\rho_p(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r}' - \frac{3}{2}\left(\frac{3}{\pi}\right)^{\frac{1}{3}} \rho_p^{4/3}(\mathbf{r})\right]$$

Which gives

$$U_C(\mathbf{r}) = \frac{\delta \mathcal{E}_{\text{Coul}}}{\delta \rho_p} = \frac{e^2}{2}\left[\int \frac{\rho_p(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} d^3\mathbf{r}' - 2\left(\frac{3}{\pi}\right)^{\frac{1}{3}} \rho_p^{1/3}(\mathbf{r})\right] \tag{1.67}$$

From a computational standpoint, the exchange part is trivial, while the direct one is more involved. One could compute the integral, but the complexity on a 3D mesh is $\mathcal{O}(N^6)$, rendering it unusable for fine meshes.
An alternative approach is to solve the poisson equation (from now on, V_c refers to the direct part only)

$$\nabla^2 V_c = 4\pi e^2 \rho_p \tag{1.68}$$

Given the proton density, we can impose Dirichlet boundary conditions, which can be extracted from a quadrupole expansion of the charge density [17]

$$V_c(\mathbf{r}) = 4\pi e^2 \sum_{\lambda=0}^{2} \sum_{\mu=-\lambda}^{\lambda} \frac{\langle Q_{\lambda\mu}\rangle\, Y_{\lambda\mu}}{r^{1+\lambda}} \text{ on } \partial\Omega \tag{1.69}$$

Where $\langle Q_{\lambda\mu}\rangle$ is defined as

$$\langle Q_{\lambda\mu}\rangle = \int r^\lambda Y^*_{\lambda\mu}(\mathbf{r}) \rho_p(\mathbf{r}) d^3\mathbf{r} \tag{1.70}$$

Since we expect a charge density confined to the nuclear shape, higher order terms in the expansion can be neglected, provided that the box is sufficiently large.
In a reference frame where the nucleus center of mass is at the origin, the expansion reduces to

$$V_c(\mathbf{r}) = \frac{Ze^2}{r} + e^2 \sum_{\mu=-2}^{2} \frac{\langle Q_{2\mu}\rangle\, Y_{2\mu}}{r^3} \text{ on } \partial\Omega \tag{1.71}$$

Refer to appendix A.1 for the definition and numerical evaluation of the spherical harmonics $Y_{\lambda\mu}$.

1.4. Energy calculation

One, if not the most important physical quantity we want to compute is the total energy of the system.

Integrated energy

The obvious way would be to evaluate the functional for a given density. We will call this *integrated energy*.

$$E_{\text{int}} = E[\rho, \tau, J_{\mu\nu}] = \int \mathcal{E} d\mathbf{r}$$

Hartree-Fock energy

An alternative approach can be used, as in a stationary point $\delta E = 0$, the single particle eigenvalue equation 1.50 stands true, summarized as

$$(\hat{t} + U)\varphi_k = \varepsilon_k \varphi_k \tag{1.72}$$

We can multiply 1.72 on the left by φ_k^* and integrate to get

$$\int -\varphi_k^* \frac{\hbar^2}{2m} \nabla^2 \varphi_k d\boldsymbol{r} + \int \varphi_k^* U \varphi_k d\boldsymbol{r} = \int \varphi_l^* \varepsilon_k \varphi_k d\boldsymbol{r} \tag{1.73}$$

The integral on the right hand side of 1.73 evaluates to ε_k due to the orthonormality constraint. If we sum over all states k we get

$$\sum_k \left\{ \int -\varphi_k^* \frac{\hbar^2}{2m} \nabla^2 \varphi_k d\boldsymbol{r} + \int \varphi_k^* U \varphi_k d\boldsymbol{r} \right\} = \sum_k \varepsilon_k \tag{1.74}$$

$$\sum_k t_k + \int \rho U = \sum_k \varepsilon_k \tag{1.75}$$

Since U is calculated as 1.52, assuming that the functional has a power dependence from ρ of the form $\mathcal{E}_{\text{Skyrme}} = A\rho^{\sigma+1}$ as in our case, we get the *rearrangement energy*

$$\rho U = \rho \frac{\delta \mathcal{E}_{\text{Skyrme}}}{\delta \rho} = \rho(\sigma+1)A\rho^{\sigma} = (\sigma+1)A\rho^{\sigma+1} = \mathcal{E}_{\text{Skyrme}} + \sigma \mathcal{E}_{\text{Skyrme}} = \mathcal{E}_{\text{Skyrme}} - \mathcal{E}_{\text{rea}} \tag{1.76}$$

If we explicit ρU in equation 1.75 using 1.76, we get to

$$\sum_k t_k + \int (\mathcal{E}_{\text{Skyrme}} - \mathcal{E}_{\text{rea}}) d\boldsymbol{r} = \sum_k \varepsilon_k$$

Isolating the Skyrme energy density

$$\int \mathcal{E}_{\text{Skyrme}} d\boldsymbol{r} = \sum_k (\varepsilon_k - t_k) + \int \mathcal{E}_{\text{rea}} d\boldsymbol{r} \tag{1.77}$$

and given the total energy of the system from 1.39

$$E = \sum_k t_k + \frac{1}{2} \int \mathcal{E}_{\text{Skyrme}} d\boldsymbol{r} \tag{1.78}$$

substituting 1.77 in 1.78 yields

$$E_{\text{HF}} = \frac{1}{2} \sum_k (\varepsilon_k + t_k) + \int \mathcal{E}_{\text{rea}} d\boldsymbol{r} = \frac{1}{2} \left(T + \sum_k \varepsilon_k \right) + E_{\text{rea}} \tag{1.79}$$

which will be called *Hartree-Fock energy* throughout this text.

Sidenote: The actual functional has a plethora of ρ terms, which can be summarized as

$$\mathcal{E}_{\text{Skyrme}} = \sum_j A_j \rho^{\sigma_j + 1} \implies E_{\text{rea}} = -\sum_j \sigma_j A_j \rho^{\sigma_j + 1}$$

This means that only terms with a $\sigma_j \neq 0, -1$ actually contribute to the rearrangement energy.

Since equation 1.79 is valid only for $\delta E = 0$, it's useful to check its equivalence with the integrated energy at convergence, so one can be sure to actually be in a stationary point.

2 | Numerical methods

This chapter is devoted to tackling the practical implementation of the nuclear Hartree-Fock method, with a focus on the discretized evaluation of the relevant PDEs, the approximate solution of the eigenvalue problem, and the solution of the self-consistent problem.

2.1. Finite differences

The framework used to numerically solve the relevant PDEs of the problem, is the one of finite differences. The core idea is to discretize the domain on a 3D mesh, use Taylor expansions to approximate differential operators and then solve the resulting system of linear equations.

2.1.1. 3D mesh

The first step of the process is representing the different fields in a numerical, discretized fashon. Generally speaking, we deal at most with 2-rank tensors, which vary in space and spin.
Discretizing the 3D cartesian space with a 3-index mesh, choosing a box which size on x, y, z is respectively $[-a_x, a_x]$, $[-a_y, a_y]$, $[-a_z, a_z]$, and a number of points N_x, N_y, N_z, the resulting lattice will be given by

$$V = \{(-a_x + ih_x, -a_y + ih_y, -a_z + ih_z)\} = \{(x_i, y_j, z_k)\}$$

Where the indices and step sizes are

$$\begin{aligned} i = 0, \dots, N_x - 1 \quad h_x = \frac{2a_x}{N_x - 1} \\ j = 0, \dots, N_y - 1 \quad h_y = \frac{2a_y}{N_y - 1} \\ k = 0, \dots, N_z - 1 \quad h_z = \frac{2a_z}{N_z - 1} \end{aligned}$$

For ease of notation, we will assume $a = a_x = a_y = a_z = a$ and $N = N_x = N_y = N_z = N$, without losing generality.
Including the spin degree of freedom, we can finally represent fields in a numerical way through

$$\varphi(\boldsymbol{r}, \sigma) \mapsto \varphi(x_i, y_j, z_k, s) = \varphi_{ijks} \tag{2.1}$$

Differential operators discretization

By using Taylor series, it's possible to write approximations to derivatives [39], in any point of the lattice, of any (reasonable) order of accuracy, involving only near neighbouring points. In the present work, 5-points derivates are used, meaning Taylor expansions are written for $\varphi(x \pm h)$ and $\varphi(x \pm 2h)$ to compute the differential operators. Formulae for first and second derivates are given in appendix A.2.
From the theory background of chapter 1, we discern two main kinds of PDEs, the Schrödinger like KS equation, and the Poisson equation. Both are similar to a diffusion equation, but the former involves the spin degree of freedom, while the latter requires careful treatment of the boundary conditions.

2.1.2. Schrödinger equation

Starting from the Schrödinger equation 1.50, reported here for clarity

$$\left[-\nabla\left(\frac{\hbar^2}{2m_q^*(\mathbf{r})}\nabla\right) + U_q(\mathbf{r}) + \delta_{\mathrm{q,proton}} U_C(\mathbf{r}) - i\mathbf{B}_q(\mathbf{r}) \cdot (\nabla \times \boldsymbol{\sigma}) \right] \varphi = \varepsilon\varphi$$

it can be compactly written as

$$f(\nabla^2\varphi, \nabla\varphi, \varphi, \boldsymbol{r}, s) = \varepsilon\varphi. \tag{2.2}$$

If f is linear in φ, it would be possible to rewrite it as a linear combination of φ on the mesh, after which we can use linear algebra methods to solve the problem.

Linearity Breaking down each part of the equation, the kinetic term

$$\nabla\left(\frac{\hbar^2}{2m_q^*(\mathbf{r})}\nabla\right)\varphi = \frac{\hbar^2}{2m_q^*(\mathbf{r})}\nabla^2\varphi + \nabla\left(\frac{\hbar^2}{2m_q^*(\mathbf{r})}\right) \cdot \nabla\varphi \tag{2.3}$$

Is evidently linear in φ.
The spin-orbit coupling, which most generally reads

$$\begin{aligned}\hat{h}_{\mathrm{SO}} &= \boldsymbol{f}(\boldsymbol{r}) \cdot (\nabla \times \boldsymbol{\sigma}) \\ &= f_x(\boldsymbol{r})(\sigma_z \partial_y - \sigma_y \partial_z) + f_y(\boldsymbol{r})(\sigma_x \partial_z - \sigma_z \partial_x) + f_z(\boldsymbol{r})(\sigma_y \partial_x - \sigma_x \partial_y)\end{aligned}$$

where $\hat{h}_{\mathrm{SO}}$ acts on the spinor φ_{ijk}. Since $\boldsymbol{\sigma}$ is the vector of Pauli matrices acting as linear operators on φ_{ijk}, this portion of the equation is linear in φ.
Finally, the mean field terms U_q, U_c

$$(U_q + \delta_{\mathrm{q,proton}} U_c)\varphi$$

are just multiplicative, hence linear.
Given that the whole equation is linear in φ, we can evaluate it on the chosen mesh, using finite differences to approximate the differential operators, yielding a linear eigenvalue problem of the form

$$\sum_{n}^{N_x \cdot N_y \cdot N_z \cdot 2} A_{mn}\varphi_n = E\varphi_m \tag{2.4}$$

Boundary conditions

We expect the nucleus to be a localized object, prompting null Dirichlet boundary conditions for the Schrödinger equation. Near the boundaries, the derivatives will involve points outside the box, setting these points to zero, is equivalent to solving

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & A & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \varphi_{-2} \\ \varphi_{-1} \\ \varphi \\ \varphi_N \\ \varphi_{N+1} \end{bmatrix} = E \begin{bmatrix} \varphi_{-2} \\ \varphi_{-1} \\ \varphi \\ \varphi_N \\ \varphi_{N+1} \end{bmatrix} \tag{2.5}$$

From this system of equations, we get for points outside the boundary:

$$\begin{cases} \varphi_{-2} = 0 \\ \varphi_{-1} = 0 \\ \dots \end{cases} \tag{2.6}$$

Meaning that φ outside the box will automatically be set to zero if the A matrix is built assuming those points to be zero when computing its coefficients.

2.1.3. Poisson equation

The other fundamental PDE we need to solve is the Poisson equation encountered in section (REF). Dropping the c and p subscripts, it reads

$$\nabla^2 V = 4\pi e^2 \rho$$

It's simpler than the Schrödinger equation, as it only involves a laplacian and it's not an eigenvalue problem. The right is side is given, and the solution is found by inverting the coefficients' matrix.

Boundary conditions

Unlike the Schrödinger equation, we do not expect the solution to rapidly decay near the boundaries; as reported in section 1.3, we have fixed, non-null boundary conditions, which we have to properly impose on the system.
We can choose a direction, say x, and look at the discretized equation at the boundaries $x = \pm a$. Since the indeces j, k won't vary, we can omit them, and ignore the other derivates in the following equations.

$$\begin{aligned} \nabla^2 V &= \partial_{xx} V + \partial_{yy} V + \partial_{zz} V \\ &= \frac{-V_{i-2} + 16V_{i-1} - 30V_i + 16V_{i+1} - V_{i+2}}{12h^2} + \cdots = 4\pi e^2 \rho_i \end{aligned} \tag{2.7}$$

Near a boundary, say $i = 0$, the formula calls for points outside the box, known as *ghost points*. Since they are not part of the linear system, but they are known, we can bring them on the right side of equation 2.7.

$$\frac{-30V_0 + 16V_1 - V_2}{12h^2} = 4\pi e^2 \rho_0 + \frac{V_{-2} - 16V_{-1}}{12h^2} = \tilde{\rho}_0 \tag{2.8}$$

The same procedure must be applied to all equations involving ghost points, e.g. for $i = 1$

$$\frac{+16V_0 - 30V_1 + 16V_2 - V_3}{12h^2} = 4\pi e^2 \rho_1 + \frac{V_{-1}}{12h^2} = \tilde{\rho}_1. \tag{2.9}$$

The proper system to solve will then be

$$AV = \tilde{\rho} \tag{2.10}$$

Where A is constructed as previously specified. Solving with $\tilde{\rho}$ on the right hand side will force the solution to abide boundary conditions.

On higher order approximations and performance

Higher and higher order approximations for derivatives involve more points that are further away. This increases accuracy by reducing the finite differences error, but it also decreases matrix sparseness.
Algorithms like Conjugate Gradient, as we'll see in the next section, and linear algebra computations, benefit from matrix sparseness, since multiplication by zero is readily known. The implication is that performance *vs* accuracy is a tradeoff that isn't univocal to every problem.
In the present work, the golden choice has been 5-point derivatives; but it's not definitive, as the lattice points rapidly cap depending on the system's memory.
As an example, take a seemingly harmless grid, made of 50 points in each direction. The resulting matrix will be $50 \times 50 \times 50 \times 2 = 2.5 \times 10^5$ both in columns and rows. It may be the case in the future, that higher order derivatives will be needed, to compensate for the limitation brought by the $\mathcal{O}(h^n)$ polynomial accuracy of the method in the step size.

2.2. Eigenvalue problem

This section is devoted to the approximate solution of the eigenvalue problem, by exploring the idea behind the Conjugate Gradient (CG) algorithm and its use in the General Conjugate Gradient (GCG).
Eigenvalue problems are ubiquitous in physics and engineering, and while solving one for a small matrix is trivial, it still requires roughly $O(n^3)$ [13] operations to do so. More often than not, real computational applications result in large-scale matrices, which are completely out of question for exact eigenvalues calculations, thus requiring the use of approximate algorithms.
We'll start by skimming through the basics of the most modern and used algorithms, by understanding shared numerical techniques and addressing their limitations, and then move on to the GCG.

2.2.1. Conjugate Gradient and numerical techniques

Iterative eigensolvers often share some common numerical techniques, like the Rayleigh-Ritz procedure, shift-and-invert, and approximate linear systems solvers. We will begin by describing these recurring techniques.

Conjugate Gradient and numerical techniques

Solving linear systems of the form

$$Ax = b \tag{2.11}$$

is crucial in many eigensolvers. The Conjugate Gradient (CG) is perhaps the most famous iterative solver in this sense, especially in regards to sparse matrices, as we'll see in a moment. CG applies to cases where A is an $n \times n$, positive-definite, symmetric matrix, and x and b are n-dimensional vectors.
Many generalizations to this method exist, which relax the requirements on the matrix, like BiCGSTAB, CGRES and so on [30]. We'll describe the working principle of CG, but the same applies to all the others, with slight variations.

Steepest descent method The quadratic form $f(x)$ derived from system 2.11 is

$$f(x) = \frac{1}{2}x^T Ax - b^T x \tag{2.12}$$

If A is symmetric, positive-definite, the shape of $f(x)$ is convex and has a global minimum for

$$\nabla_x f(x) = Ax_m - b = 0 \implies Ax_m = b \tag{2.13}$$

This implies that the extremum of the quadratic form is the also the solution of the linear system 2.11.
We can employ the well-known gradient descent technique [31] to find such point: starting from a guess x_0, we compute the direction where f decreases the most (the residual r_i), compute the step size for maximal decrease, and update x_i at each iteration accordingly, repeating until convergence.

$$d_i = r_i = = b - Ax_i \tag{2.14}$$

$$x_{i+1} = x_i + \alpha_i r_i \tag{2.15}$$

$$\text{with } \alpha_i \text{ such that } \frac{\mathrm{d}f}{\mathrm{d}\alpha_i} = 0 \implies \alpha_i = \frac{r_i^T r_i}{r_i^T A r_i} \tag{2.16}$$

This is a powerful but highly inefficient procedure. We are not ensuring that the search direction doesn't end up with components in subspaces that were explored already.
It can be proven [31] that the norm of the error $e_i = x_i - x_m$ is minimal at each iteration if the search directions d_i are chosen to be A-orthogonal to the next error, i.e. $d_i^T A e_{i+1} = 0$. This makes the algorithm converge at the exact solution in n steps, but most importantly it allows to truncate the iterations without a large error on the approximation x_i.

In this case, the algorithm is called Conjugate Gradient Method and is formulated as

$$\alpha_i = \frac{r_i^T r_i}{d_i^T A d_i} \tag{2.17}$$

$$x_{i+1} = x_i + \alpha_i d_i \tag{2.18}$$

$$r_{i+1} = r_i - \alpha_i A d_i \tag{2.19}$$

$$\beta_{i+1} = \frac{r_{i+1}^T r_{i+1}}{r_i^T r_i} \tag{2.20}$$

$$d_{i+1} = r_{i+1} + \beta_{i+1} d_i \tag{2.21}$$

Where iterations are truncated if the norm of the residual r_i is smaller than a certain threshold. CG represents a great method for sparse matrices, because it can be proven to be of complexity $O(m)$, where m is the number of non-zero elements in A [31].

Preconditioning

The CG method convergence is known to be limited by the modulus of the condition number of A [31], given by

$$\kappa(A) = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}. \tag{2.22}$$

If we were able to find a good *preconditioner* M, symmetric and positive-definite, such that $\kappa(M^{-1}A) \lll \kappa(A)$, and M^{-1} is easy to compute, then the algorithm would converge much faster, by solving $M^{-1}Ax = M^{-1}b$, since x is also the solution of $Ax = b$.

$$x = (M^{-1}A)^{-1}M^{-1}b = A^{-1}MM^{-1}b = A^{-1}b \tag{2.23}$$

Without delving the details of the preconditioner implementation, detailed in [31], note that, in general, $M^{-1}A$ is neither positive-definite nor symmetric, which requires a decomposition of the type $M = EE^T$ to be used.
The catch with preconditioning is that M has no unique recipe. Preconditioners are widely spread across numerical analysis, so many methods have been explored and implemented [25].

Rayleigh-Ritz procedure

A common denominator of all these algorithms is the search of good approximations for the correct eigenvectors in a certain subspace $\mathcal{K}$. The method is called Rayleigh-Ritz (RR) procedure [30], and is here outlined.
Suppose to have a matrix A of size $n \times n$, with entries in $\mathbf{C}$ and a collection of vectors

k which form a subspace $\mathcal{K} \subset \mathbf{C}^n$, where $\mathcal{K}$ is an $n \times k$ matrix. Generally speaking, n is large, while k is much smaller.
The best approximation of the true eigenvectors of A in $\mathcal{K}$ can be computed by solving the small scale eigenvalue problem

$$\mathcal{K}^\dagger A \mathcal{K} C = C\Lambda \tag{2.24}$$

Resulting in matrices $\mathcal{K}^\dagger A\mathcal{K}$ and C being of size $k \times k$. Computing $\mathcal{K}C$ gives a matrix of size $n \times k$, whose column vectors are the best approximations of the eigenvectors of A in $\mathcal{K}$, with their corresponding eigenvalues in the entries of the diagonal matrix Λ.

Shift and Invert

The power iteration is the technique on which Krylov subspace search methods are based [13]. By repeatedly applying matrix A to a vector x, it gets skewed towards the eigenvector whose eigenvalue is of largest magnitude λ_n.
Assume A is a hermitian matrix, thus diagonalizable. This means we can write an arbitrary vector $x^{(0)}$ as a linear combination of the eigenvectors $\{v_i\}$ of A.

$$x^{(0)} = \sum_i^n \alpha_i v_i \tag{2.25}$$

If we apply A to $x^{(0)}$ k times, we get

$$x^{(k)} = A^k x^{(0)} = \sum_i^n \alpha_i A^k v_i = \sum_i^n \alpha_i \lambda_i^k v_i \tag{2.26}$$

It can be proven that taking the ratio of the j-th component of $x_j^{(k)}$ and $x_j^{(k-1)}$ converges to λ_n

$$\lim_{k\to\infty} \frac{x_j^{(k)}}{x_j^{(k-1)}} = \lambda_n \tag{2.27}$$

which means, that for large enough k, we have the relation

$$Ax^{(k)} \approx \lambda_n x^{(k)} \tag{2.28}$$

So $x^{(k)}$ is an approximation of the eigenvector v_n of A whose eigenvalue is λ_n.

Smallest eigenvalue If instead of the largest eigenvalue, we were intereseted in the smallest one (in magnitude) λ_0, then we would need to apply the inverse matrix A^{-1} to

$x^{(k)}$, which would change the ratio 2.27 to

$$\lim_{k\to\infty} \frac{x_j^{(k)}}{x_j^{(k-1)}} = \lambda_0 \tag{2.29}$$

Assume for a moment that we're solving a nuclear single-particle hamiltonian, where we have a certain number of bound states of negative energy and a much larger number of unbound states with positive energy. In this case, the inverse power iteration would converge to the states which energy is closer to zero, avoiding the interesting ones on the bottom of the spectrum.
The solution is, before inverting, to shift the matrix by a quantity σ that is very close to the lowest eigenvalue we want to compute, call it λ_σ (eigenvector v_σ). Now, the eigenvalue of lowest magnitude of $(A - \sigma I)$ is $\lambda_\sigma - \sigma$ and by applying $(A - \sigma I)^{-1}$ to $x^{(k)}$, we'll get the approximation to the eigenvector v_σ.

2.2.2. Iterative eigensolvers

Now that the main techniques used by iterative eigensolvers have been laid out, we can look at three general methods, which are the most commonly used ones.

Jacobi-Davidson

The Jacobi-Davidson method [33] performs the RR procedure on a subspace which is enriched at each iteration by a correction to the previous eigenvectors.
Given an approximation (u, θ) of an eigenpair of matrix A, if the residual

$$r = Au - \theta u \tag{2.30}$$

is ≈ 0, then the eigenpair converged. Otherwise, we want to fine a correction t such that

$$r = A(u + t) - (\theta + \delta\theta)(u + t) = 0 \tag{2.31}$$

Linearizing this equation in t gives

$$(A - \theta I)t = -r \tag{2.32}$$

To avoid singularity of the equation near convergence, since u approximately spans a subspace of $\ker(A-\theta I)$, and enrich the subspace search with a useful orthogonal correction,

we project the problem onto the orthogonal subspace of u, which finally gives

$$(I - uu^\dagger)(A - \theta I)(I - uu^\dagger)t = -r \tag{2.33}$$

Algorithm 2.1 Jacobi-Davidson method for $Ax = \lambda x$

1: Choose normalized initial vectors $\{u_k\}$, set $V = [u_1, \ldots, u_k]$
2: **repeat**
3: Compute Ritz pair: $T = V^\dagger AV$, solve $Ty = \theta y$
4: Set $u = Vy$, residual $r = Au - \theta u$
5: **if** $\|r_k\| < \varepsilon \ \forall k$ **then**
6: **return** (θ, u)
7: **end if**
8: Solve approximately $(I - u_k u_k^\dagger)(A - \theta I)(I - u_k u_k^\dagger)t_k = -r_k$ using preconditioned iterative solver, ensuring $t_k \perp u_k$
9: Normalize: $v_k = t_k / \|t_k\|$
10: Expand subspace, setting $V = [V, v]$
11: **until** convergence for $k = 1, \ldots,$ nev

Although simple, this method is computationally efficient only by using preconditioning, which is known to be unstable in many cases [30].

Lanczos

Lanczos algorithm [19] is probably the most used iterative eigensolver in regards to hermitian matrices. It's a Krylov subspace search method, meaning the Rayleigh-Ritz procedure is done on a subspace formed as

$$\mathcal{K} = \{v_1, Av_1, A^2v_1, \ldots, A^{k-1}v_1\} \tag{2.34}$$

which exploits the power iteration. After orthogonalizing the new approximation to the previous one and diagonalizing the small scale problem, we end up with the new best approximations to the eigenvectors of A.

Algorithm 2.2 Lanczos Method for Computing nev Lowest Eigenpairs of Hermitian A

Choose normalized initial vector v_1, set $\beta_0 = 0$, $m =$ subspace size.
repeat
  for $j = 1, 2, \ldots, m$ **do**
    $w \leftarrow Av_j - \beta_{j-1}v_{j-1}$
    $\alpha_j \leftarrow v_j^* w$
    $w \leftarrow w - \alpha_j v_j$
    $\beta_j \leftarrow \|w\|$
    if $\beta_j = 0$ **then**
      break
    end if
    $v_{j+1} \leftarrow w/\beta_j$
  end for
  Form tridiagonal matrix $T_m = \text{tridiag}(\beta_{1:m-1}, \alpha_{1:m}, \beta_{1:m-1})$
  Compute eigen-decomposition $T_m y_k = \theta_k y_k$, for $k = 1, \ldots, \text{nev}$
  Form Ritz approximations $x_k = V_m y_k$, where $V_m = [v_1, \ldots, v_m]$
  Compute residual norms $r_k = \|Ax_k - \theta_k x_k\|$ for all k
until convergence for $k = 1, \ldots, \text{nev}$

Lanczos is extremely efficient, memory and CPU wise for extremal eigenvalues, but this limits its applicability to discretized unbound operators, where unwanted eigenvalues may dominate the Krylov subspace, and most importantly, for applications where the inner part of the spectrum is of interest, as in the case of Hartree-Fock-Bogoliubov (HFB).
A shift-and-invert strategy would be unfeasible in the case of large scale problems, since all Lanczos steps need to be performed exactly to avoid instabilities, a well known problem in the Arnoldi generalization [30].

LOBPCG

The last algorithm of this short list is LOBPCG, it's the newest and most sofisticated one of the three.
Introduced by A. V. Knyazev in 1991 [18], it's a block, preconditioned conjugate gradient method, explicitly targeted at solving large-scale eigenvalue problems, and it's been used in modern solutions of the Schrödinger/KS equation in recent years [20, 22, 24, 38]. In this algorithm, the Rayleigh-Ritz procedure is done on a subspace formed as

$$V = [X, W, P] \tag{2.35}$$

Where X is the current best eigenvectors approximation, P is the block of previous search directions and W is a block formed by preconditioning P.
We won't go into the details of LOBPCG, since GCG shares with it many aspects, like blocking and search directions calculation.
LOBPCG works very well for large scale problems, but it has limitations. For one, it's not possible arbitrarily select the portion of the matrix spectrum to calculate, which is required for problems where variational collapse happens, like in HFB or the Dirac equation, which manifests particle/antiparticle solutions [20]. To solve this, an additional filtering step is required [20, 22], which introduces a computational cost in the algorithm. Lastly, LOBPCG may fail when poor conditioning is present or when high precision on the eigenvalues is required [21].

2.2.3. General Conjugate Gradient

The General Conjugate Gradient is an iterative eigensolver designed with the aim of improving LOBPCG, it's a blocked algorithm, which uses the inverse power method and previous search directions to generate the search subspace. GCG is proven to be faster and more stable than LOBPCG [21].
A slightly different implementation of the algorithm is employed in the present work, detailed in algorithm 2.3, to improve applicability to HF calculations and reduce the computational cost.

Eigenvalue problem The original algorithm aims at solving the general eigenvalue problem $AX = \lambda BX$. Since in our case $B = I$, it is omitted from the algorithm, reducing the computational cost of P generation, and orthogonalization of V.

Blocking The algorithm allows to save converged eigenpairs (implemented) and work on a subset of the active eigenvectors. Since in a self-consistent calculation the matrix changes rapidly and at each HF iteration, it will be the case that the maximum number of iterations is reached before convergence, so we must work at all times on the remaining unconverged eigenvectors.

Orthogonalization The original paper [21] suggests an improved orthogonalization procedure; being beyond the scope of this work, the simpler Gram-Schmidt [4] orthogonalization is used in the present work.

Preconditioning The use of a preconditioner is beyond the scope of this work, a simple diagonal preconditioner is used.

Shift update The shift update is either fixed, in case of known spectrum, e.g. for HFB `shift` $= 0$, or adaptive [21], so that the inverse power step can find the correct eigenvalues, using the update formula 2.36

$$\texttt{shift} = (\lambda_{\text{nev}} - 100\lambda_1)/99 \tag{2.36}$$

Where λ_{nev} is the largest eigenvalue of the RR procedure and λ_1 is the smallest.

Convergence Convergence on an eigenpair is checked by computing the norm of the corresponding column vector of the residual matrix R and comparing it against a threshold.

Algorithm 2.3 GCG Algorithm

1: **Input:** Matrix A, number of desired eigenpairs `nev`, X_{guess} initial guess with $\text{col}(X_{\text{guess}}) = k \geq \text{nev}$, `max_iter` maximum number of iterations.
2: Initialize block $X = [X_{\text{c}}, X_{\text{a}}] \leftarrow [X_{\text{guess}}]$
3: Initialize blocks P and W with k null vectors
4: Solve the Rayleigh Ritz problem $X^{\dagger}AXC = C\Lambda$
5: Update $X = XC$
6: Initialize `shift`, Initialize `iter` $= 0$
7: **while** $\text{col}(X_c) <$ `nev` and `iter` $<$ `max_iter` **do**
8: Solve approximately $(A + \texttt{shift} \cdot I)W = X\Lambda$ with some CG steps, initial value X to generate W
9: Orthogonalize $V = [X, P, W]$
10: Solve the Rayleigh Ritz problem $V^{\dagger}(A + \texttt{shift} \cdot I)VC = C\Lambda$
11: Update $X_{\text{new}} \leftarrow VC$ and $\Lambda_{\text{new}} = \Lambda - \texttt{shift} \cdot I$
12: Compute the residual $R = AX_{\text{new}} - \Lambda X$
13: Check convergence on k-th column of R, update X_c and X_a accordingly
14: Update `shift` and `iter`
15: **end while**
16: **Output:** Approximate eigenpairs (Λ, X)

2.3. Code implementation details

In this last section regarding numerical methods, a few important generalities about the actual code implementation of the HF method are discussed. Mainly, the implementation of the Augmented Lagrangian Method to enforce spatial constraints on the HF

solution, the pseudocode of the entire self-consistent procedure, and the choice of optimal parameters for the functional minimization.

2.3.1. Constraints

The HF ground state is a great starting point to get theoretical results regarding many nuclear properties; however, for a plethora of applications, calculating the state of the nuclear system under certain constraints becomes necessary.
Effectively, a need to explore the energy surface arises. This means minimizing the energy functional, under a series of constraints of the type $\langle \mathcal{Q} \rangle = q_0$, where q_0 is the desired expectation value of an operator $\mathcal{Q}$.
This is an equality-constrained optimization problem (ECP), formulated as

$$\min_{|\Psi\rangle} E \tag{2.37}$$

$$\text{constrainted to } \langle \Psi | \mathcal{Q} | \Psi \rangle = \langle \mathcal{Q} \rangle = q_0 \tag{2.38}$$

Which yields the Lagrangian

$$E' = E + \lambda(\langle \mathcal{Q} \rangle - q_0) \tag{2.39}$$

where λ is a Lagrange multiplier determined by the condition $\langle \mathcal{Q} \rangle = q_0$. After finding the minimum of E', it's trivial to show that for a given λ, we get [12]

$$\frac{\mathrm{d}E}{\mathrm{d}\langle \mathcal{Q} \rangle} = -\lambda. \tag{2.40}$$

From a numerical standpoint, λ needs to be tuned at each iteration to reach the desired value of q_0. This method was the one used in early constrained Hartree-Fock calculations [10].
Although this method is simple, it's often the case that it fails. Moreover, for the same value of λ, many, possibly infinite values of $\langle \mathcal{Q} \rangle$ can be obtained, for which we are only allowed to get the one with the most stable solution.
A different method is provided by the Quadratic Penalty Method (QPM). Briefly speaking, instead of a Lagrange multiplier, we add a quadratic contribution to the functional, such that

$$E' = E + \frac{c}{2}(\langle \mathcal{Q} \rangle - q_0)^2. \tag{2.41}$$

This is a straightforward method; intuitively one penalizes (hence the name), any solution for which $\langle \mathcal{Q} \rangle \neq q_0$ by increasing its energy. However, the success of such procedure is heavily influenced by the choice of c, often leading to instabilities for large values.

What happens is that for small values of c, the penalty may be insufficient to reach the target q_0, while for bigger values, the penalty may be so strong that the self consistent calculation oscillates and fails.

Augmented Lagrangian Method

A modern, robust approach, used by HF/HFB codes [7, 29] is given in the form of the Augmented Lagrangian Method (ALM) [34]. Its main idea is to combine the precision of the ECP with the accuracy of the QPM.
Without delving into cumbersome mathematical details, we'll see how the algorithm is practically implemented in the code.
Given the functional

$$E' = E + \lambda(q - q_0) + \frac{c}{2}(q - q_0)^2 \tag{2.42}$$

where $q = \langle \mathcal{Q} \rangle$, the resulting mean field potential will be given by

$$U' = U + \lambda\mathcal{Q} + c(q - q_0)\mathcal{Q} \tag{2.43}$$

$$= U + c(q - q_0(\lambda))\mathcal{Q} \tag{2.44}$$

where $q_0(\lambda)$ is updated at each iteration with the formula

$$q_0(\lambda) = q_0 - \frac{\lambda}{c} \tag{2.45}$$

$$\lambda^{(i+1)} = \lambda^{(i)} + \mu c(q - q_0) \tag{2.46}$$

Here, a slight deviation from the original ALM is present. Since the original work [34] doesn't provide guidance regarding what is considered an *iteration*, we employ the strategy [7] of using a damping factor $\mu \in [0, 1]$, so λ can be updated at each HF iteration for fast convergence, without large oscillations or instabilities of any kind.
This method is what powers the deformation curves that are shown in section (REF), allowing to explore the energy surface with arbitrary precision in reaching the value of q_0 at convergence, provided that enough HF iterations are performed.
Note that, since $\lambda^{(0)} = 0$, for $\mu = 0$ ALM reduces to the standard QPM.

2.3.2. Code implementation

The whole Hartree-Fock framework presented up to this point, has been implemented using the C++ language [36] and the Eigen linear algebra library [15], which implements linear algebra operatrions through low level routines such as `LAPACK` and `BLAS`. In figure

2.1, the schematics of the program structure is reported.

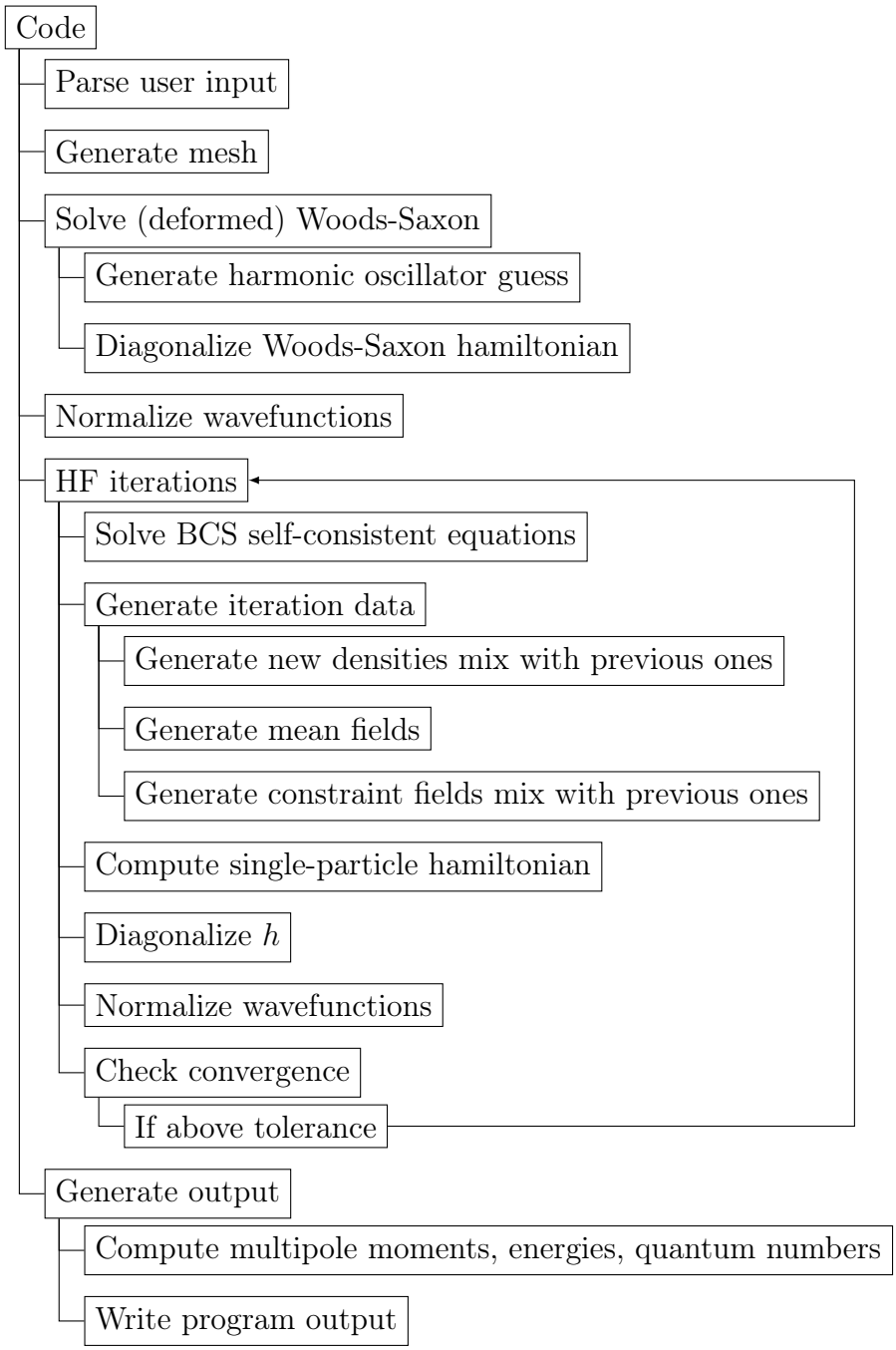

Figure 2.1: Pseudocode of the Hartree-Fock program.

2.3.3. Optimal parameters choice

For a given set of tolerances of the HF convergence criterion, the paramters that need to be chosen carefully are essentially the inverse power step tolerance and the number of

maximum GCG iterations.

Inverse power step tolerance

The inverse power step tolerance is one of the biggest bottlenecks of the computational cost of the solver. A careful choice is needed for correct eigenvalues convergence, while mainting the computational cost at bay.
From figure 2.2, it's clear that at least a tolerance of 10^{-3} is needed for good convergence, while tolerances $\geq 10^{-4}$ stop offering increasing returns, rendering a choice between 10^{-4} and 10^{-5} an optimal one.

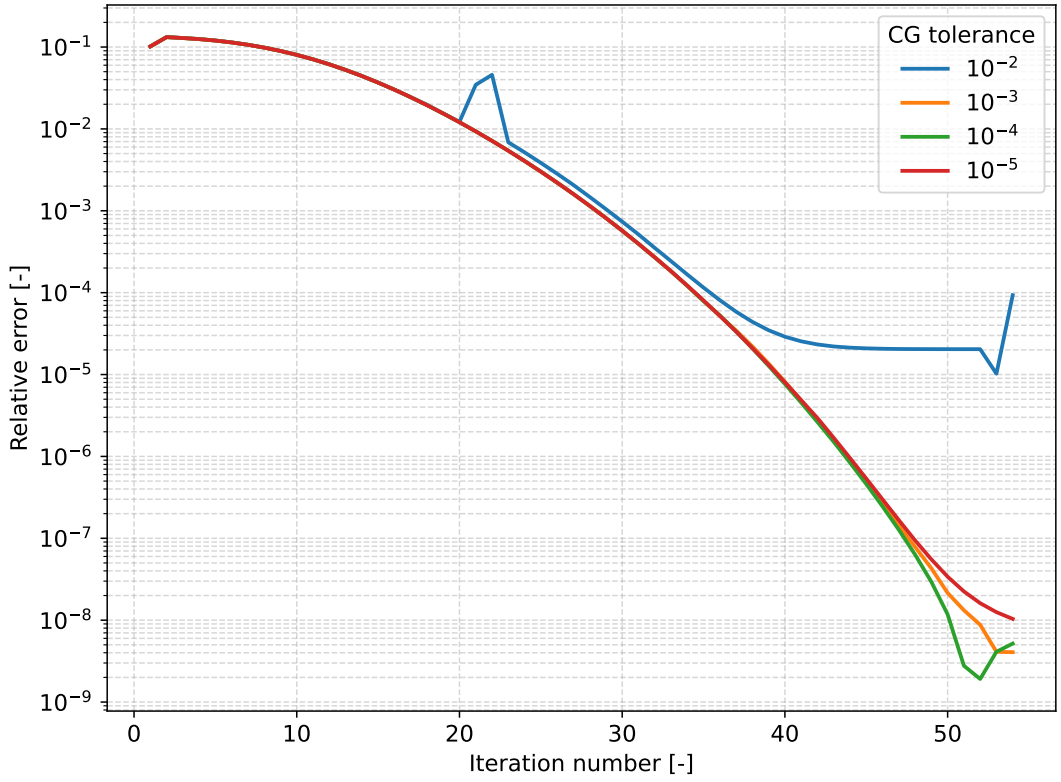

Figure 2.2: HF calculation convergence with varying CG tolerance for ^{16}O, box $[-9, 9]$ fm, step size 0.3 fm.

Inner GCG iterations

The number of inner GCG maximum iterations, here named 'inverse power steps' to avoid confusion, is slightly more nuianced than the CG tolerance. One could think that a higher number of steps would bring to convergence faster, since the precision on the eigenalues increases, but this is not the case.

In figures 2.3 and 2.4, the convergence of the HF calculation is plotted for different number of steps, respectively, for the spherical nucleus ^{16}O and the deformed nucleus ^{24}Mg.

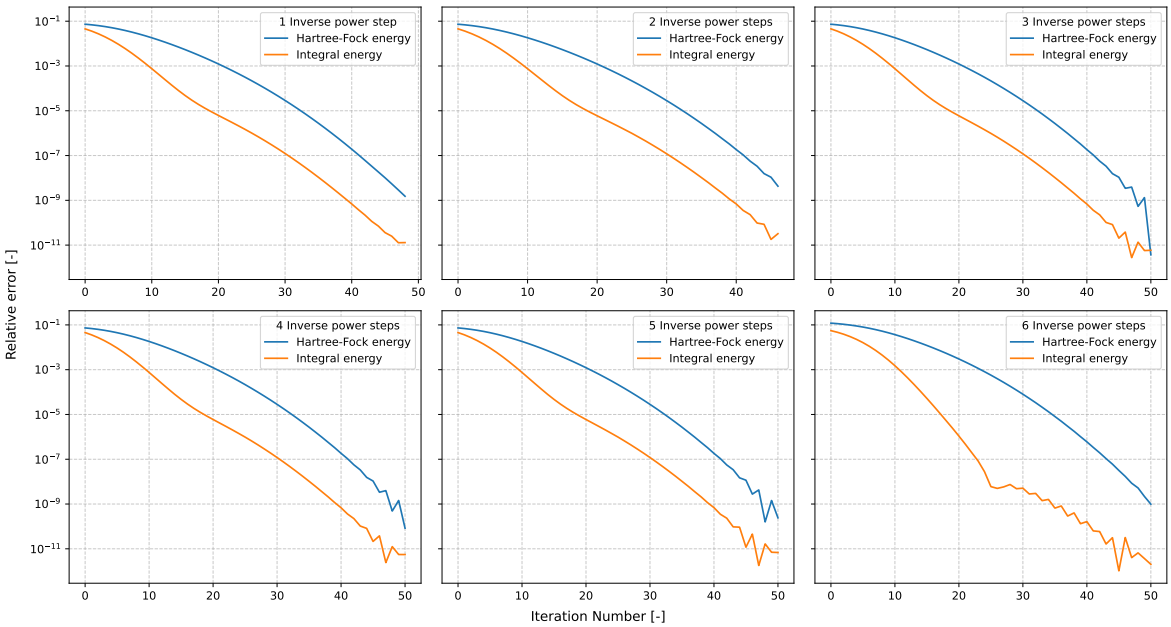

Figure 2.3: HF calculation convergence with varying number of inverse power steps for ^{16}O, box $[-9, 9]$ fm, step size 0.3 fm.

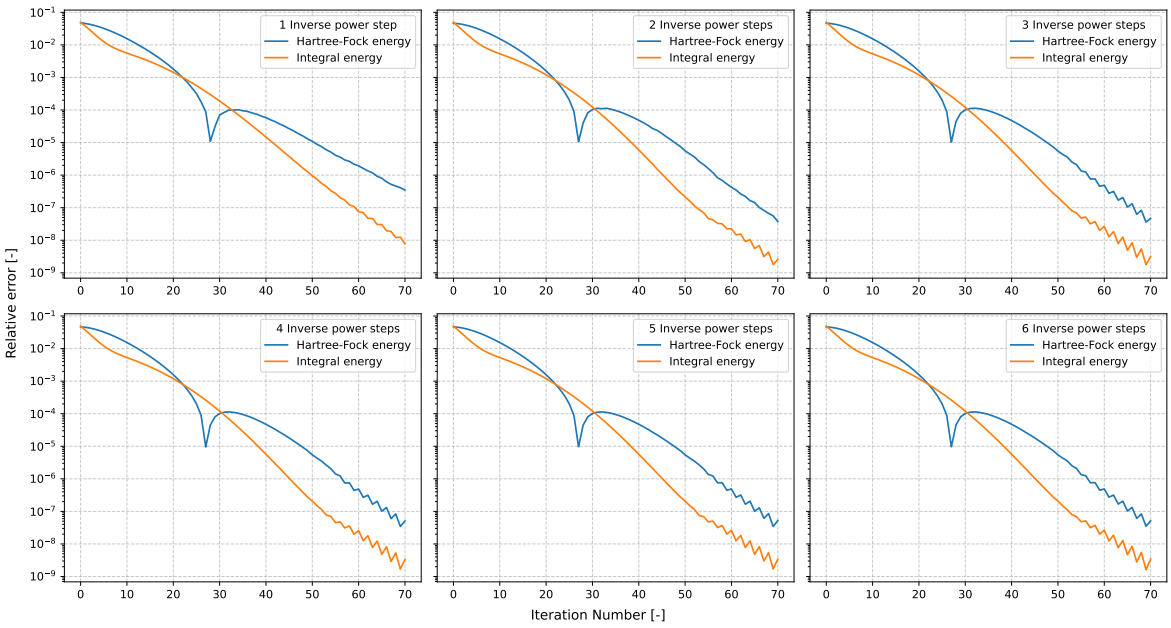

Figure 2.4: HF calculation convergence with varying number of inverse power steps for the deformed nucleus ^{24}Mg, box $[-10, 10]$ fm, step size 0.33 fm.

It's evident that in both cases, a steps number greater than 3 leads to an unstable convergence, while in the case of the spherical nucleus, just one step is enough to quickly, and reliably reach convergence. In any case, it's clear that delaying the inverse power steps to later HF iterations is safer in terms of stability.
This counter intuitive behavior is likely due to the fact that at each HF iteration the hamiltonian changes and a great number of steps leads to solutions too biased towards the current matrix eigenvectors, at the expense of the next iteration; however, in the case of deformed nuclei, due to sharp shape changes at the start of the calculation, just one step may not be enough to sustain the pace at which the Hamiltonian changes, hence the quicker convergence with more steps.

2.3.4. Numerical stability

As a final remark, the numerical stability of the solver is reported in figure 2.5. The map is produced for a spherical calculation of ^{16}O, with varying box and mesh sizes.
It's possible to observe that for a box whose side is at least ≈ 2.5 times the nuclear radius, the solver numerical stability is loosely dependent on the box extension, but rather on the step size. This is not surprising, as the points separation in space h dictates the precision of the discretized derivatives, as mentioned in section 2.1.

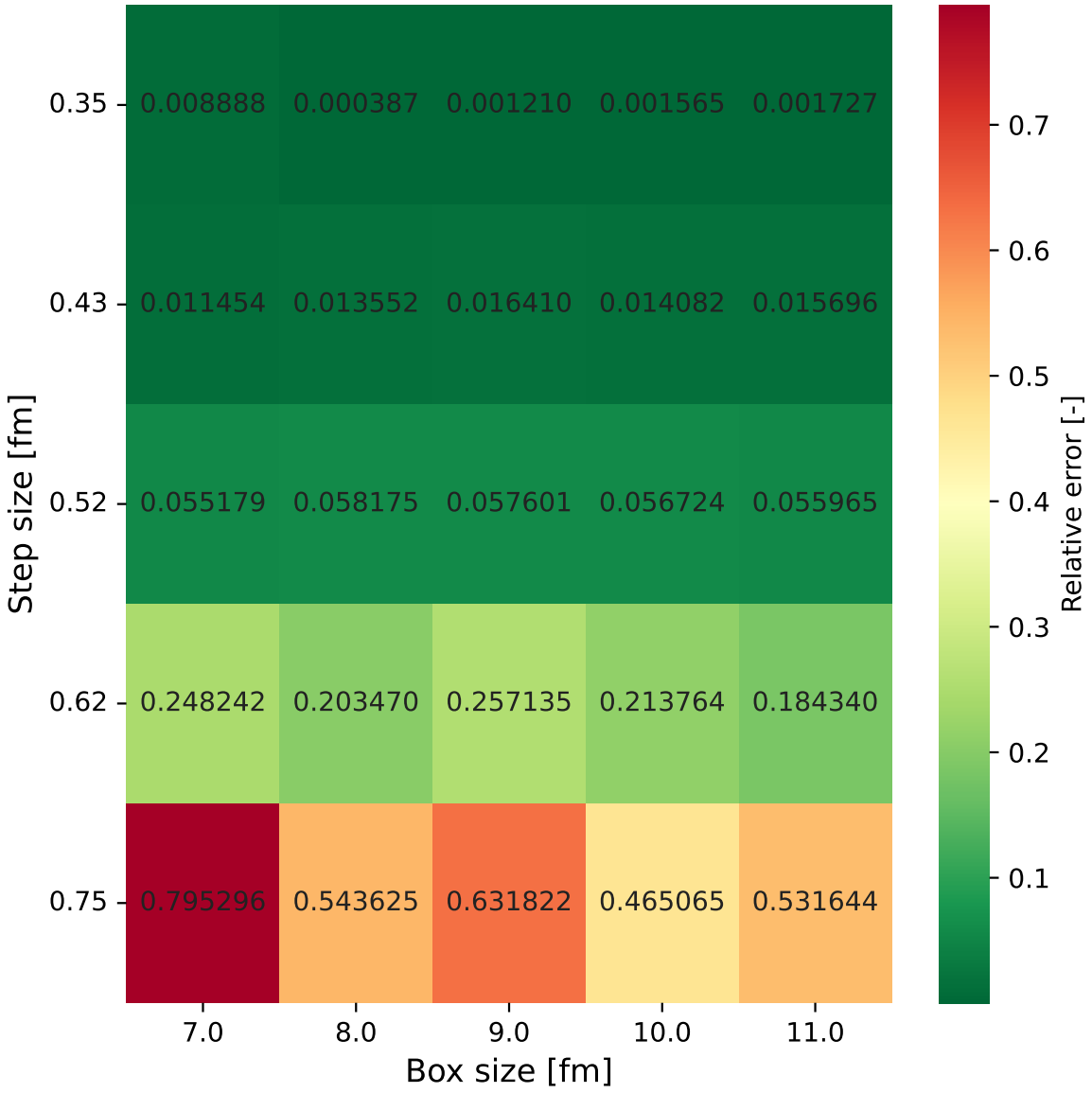

Figure 2.5: Numerical stability map of the HF solver for ^{16}O. Relative error is taken as the

3 | Results for spherical nuclei

In this chapter, results for spherical nuclei are presented. These are mostly calculated for double magic nuclei, with the exception of ^{90}Zr. The reason behind choosing spherical nuclei as an initial benchmark, is that numerous spherical HF codes are available and they have the advantage of being one-dimensional, which allows the use of very fine meshes, with a step size that can go down to the physical scale of the problem, which is roughly 0.1 fm in this case, without any major computational limit. We can use these codes as a reference ideal value for the different quantities produced by our code. The choice for benchmarking spherical results in the present work has been the `hfbcs_qrpa` code [8].

3.1. Physical quantities

After finding a nuclide's ground state, we are able to compute different physical properties of the system. We can use these values as a numerical reference when comparing our results with other codes.

3.1.1. Mean square radii

An important set of quantities characterizing the nuclear density is certainly the one of mean square radii. The individual nuclear spieces' mean square radius is defined as

$$\left\langle r_q^2 \right\rangle = \frac{\int \rho_q(\boldsymbol{r}) r^2 d\boldsymbol{r}}{\int \rho_q(\boldsymbol{r}) d\boldsymbol{r}}. \tag{3.1}$$

While the charge mean square radius formula is derived from the convolution of the neutron and proton particle densities with their respective internal charge distribution [3], resulting in equation 3.2.

$$\left\langle r_{ch}^2 \right\rangle = \left\langle r_p^2 \right\rangle + \left\langle r^2 \right\rangle_P + \frac{N}{Z} \left\langle r^2 \right\rangle_N + \frac{2}{Z} \left(\frac{\hbar}{mc} \right)^2 \sum_{\alpha q} \mu_q \left\langle \boldsymbol{\sigma} \cdot \boldsymbol{\ell} \right\rangle_{\alpha q} \tag{3.2}$$

where q runs over the nuclear species and α runs over all single particle states of spcies q. $\boldsymbol{\sigma}$ is the vector operator of Pauli matrices, while $\boldsymbol{\ell}$ is the angular momentum operator $-i(\boldsymbol{r}\times\nabla)$. $\langle r^2\rangle_P$ and $\langle r^2\rangle_N$ refer to the square charge radii of the proton and the neutron, while μ_q to their respective magnetic dipole moment in units of nuclear magneton.
All square charge radii computed in this work use the set of parameters in table 3.1.

Parameter	Value	Units
$\langle r^2\rangle_P$	0.64	fm^2
$\langle r^2\rangle_N$	-0.11	fm^2
μ_p	2.792847	-
μ_n	-1.913043	-

Table 3.1: Parameters used to compute the charge mean square radius

3.1.2. Deformation parameters

When dealing with deformed nuclei, mean square radii are not sufficient to characterize the nuclear density. The main parameter used is the quadrupole deformation parameter β_2, defined already in section (REF), it can be computed through the actual mean square radius with formula 3.3

$$\beta_2 = \frac{4\pi\,\langle Y_{20}\rangle}{3A\,\langle r^2\rangle} \tag{3.3}$$

where $\langle r^2\rangle$ is the total mean square radius of the nucleus

$$\langle r^2\rangle = \frac{\int(\rho_n+\rho_p)r^2 d\boldsymbol{r}}{\int(\rho_n+\rho_p)d\boldsymbol{r}} = \langle x^2+y^2+z^2\rangle\,. \tag{3.4}$$

For spherical nuclei, $\beta_2 = 0$, while for deformed ones, thanks to the normalization with respect to the total radius and mass, the β_2 parameter can be used to compare different nuclei across the nuclide chart.

3.2. Parameters and mesh choice

All `hfbcs_qrpa` calculations were performed using a mesh size of 0.1 fm, no pairing interaction, and a radial mesh size whose radius is equal to the side of the box in our computation. The lattice of our code depends on the extension of the nucleus, which is directly determined by its mass A; since the number of subdivisions that allows reasonable CPU times on a laptop caps around $60-70$, step sizes vary across different calculations. In the results shown here, for ^{16}O, we are able to reach a 0.3 fm step size, while for the

heaviest, ^{90}Zr, we are only able to reach 0.42 fm. The reason behind this choice is that as the nucleus size increases, a bigger box is needed to ensure that all relevant states are able to decay to zero at the boundary. All the data reported in this chapter is computed with the SLy5 parametrization [6].

3.3. Results for ^{16}O

The first results we will take a look at are the ones for ^{16}O. It's the best candidate for gauging the solver's performance, as it is a very light, double magic nucleus, meaning it has no pairing interaction and a spherical shape.
All calculations are performed on a box of size $[-9, 9]$ fm in all three directions and a step size of 0.3 fm, correponding to $2 \cdot 60^3$ mesh points.

3.3.1. Results neglecting Coulomb interaction

Since the Skyrme functional is complex and nuanced, results are shown for increasing terms in 1.41, starting from C_0^ρ, C_1^ρ, C_0^τ, C_1^τ reported in table 3.2, neglecting the Coulomb interaction. Without further terms, the spin-orbit field $\boldsymbol{B}(\boldsymbol{r})$ is null, hence the $1p_{3/2}$ and $1p_{1/2}$ levels show degeneration in energy.
Since $N = Z$, assuming equal masses the single-particle equations will be exactly equal between the two species, therefore only neutron results are reported. Note that C_1 terms reduce to being null in this case, that is until we either break the $N = Z$ equality or introduce the Coulomb interaction. In table 3.3 the $C_0^{\boldsymbol{\nabla}\cdot\boldsymbol{J}}$ and $C_1^{\boldsymbol{\nabla}\cdot\boldsymbol{J}}$ terms are included

Physical quantities

		GCG	hfbcs_qrpa	Δ	Δ%
E_{TOT}	[MeV]	-141.582	-141.582	-	-
$\langle r_n^2 \rangle^{1/2}$	[fm]	2.6504	2.6510	0.0006	2.26×10^{-2}
$\langle r_{ch}^2 \rangle^{1/2}$	[fm]	2.7486	2.7491	0.0005	1.82×10^{-2}

Neutron energy levels

		GCG	hfbcs_qrpa	Δ	Δ%
$1s_{1/2}$	[MeV]	-36.142	-36.139	0.003	8.30×10^{-3}
$1p_{3/2}$	[MeV]	-18.573	-18.572	0.001	5.38×10^{-3}
$1p_{1/2}$	[MeV]	-18.573	-18.572	0.001	5.38×10^{-3}

Table 3.2: ^{16}O including C_0^ρ, C_1^ρ, C_0^τ, C_1^τ terms, neglecting Coulomb interaction.

just for the spin-orbit field $\boldsymbol{B}(\boldsymbol{r})$, but not for the mean field $U(\boldsymbol{r})$; from an interaction point of view, it's as if we were neglecting the spin-gradient coupling term [6]. As expected, the $1p_{3/2}$ and $1p_{1/2}$ degeneration is removed, displaying the spin-orbit splitting, which lowers the $j = 3/2$ level and raises the $j = 1/2$ level.

Physical quantities

		GCG	hfbcs_qrpa	Δ	Δ%
E_{TOT}	[MeV]	-142.080	-142.080	-	-
$\langle r_n^2 \rangle^{1/2}$	[fm]	2.6516	2.6516	-	-
$\langle r_{ch}^2 \rangle^{1/2}$	[fm]	2.7497	2.7497	-	-

Neutron energy levels

		GCG	hfbcs_qrpa	Δ	Δ%
$1s_{1/2}$	[MeV]	-36.314	-36.312	0.002	5.5×10^{-3}
$1p_{3/2}$	[MeV]	-20.696	-20.696	-	-
$1p_{1/2}$	[MeV]	-14.335	-14.335	-	-

Table 3.3: ^{16}O including C_0^{ρ}, C_1^{ρ}, C_0^{τ}, C_1^{τ}, $C_0^{\boldsymbol{\nabla}\cdot\boldsymbol{J}}$, $C_1^{\boldsymbol{\nabla}\cdot\boldsymbol{J}}$ terms, neglecting Coulomb interaction and spin-gradient coupling.

Lastly, the $C_0^{\boldsymbol{\nabla}\cdot\boldsymbol{J}}$ and $C_1^{\boldsymbol{\nabla}\cdot\boldsymbol{J}}$ terms are also included in the calculation of the mean-field, resulting in the full implementation of the Skyrme functional. As shown in table 3.4, the effect of this addition on the ground state is little, as the spin current $J_{\mu\nu}$ is small in light, closed shell nuclei.

Physical quantities

		GCG	hfbcs_qrpa	Δ	Δ%
E_{TOT}	[MeV]	-142.074	-142.074	-	-
$\langle r_n^2 \rangle^{1/2}$	[fm]	2.6515	2.6516	0.0001	3.77×10^{-3}
$\langle r_{ch}^2 \rangle^{1/2}$	[fm]	2.7497	2.7497	-	-

Neutron energy levels

		GCG	hfbcs_qrpa	Δ	Δ%
$1s_{1/2}$	[MeV]	-36.309	-36.308	0.001	2.75×10^{-3}
$1p_{3/2}$	[MeV]	-20.684	-20.685	0.001	4.83×10^{-3}
$1p_{1/2}$	[MeV]	-14.361	-14.361	-	-

Table 3.4: ^{16}O neglecting Coulomb interaction.

Results including Coulomb interaction

As the final addition to get a complete and accurate description of ^{16}O, the Coulomb interaction is included as detailed in section 1.3. Results are shown in table 3.5. As

Physical quantities

		GCG	hfbcs_qrpa	Δ	Δ%
E_{TOT}	[MeV]	-128.402	-128.400	0.002	1.56×10^{-3}
$\langle r_n^2 \rangle^{1/2}$	[fm]	2.6584	2.6585	0.0001	3.76×10^{-3}
$\langle r_p^2 \rangle^{1/2}$	[fm]	2.6835	2.6836	0.0001	3.73×10^{-3}
$\langle r_{ch}^2 \rangle^{1/2}$	[fm]	2.7805	2.7803	0.0002	7.19×10^{-3}

Neutron energy levels

		GCG	hfbcs_qrpa	Δ	Δ%
$1s_{1/2}$	[MeV]	-36.140	-36.137	0.003	8.30×10^{-3}
$1p_{3/2}$	[MeV]	-20.611	-20.611	-	-
$1p_{1/2}$	[MeV]	-14.427	-14.428	0.001	6.93×10^{-3}

Proton energy levels

		GCG	hfbcs_qrpa	Δ	Δ%
$1s_{1/2}$	[MeV]	-32.349	-32.345	0.004	1.24×10^{-2}
$1p_{3/2}$	[MeV]	-17.137	-17.137	-	-
$1p_{1/2}$	[MeV]	-11.081	-11.082	0.001	9.02×10^{-3}

Table 3.5: ^{16}O complete of the Skyrme functional and Coulomb interaction.

shown in tables 3.2, 3.3 and 3.4, results for ^{16}O are in great agreement with the output of the well established `hfbcs_qrpa` code.

3.4. Results for heavier nuclei

In the following section, results for some spherical nuclei heavier than ^{16}O are presented in tables 3.6, 3.7 and 3.8. Our code still shows good agreement with the `hfbcs_qrpa` one. A slight increase of the numerical error can be observed as the step size increaseas, which is compatible with the polynomial error in the finite difference method.

Physical quantities

		GCG	hfbcs_qrpa	Δ	Δ%
E_{TOT}	[MeV]	-415.955	-415.931	0.024	5.77×10^{-3}
$\langle r_n^2 \rangle^{1/2}$	[fm]	3.6106	3.6110	0.0004	1.11×10^{-2}
$\langle r_p^2 \rangle^{1/2}$	[fm]	3.4502	3.4507	0.0005	1.45×10^{-2}
$\langle r_{ch}^2 \rangle^{1/2}$	[fm]	3.5274	3.5060	0.0214	0.610

Neutron energy levels

		GCG	hfbcs_qrpa	Δ	Δ%
$1s_{1/2}$	[MeV]	-49.758	-49.752	0.006	1.21×10^{-2}
$1p_{3/2}$	[MeV]	-35.952	-35.949	0.003	8.34×10^{-3}
$1p_{1/2}$	[MeV]	-33.891	-33.891	-	-
$1d_{5/2}$	[MeV]	-22.170	-22.169	0.001	4.51×10^{-3}
$2s_{1/2}$	[MeV]	-17.720	-17.720	-	-
$1d_{3/2}$	[MeV]	-17.431	-17.434	0.003	1.72×10^{-2}
$1f_{7/2}$	[MeV]	-9.262	-9.261	0.001	1.08×10^{-2}

Proton energy levels

		GCG	hfbcs_qrpa	Δ	Δ%
$1s_{1/2}$	[MeV]	-45.936	-45.930	0.006	1.31×10^{-2}
$1p_{3/2}$	[MeV]	-34.314	-34.311	0.003	8.74×10^{-3}
$1p_{1/2}$	[MeV]	-30.482	-30.483	0.001	3.28×10^{-3}
$1d_{5/2}$	[MeV]	-22.455	-22.454	0.001	4.45×10^{-3}
$2s_{1/2}$	[MeV]	-16.753	-16.751	0.002	1.19×10^{-2}
$1d_{3/2}$	[MeV]	-15.337	-15.340	0.003	1.96×10^{-2}

Table 3.6: ^{48}Ca, box size [-12, 12] fm, step size 0.34 fm

3.4.1. Comparison with experimental binding energies

The Skyrme functional is highly successful at producing theoretical values in great accordance with experimental data, just by fitting a small set of parameters [2]. In table 3.9, binding energies of some of the nuclei studied in this work are compared with experimental values, taken from the Atomic Mass Data Center [1].

Physical quantities

		GCG	hfbcs_qrpa	Δ	Δ%
E_{TOT}	[MeV]	-482.805	-482.700	0.105	2.18×10^{-2}
$\langle r_n^2 \rangle^{1/2}$	[fm]	3.6422	3.6433	0.0011	3.02×10^{-2}
$\langle r_p^2 \rangle^{1/2}$	[fm]	3.6968	3.6979	0.0011	2.97×10^{-2}
$\langle r_{ch}^2 \rangle^{1/2}$	[fm]	3.7722	3.7682	0.0040	0.106

Neutron energy levels

		GCG	hfbcs_qrpa	Δ	Δ%
$1s_{1/2}$	[MeV]	-54.277	-54.260	0.017	3.13×10^{-2}
$1p_{3/2}$	[MeV]	-41.571	-41.562	0.009	2.16×10^{-2}
$1p_{1/2}$	[MeV]	-39.613	-39.611	0.002	5.05×10^{-3}
$1d_{5/2}$	[MeV]	-28.536	-28.530	0.006	2.10×10^{-2}
$2s_{1/2}$	[MeV]	-23.539	-23.545	0.006	2.55×10^{-2}
$1d_{3/2}$	[MeV]	-23.367	-23.361	0.006	2.57×10^{-2}
$1f_{7/2}$	[MeV]	-16.019	-16.018	0.001	6.24×10^{-3}

Proton energy levels

		GCG	hfbcs_qrpa	Δ	Δ%
$1s_{1/2}$	[MeV]	-43.754	-43.740	0.014	3.20×10^{-2}
$1p_{3/2}$	[MeV]	-31.561	-31.555	0.006	1.90×10^{-2}
$1p_{1/2}$	[MeV]	-29.545	-29.545	-	-
$1d_{5/2}$	[MeV]	-19.017	-19.016	0.001	5.26×10^{-3}
$2s_{1/2}$	[MeV]	-14.004	-14.012	0.008	5.71×10^{-2}
$1d_{3/2}$	[MeV]	-13.891	-13.887	0.004	2.88×10^{-2}
$1f_{7/2}$	[MeV]	-6.934	-6.935	0.001	1.44×10^{-2}

Table 3.7: ^{56}Ni, box size [-13, 13] fm, step size 0.37 fm

Physical quantities

		GCG	hfbcs_qrpa	Δ	Δ%
E_{TOT}	[MeV]	-783.587	-783.325	0.262	3.34×10^{-2}
$\langle r_n^2 \rangle^{1/2}$	[fm]	4.2854	4.2872	0.0018	4.20×10^{-2}
$\langle r_p^2 \rangle^{1/2}$	[fm]	4.2196	4.2212	0.0016	3.79×10^{-2}
$\langle r_{ch}^2 \rangle^{1/2}$	[fm]	4.2767	4.2704	0.0063	0.148

Neutron energy levels

		GCG	hfbcs_qrpa	Δ	Δ%
$1s_{1/2}$	[MeV]	-55.636	-55.615	0.021	3.78×10^{-2}
$1p_{3/2}$	[MeV]	-45.324	-45.309	0.015	3.31×10^{-2}
$1p_{1/2}$	[MeV]	-44.172	-44.160	0.012	2.72×10^{-2}
$1d_{5/2}$	[MeV]	-34.148	-34.137	0.011	3.22×10^{-2}
$2s_{1/2}$	[MeV]	-31.393	-31.391	0.002	6.37×10^{-3}
$1d_{3/2}$	[MeV]	-29.802	-29.797	0.005	1.68×10^{-2}
$1f_{7/2}$	[MeV]	-22.755	-22.748	0.007	3.08×10^{-2}
$2p_{3/2}$	[MeV]	-17.837	-17.840	0.003	1.68×10^{-2}
$1f_{5/2}$	[MeV]	-17.568	-17.563	0.005	2.85×10^{-2}
$2p_{1/2}$	[MeV]	-15.729	-15.723	0.006	3.82×10^{-2}
$1g_{9/2}$	[MeV]	-11.586	-11.580	0.006	5.18×10^{-2}

Proton energy levels

		GCG	hfbcs_qrpa	Δ	Δ%
$1s_{1/2}$	[MeV]	-44.973	-44.956	0.017	3.78×10^{-2}
$1p_{3/2}$	[MeV]	-36.347	-36.336	0.011	3.03×10^{-2}
$1p_{1/2}$	[MeV]	-34.121	-34.115	0.006	1.76×10^{-2}
$1d_{5/2}$	[MeV]	-26.766	-26.759	0.007	2.62×10^{-2}
$2s_{1/2}$	[MeV]	-22.175	-22.178	0.003	1.35×10^{-2}
$1d_{3/2}$	[MeV]	-21.216	-21.214	0.002	9.43×10^{-3}
$1f_{7/2}$	[MeV]	-16.722	-16.718	0.004	2.39×10^{-2}
$2p_{3/2}$	[MeV]	-10.239	-10.236	0.003	2.93×10^{-2}
$1f_{5/2}$	[MeV]	-9.613	-9.618	0.005	5.20×10^{-2}
$2p_{1/2}$	[MeV]	-8.108	-8.104	0.004	4.94×10^{-2}

Table 3.8: ^{90}Zr, box size [-15, 15] fm, step size 0.43 fm

	^{16}O	^{48}Ca	^{56}Ni	^{90}Zr
E_{th}	128.40	415.95	482.80	783.59
E_{exp}	127.62	414.33	483.99	783.89

Table 3.9: Comparison of experimental binding energies in MeV with theoretical calculated values.

4 | Results for deformed nuclei

Having established that the code works well for spherical nuclei, we can start treading in deformation territory.

4.1. ^{24}Mg

In the following section, results for ^{24}Mg are presented, it's a natural choice to study how well deformations are represented by our framework, since it's light, very deformed and shows no pairing interaction in its ground state.

4.1.1. `HFBTHO` code and calculation details

`HFBTHO`

To benchmark the code in the case of nuclear deformation, the `HFBTHO` code was used [23], it's a HFB code which minimizes the energy functional on a (Transformed) Harmonic Oscillator basis. Since ^{24}Mg is a light nucleus, it still works well in this case.
All calculations are carried out using a number of shells of 12 and a null pairing interaction. Default paramters for the quadrupole constraints were used.

Code parameters and axial constraint

As for our code, calculations are performed on a box $[-10, 10]$ fm. In the case of the ground state calculation, a step size of 0.33 fm is used, with a starting guess of a deformed Woods-Saxon with $\beta_2 = 0.4$.
The calculation in the case of the deformation curve is carried out imposing the following constraints

$$\langle \mathrm{Re}\, \mathcal{Q}_{22} \rangle = 0 \tag{4.1}$$

$$\langle \mathrm{Im}\, \mathcal{Q}_{22} \rangle = 0 \tag{4.2}$$

$$\langle \mathcal{Q}_{20} \rangle = q_{20}. \tag{4.3}$$

These constraints altogether impose an axial deformation on the system. This is done because on a full mesh like in our case, the nucleus may deform on a different axis from the chosen one (z), resulting in spurious contributions to the real deformation curve; moreover, the axial symmetry of `HFBTHO` doesn't allow broken axial symmetry configurations.
Regarding the stiffness c and damping parameter μ of ALM 2.3.1, $c = 0.001$ and $\mu = 0.1$ were used. As for convergence criteria, a tolerance of 0.001 on the value of $\beta_2 - \beta_{2,\text{target}}$ was used.

Ground state

Table 4.1 reports data for the ground state of ^{24}Mg, compare with `HFBTHO` results, while figure 4.1 shows the total particle density. Charge radii for the two codes are displayed but not compared, due to different formulas used for their computation.

		GCG	`HFBTHO`	Δ	Δ%
E_{TOT}	[MeV]	-195.854	-197.030	1.176	0.597
$\langle r_n^2 \rangle^{1/2}$	[fm]	3.0124	2.9996	0.0128	0.427
$\langle r_p^2 \rangle^{1/2}$	[fm]	3.0475	3.0326	0.0149	0.491
$\langle r_{ch}^2 \rangle^{1/2}$	[fm]	3.5390	3.4614	-	-
$\langle z^2 \rangle^{1/2}$	[fm]	2.1454	-	-	-
$\langle x^2 \rangle^{1/2}$	[fm]	1.5112	-	-	-
$\langle y^2 \rangle^{1/2}$	[fm]	1.5147	-	-	-
β_2	[-]	0.399	0.390	0.009	2.3

Table 4.1: Results for ^{24}Mg ground state, no pairing interaction, box $[-10, 10]$ fm, step size 0.33 fm, SKM* parametrization.

Deformation curve

In figure 4.2, the deformation curve is shown for ^{24}Mg, without pairing. To counteract the sharp rise in CPU time, due to the high number of points in the curve, a coarser grid is used, hence the different minimum energy and β_2 values than the ones reported in table 4.1.

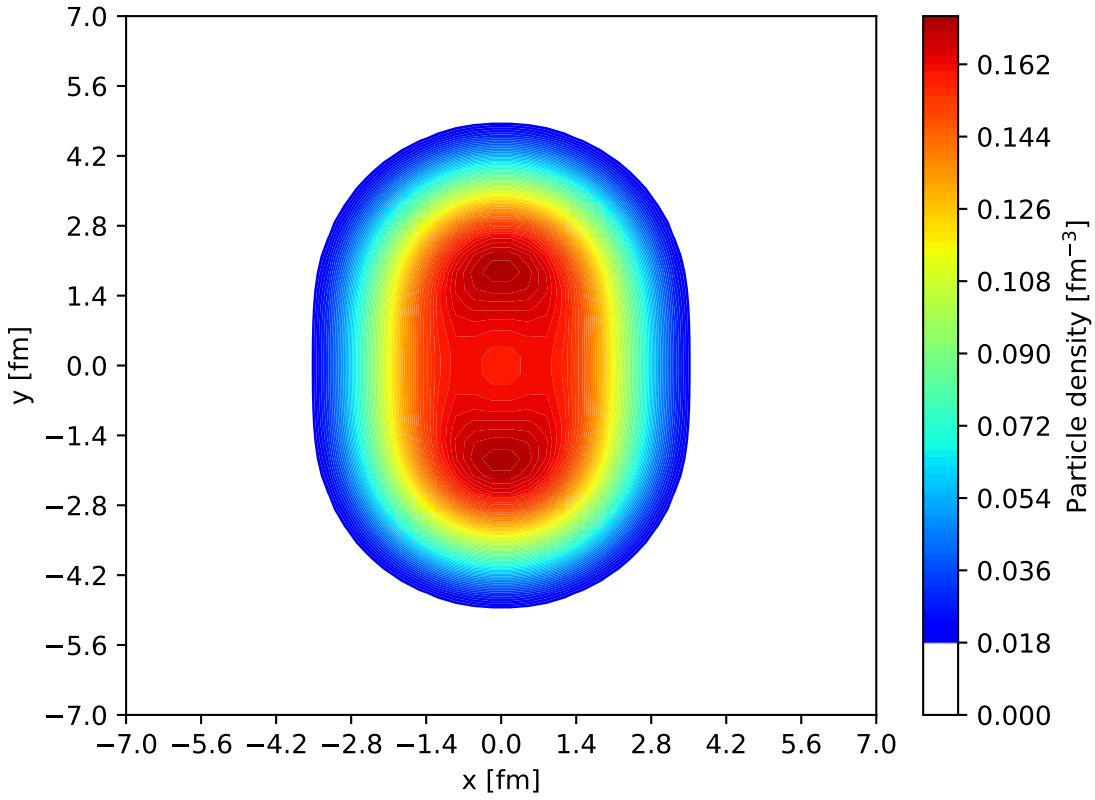

Figure 4.1: Magnesium ground state density, calculation done on a box $[-10, 10]$ fm, step size 0.33 fm, SKM* parametrization

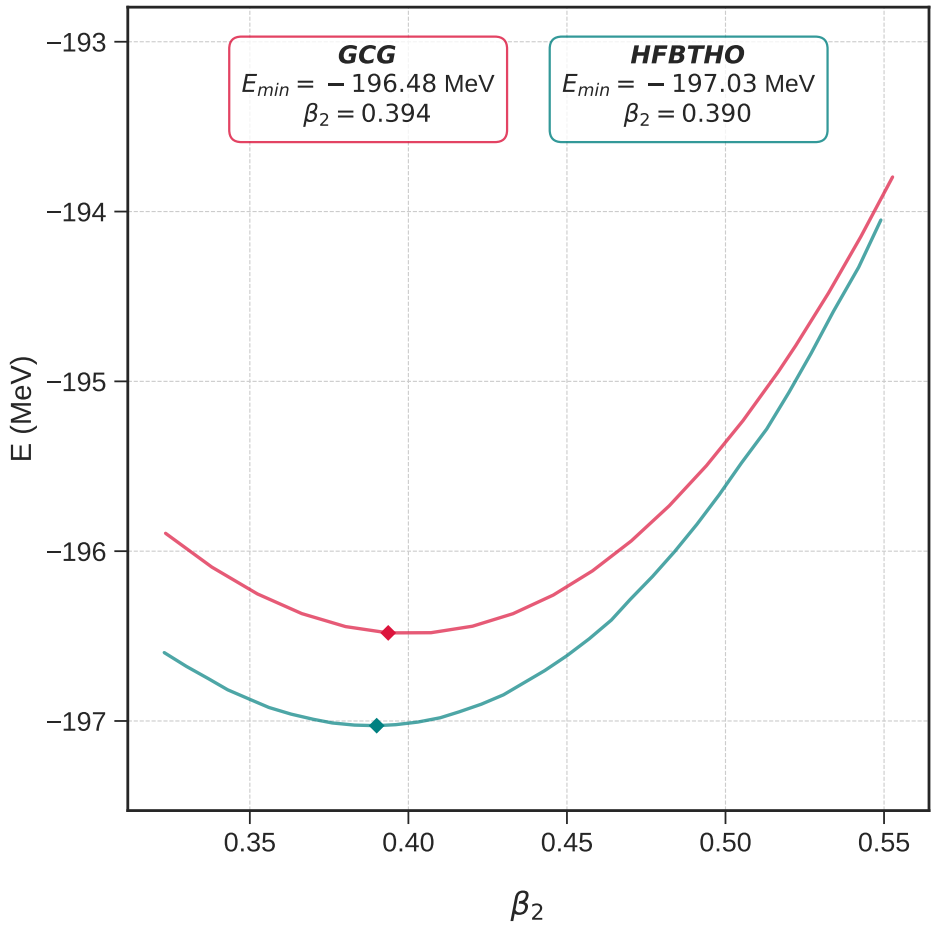

Figure 4.2: Magnesium deformation curve, no pairing interaction, calculation done on a box $[-10, 10]$ fm, step size 0.66 fm, SKM* parametrization.

Bibliography

[1] Atomic Mass Data Center. Atomic Mass Data Center. URL https://www-nds.iaea.org/amdc/. Accessed: 19 October 2025.

[2] M. Bender, P.-H. Heenen, and P.-G. Reinhard. Self-consistent mean-field models for nuclear structure. *Reviews of Modern Physics*, 75(1):121–180, 2003. doi: 10.1103/RevModPhys.75.121.

[3] W. Bertozzi, J. Friar, J. Heisenberg, and J. Negele. Contributions of neutrons to elastic electron scattering from nuclei. *Physics Letters B*, 41(4):408–414, 1972. ISSN 0370-2693. doi: https://doi.org/10.1016/0370-2693(72)90662-4. URL https://www.sciencedirect.com/science/article/pii/0370269372906624.

[4] Å. Björck. Numerics of gram-schmidt orthogonalization. *Linear Algebra and Its Applications*, 197:297–316, 1994.

[5] E. Chabanat, P. Bonche, P. Haensel, J. Meyer, and R. Schaeffer. A skyrme parametrization from subnuclear to neutron star densities. *Nuclear Physics A*, 627(4):710–746, 1997. ISSN 0375-9474. doi: https://doi.org/10.1016/S0375-9474(97)00596-4. URL https://www.sciencedirect.com/science/article/pii/S0375947497005964.

[6] 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. *Nuclear Physics A*, 635(1):231–256, 1998. ISSN 0375-9474. doi: https://doi.org/10.1016/S0375-9474(98)00180-8. URL https://www.sciencedirect.com/science/article/pii/S0375947498001808.

[7] M. Chen, T. Li, B. Schuetrumpf, P.-G. Reinhard, and W. Nazarewicz. Three-dimensional skyrme hartree-fock-bogoliubov solver in coordinate-space representation. *Computer Physics Communications*, 276:108344, 2022. ISSN 0010-4655. doi: https://doi.org/10.1016/j.cpc.2022.108344. URL https://www.sciencedirect.com/science/article/pii/S0010465522000625.

[8] G. Colo' and X. Roca-Maza. User guide for the hfbcs-qrpa(v1) code.

[9] G. Colò. Nuclear density functional theory. *Advances in Physics: X*, 5(1):1740061, 2020. doi: 10.1080/23746149.2020.1740061. URL https://doi.org/10.1080/23746149.2020.1740061.

[10] R. Y. Cusson, P. G. Reinhard, M. R. Strayer, J. A. Maruhn, and W. Greiner. Density as a constraint and the separation of internal excitation energy in tdhf. *Zeitschrift für Physik A Atoms and Nuclei*, 320:475–482, 1985. doi: 10.1007/BF01415725. URL https://doi.org/10.1007/BF01415725.

[11] J. Erler, P. Klüpfel, and P.-G. Reinhard. Misfits in skyrme–hartree–fock. *Journal of Physics G: Nuclear and Particle Physics*, 37(6):064001, mar 2010. doi: 10.1088/0954-3899/37/6/064001. URL https://doi.org/10.1088/0954-3899/37/6/064001.

[12] H. Flocard, P. Quentin, A. Kerman, and D. Vautherin. Nuclear deformation energy curves with the constrained hartree-fock method. *Nuclear Physics A*, 203(3):433–472, 1973. ISSN 0375-9474. doi: https://doi.org/10.1016/0375-9474(73)90357-6. URL https://www.sciencedirect.com/science/article/pii/0375947473903576.

[13] G. H. Golub and C. F. van Loan. *Matrix Computations*. JHU Press, fourth edition, 2013. ISBN 1421407949 9781421407944. URL http://www.cs.cornell.edu/cv/GVL4/golubandvanloan.htm.

[14] H.-Q. Gu, H. Liang, W. H. Long, N. Van Giai, and J. Meng. Slater approximation for coulomb exchange effects in nuclear covariant density functional theory. *Phys. Rev. C*, 87:041301, Apr 2013. doi: 10.1103/PhysRevC.87.041301. URL https://link.aps.org/doi/10.1103/PhysRevC.87.041301.

[15] G. Guennebaud, B. Jacob, et al. Eigen. http://eigen.tuxfamily.org, 2010.

[16] P. Hohenberg and W. Kohn. Inhomogeneous electron gas. *Phys. Rev.*, 136:B864–B871, Nov 1964. doi: 10.1103/PhysRev.136.B864. URL https://link.aps.org/doi/10.1103/PhysRev.136.B864.

[17] J. D. Jackson. *Classical Electrodynamics*. John Wiley & Sons, New York, 3rd edition, 1998. ISBN 978-0471309321.

[18] A. V. Knyazev. *A Preconditioned Conjugate Gradient Method for Eigenvalue Problems and its Implementation in a Subspace*, pages 143–154. Birkhäuser Basel, Basel, 1991. ISBN 978-3-0348-6332-2. doi: 10.1007/978-3-0348-6332-2_11. URL https://doi.org/10.1007/978-3-0348-6332-2_11.

[19] C. Lanczos. Solution of systems of linear equations by minimized iterations. *J. Res. Nat. Bur. Standards*, 49(1):33–53, 1952.

[20] B. Li, Z. Ren, and P. Zhao. Efficient solution for the dirac equation in 3d lattice space with the conjugate gradient method. *Physical Review C*, 102(4):044307, 2020.

[21] Y. Li, H. Xie, R. Xu, C. You, and N. Zhang. A parallel generalized conjugate gradient method for large scale eigenvalue problems. *CCF Transactions on High Performance Computing*, 2(2):111–122, jun 2020. ISSN 2524-4930. doi: 10.1007/s42514-020-00029-6. URL `https://doi.org/10.1007/s42514-020-00029-6`. The corresponding computing package can be downloaded from the web site: `https://github.com/pase2017/GCGE-1.0`.

[22] L. Lin, S. Shao, and W. E. Efficient iterative method for solving the dirac–kohn–sham density functional theory. *Journal of Computational Physics*, 245:205–217, 2013. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2013.03.030. URL `https://www.sciencedirect.com/science/article/pii/S0021999113002088`.

[23] P. Marević, N. Schunck, E. Ney, R. Navarro Pérez, M. Verriere, and J. O'Neal. Axially-deformed solution of the skyrme-hartree-fock-bogoliubov equations using the transformed harmonic oscillator basis (iv) hfbtho (v4.0): A new version of the program. *Computer Physics Communications*, 276:108367, 2022. ISSN 0010-4655. doi: https://doi.org/10.1016/j.cpc.2022.108367. URL `https://www.sciencedirect.com/science/article/pii/S0010465522000868`.

[24] T. Nottoli, I. Gianni, A. Levitt, and F. Lipparini. A robust, open-source implementation of the locally optimal block preconditioned conjugate gradient for large eigenvalue problems in quantum chemistry. *Theoretical Chemistry Accounts*, 142 (8):69, 2023. ISSN 1432-2234. doi: 10.1007/s00214-023-03010-y. URL `https://doi.org/10.1007/s00214-023-03010-y`.

[25] J. W. Pearson and J. Pestana. Preconditioners for krylov subspace methods: An overview. *GAMM-Mitteilungen*, 43(4):e202000015, 2020.

[26] P.-G. Reinhard and H. Flocard. Nuclear effective forces and isotope shifts. *Nuclear Physics A*, 584(3):467–488, 1995. ISSN 0375-9474. doi: https://doi.org/10.1016/0375-9474(94)00770-N. URL `https://www.sciencedirect.com/science/article/pii/037594749400770N`.

[27] P. Ring and P. Schuck. *The nuclear many-body problem*. Springer Science & Business Media, 2004.

[28] L. M. Robledo, T. R. Rodríguez, and R. R. Rodríguez-Guzmán. Mean field and beyond description of nuclear structure with the gogny force: a review. *Journal of Physics G: Nuclear and Particle Physics*, 46(1):013001, dec 2018. doi: 10.1088/1361-6471/aadebd. URL https://doi.org/10.1088/1361-6471/aadebd.

[29] 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. *Computer Physics Communications*, 187:175–194, 2015. ISSN 0010-4655. doi: https://doi.org/10.1016/j.cpc.2014.10.001. URL https://www.sciencedirect.com/science/article/pii/S0010465514003361.

[30] Y. Saad. *Numerical Methods for Large Eigenvalue Problems*. Manchester University Press, Manchester, UK, 1992.

[31] J. R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. 1994. URL http://www.cs.cmu.edu/~quake-papers/painless-conjugate-gradient.pdf.

[32] T. Skyrme. The effective nuclear potential. *Nuclear Physics*, 9(4):615–634, 1958. ISSN 0029-5582. doi: https://doi.org/10.1016/0029-5582(58)90345-6. URL https://www.sciencedirect.com/science/article/pii/0029558258903456.

[33] G. L. G. Sleijpen and H. A. Van der Vorst. A jacobi–davidson iteration method for linear eigenvalue problems. *SIAM Review*, 42(2):267–293, 2000. doi: 10.1137/S0036144599363084. URL https://doi.org/10.1137/S0036144599363084.

[34] A. Staszczak, M. Stoitsov, A. Baran, and W. Nazarewicz. Augmented lagrangian method for constrained nuclear density functional theory. *The European Physical Journal A*, 46:85–90, 2010. doi: 10.1140/epja/i2010-11018-9. URL https://doi.org/10.1140/epja/i2010-11018-9.

[35] P. Stevenson and M. Barton. Low-energy heavy-ion reactions and the skyrme effective interaction. *Progress in Particle and Nuclear Physics*, 104:142–164, 2019.

[36] B. Stroustrup. An overview of c++. In *Proceedings of the 1986 SIGPLAN workshop on Object-oriented programming*, pages 7–18, 1986.

[37] D. Vautherin and D. M. Brink. Hartree-fock calculations with skyrme's interaction. i. spherical nuclei. *Phys. Rev. C*, 5:626–647, Mar 1972. doi: 10.1103/PhysRevC.5.626. URL https://link.aps.org/doi/10.1103/PhysRevC.5.626.

[38] J. Woo, S. Kim, and W. Y. Kim. Gaussian-approximated poisson preconditioner for iterative diagonalization in real-space density functional theory. *The Journal of*

Physical Chemistry A, 127(17):3883–3893, 2023. doi: 10.1021/acs.jpca.2c09111. URL https://doi.org/10.1021/acs.jpca.2c09111. PMID: 37094552.

[39] P.-b. Zhou. *Finite Difference Method*, pages 63–94. Springer Berlin Heidelberg, Berlin, Heidelberg, 1993. ISBN 978-3-642-50319-1. doi: 10.1007/978-3-642-50319-1_3. URL https://doi.org/10.1007/978-3-642-50319-1_3.

A | Appendix

A.1. Spherical harmonics

Spherical harmonics, of order λ, μ, are defined as

$$Y_{\lambda\mu}(\theta, \phi) = (-1)^{\mu} \sqrt{\frac{2\lambda + 1}{4\pi} \frac{(\lambda - \mu)!}{(\lambda + \mu)!}} P_{\lambda}^{\mu}(\cos\theta)\, e^{i\mu\phi}. \tag{A.1}$$

Being able to provide the expression for arbitrary μ, λ through an algorithm is important in the current framework, to solve the Poisson equation and investigate nuclear properties. The major challenge is to generate the associated Legendre polynomials P_{λ}^{μ}. They can be expressed in the form (for positive μ)

$$P_{\lambda}^{\mu}(x) = (1 - x^2)^{\mu/2} \frac{\mathrm{d}^{\mu} P_{\lambda}(x)}{\mathrm{d}x^{\mu}}, \tag{A.2}$$

where $x = \cos\theta$ and

$$P_{\lambda}(x) = \frac{1}{2^{\lambda}\lambda!} \frac{\mathrm{d}^{\lambda}(x^2 - 1)^{\lambda}}{\mathrm{d}x^{\lambda}}. \tag{A.3}$$

To compute the arbitrary λ, μ associated Legendre polynomial we can employ a recursive approach, setting $\lambda = \mu$

$$P_{\mu}^{\mu}(x) = (2\mu - 1)!!\, (1 - x^2)^{\mu/2}, \tag{A.4}$$

where $(2\mu - 1)!! = 1 \cdot 3 \cdot 5 \ldots (2\mu - 1)$ denotes the double factorial. Once $P_{\mu}^{\mu}(x)$ is known, the next element with $\lambda = \mu + 1$ reads

$$P_{\mu+1}^{\mu}(x) = x(2\mu + 1) P_{\mu}^{\mu}(x). \tag{A.5}$$

All higher orders are then generated using the standard upward recurrence relation in λ:

$$(\lambda - \mu + 1)\, P_{\lambda+1}^{\mu}(x) = (2\lambda + 1)\, x\, P_{\lambda}^{\mu}(x) - (\lambda + \mu)\, P_{\lambda-1}^{\mu}(x), \tag{A.6}$$

valid for all $\lambda \geq \mu + 1$.

A.1.1. Algorithm

1. Compute the base case P_{μ}^{μ} from the closed-form formula.
2. If $\mu = \lambda$ the procedure ends, otherwise
3. Evaluate $P_{\mu+1}^{\mu}$, if $\lambda = \mu + 1$ the procedure ends, otherwise
4. Apply the recurrence relation $P_{\lambda+1}^{\mu}$ until the desired degree is reached

This ought to be applied only for $\mu \geq 0$. For $\mu < 0$ the procedure is carried out using $-\mu$ and in the end using the relation

$$Y_{\lambda-\mu} = (-1)^{\mu} Y_{\lambda\mu}^{*} \tag{A.7}$$

A.2. 5-point derivatives

The first and second derivatives of a function $\psi(x)$ in $x = x_i$, using 5-points formulae, read

$$\psi'(x_i) = \frac{\psi_{i-2} - 8\psi_{i-1} + 8\psi_{i+1} - \psi_{i+2}}{12h} \tag{A.8}$$

$$\psi''(x_i) = \frac{-\psi_{i-2} + 16\psi_{i-1} - 30\psi_i + 16\psi_{i+1} - \psi_{i+2}}{12h^2} \tag{A.9}$$

List of Figures

List of Tables

List of Symbols

Variable	Description	SI unit
$\boldsymbol{u}$	solid displacement	m
$\boldsymbol{u}_f$	fluid displacement	m

Acknowledgements

Here you might want to acknowledge someone.