WEEK3: Skyrme Potential

Ni Fang
Kai Wang
Claudia Gonzalez-Boquera

TALENT 2016

1 Structure

This program calculates the eigenvalues and wavefunctions of each given nlj orbital when having a t0-t3 Skyrme potential.

The program consists in a self-consistent method to calculate the levels of a given nucleus.

We start from wavefunctions and densities trials (using the WS 3D code). When having this, we calculate the self-consistent loop. We start calculating the full series of levels for the nucleus we want to study. As in the previous codes, the eigenvalues are found with the bissection method, which includes the numerov method in order to calculate the eigenfunction for each eigenvalue. For each eigenfunction found, we calculate its nodes, and we modify the extremes of the bissection method according to them. When the extremes are closer than epsi, we exit the loop, and put the eigenvalue as the right extreme of the bissection method. Then we recalculate the eigenfunction with the shooting method, with right hand side equal to 0, in order to find the convergence, and we normalize it.

We do this for neutrons and for protons separately.

When all possible orbitals are calculated, we rearrange the energy in ascending order, in order to find the correct order in the levels.

After, the density profiles are calculated.

When having all of this, we calculate and compare the Hartree-Fock total energy with the one obtained in the previous loop. If the difference is less than epsilon, we exit the loop, as we have reached the convergence.

1.1 Functions and Subroutines

- Subroutine WoodsSaxon3D(energy_n,energy_p,quant_n,quant_p,wf_n, wf_p): subroutine to get the trial energies and wavefunctions. The energies of the neutrons (protons) will be stored in the array energy_n (energy_p), the neutron (proton) quantum numbers in quant_n (quant_p) and the wavefunctions in wf_n (wf_p). In order to arrange the energy levels, we use the subroutine arrage_energy(matrix,wavef).

- Subroutine arrange_energy(matrix,wavef): function that arranges the eigenvalues in ascending order. These values are inside matrix, which includes the energies and quantum numbers. All is ordered following the energy

ascendancy. Also, the ,matrix that includes all the wavefunctions (wavef) is also ordered.

- Subroutine simpson(N, dx, f, res) : subroutine to integrate the densities, in order to normalize the eigenfunctions. Inputs: N number of points in the mesh, dx space between the points in the mesh,f array to integrate. Output:res value of the integration.
- Function Vcoulomb (N, rho_p): function to calculate the Coulomb potential for protons. As an input the position in the mesh to calculate (N) and the proton density (rho_p).

2 Files

- global.f90: module that includes the global variables.
- initial_phi.f90: file that contains the WoodsSaxon3D subroutine to find the initial eigenvalues and wavefunctions.
- nucleon.txt: input file (see below).
- Coulomb.f90: file that includes the function Vcoulomb to calculate the value of the Coulomb potential.
- main.f90: main file that includes the code to find the eigenvalues and eigenfunctions in a self-consistent way.
- Makefile : makefile to compile the code (see below).
- simpson.f90: file that contains the subroutine that integrates using the Simpson method.
- arrange.f90: file that contains the subroutine that arranges the energy in ascending order for the Hartree-Fock eigenvalues.
- arrange2.f90: file that contains the subroutine that arranges the energy in ascending order for the WoodsSaxon3D subroutine.
- data_plot.plt: file to plot with gnuplot the proton and neutron densities.
- data_plot2.plt: file to plot with gnuplot the proton and neutron kinetic densities.

3 Input file—nucleon.txt

- proton, neutron = number of protons and neutron of the nucleus to study.
- h2m = value of $\hbar^2/2m$.
- R_{max} = size of our considered and studied system.
- N= number of points in the mesh.
- charge= not used

- epsi= epsilon error to find the solution.
- E_m= left value of the energy when finding the eigenvalues using the bissection method.
- E_p = right value of the energy when finding the eigenvalues using the bissection method.
- e2= value of e^2, value of the charge for the Coulomb potential.
- out_wave_func = logical parameter to output all the wavefunctions calculated in files or not.
- n_max, l_max = maximum n and l quantum numbers to calculate.
- r0 = parameter to calculate R0 in Wood-Saxon potential.
- a = difusivity of the Wood-Saxon potential.
- CMcorrection = logical parameter to indicate if CM correction is used.

4 Output

- Energy levels of neutrons (energy_level_n.dat) and protons (energy_level_p.dat).
- Density profiles of neutrons (density_n_new.dat) and protons(density_p_new.dat).
- If wanted, output with all wavefunctions calculated. (output_wave_func =.true.)
- Console: iteration, nucleon number, total energy of the system, single particle energy, kinetic energy, Skyrme energy, Coulomb energy.
- plots of the neutron and proton densities and kinetic densities.

5 Makefile

To run the program, there is a Makefile, that compiles all files that form the code. To compile, type **make**, and an executable **main** is created. To run the code, type **./main**. And to delete all exectuables and outputs, type either **make clean** or **make clear**. Makefile has to be modified with vi or vim programs.

6 Extras

6.1 Coulomb potential

Coulomb potential has been added, but the total energy does not give the correct value. **It is not correct. It has to be modified**.

6.2 BCS.

With the t0-t3 Skyrme code, we have added, in a different folder, the BCS calculation. New subroutines: solve_BCS_equation(L,E_data,N,E_tot, lambda, Delta, v): inputs: L: the number of shell E_data(1:L,1:4): 1:energy, 2:n, 3:angular momentum, 4:spin, N: particle number in BCS system output, E_tot: total energy in BCS system, lambda: chemical potential, Delta: pairing gap, v(1:L): occupation probability in BCS system.. Subroutine LU: subroutine that implements the LU decomposition to solve $df/dE * dE = -f$.

6.3 Full Skyrme

Not finished.