Numerical method to find the ground state of an Hamiltnian

Technical report for fortran numerical project

Paradis Enzo
Student at the university of Bourgogne Franche-Comté
Master CompuPhys - 1st year

Contents

I Introduction

In quantum mechanics the eigenstates of an Hamiltonian are really important, especially the ground state. This state is the one that is the most populated and it's teh satrting point for the study of light-matter interaction. This program goals to find this ground state for any Hamiltonian by partially diagonalise the Hamiltonian. We will try to use the power iterative method and the Lanczos algorithm. This program was written in Fortran 95 under manjaro. There are 3 f95 files. The first one is `main.f95`, wich is the main file. If you want to add a new Hamiltonian or choose a method to diagonalise one, it's in this file. The second file is `state.f95`, wich contains all the code for the power iterative method (in the subroutine `subroutine state`). You don't have to modify this file. And the third one is `lanczos.f95`, which contains the code for the Lanczos algorithm (in the subroutine `subroutine lanczos`). You don't have to modify this file but this part of the code don't work so you can try to improve it if you want. There is a file named `main` wich is the compile form of the program. You can find all these files on github.

II Functional requirement of the program

In the main file, you can find the `program pool`. In this program there is the creation of Hamiltonians that we want to diagonalize and the called of subroutines for each method. There are already 3 functions for Hamiltonians. There is also an "Hamiltonian" named `test`, wich looks like an Hamiltonian for tests. So after creating the Hamiltonian wanted you can use the `subroutine main`, wich create initial data and call the `subroutine state`, or the `subroutine lanczos` or both.

II.1 Hamiltonians

For this program we had created 3 functions for Hamiltonian. There are tables to describe them :

`function H_2lvl(omega, phi, delta)`		
Description	Input	Output
This function retrieve a Hamiltonian for a 2 level atom interacting with laser fields.	• `omega` : the laser amplitude multiplied by the atomic dipolar moment amplitude. • `phi` : the laser phase. • `delta` : the detuning.	This function will retrieve a 2x2 matrix.

Table 1: `function H_2lvl`

After using the `function molecular_H` you have to add to it the wanted potential to the kinetic Hamiltonian

`function H_3lvl(omega_p, phi_p, delta_p, omega_s, phi_s, delta_s)`		
Description	Input	Output
This function retrieve a Hamiltonian for a 3 level atom interacting with laser fields.	The "p" is for pump mode and "s" for Stokes mode. • `omega_p/omega_s` : the laser amplitude multiplied by the atomic dipolar moment amplitude. • `phi_p/phi_s` : the laser phase. • `delta_p/delta_s` : the detuning.	This function will retrieve a 3x3 matrix.

Table 2: `function H_2lvl`

`function molecular_H(N, L, m)`		
Description	Input	Output
This function retrieve a molecular kinetic Hamiltonian.	• `N` : number of particle (size of the Hamiltonian) • `L` : size of the molecule • `m` : mass of the molecule	This function will retrieve a NxN matrix, wich is diagonal.

Table 3: `function H_2lvl`

(all of them are commented under the example call). There are three different potatentials.

- Particle in a box (`function box_potential(N, L)`)

$$V(x) = \begin{cases} +\infty & if\, x = 0 \\ 0 & if\, x \in]0, L[\\ +\infty & if\, x = L \end{cases} \tag{II.1}$$

- Harmonic oscillator (`function ho_potential(N, L, k)`)

$$V(x) = \frac{1}{2}k\left(x - \frac{L}{2}\right)^2 \tag{II.2}$$

- Vibration of the H_2^+ molecule (`function molecular_potential(N, L, V0, a)`)

$$V(x) = V_0\left(\mathrm{e}^{-2a(x-x_0)} - 2\,\mathrm{e}^{a(x-x_0)}\right) \tag{II.3}$$

III Internal structure of the program

III.1 Description of the diagonalization algorithms

III.1.1 The power method algorithm

The power method algorithm consist to apply a matrix to a vector an infinite number of time. So to find the eigenvector we have :

$$|\phi_0\rangle = H^{\infty}|\psi\rangle \tag{III.1}$$

With H the matrix we want to diagonalize, $|\phi_0\rangle$ the eigenvector ans $|\psi\rangle$ a random vector. So the associated eigenvalue, λ_i, is $\lambda_i = \langle\phi_0|M|\phi_0\rangle$.
But we can't apply an infinite number of time a matrix to a vector. So we will transform a little the Equation III.1 and for the numerical calculation add a condition to stop before the infinity. We rewrite the Equation III.1 as :

$$\lim_{k\to+\infty} \frac{H^k|\psi\rangle}{||H^k|\psi\rangle||} = |\phi_0\rangle \tag{III.2}$$

In our case H is a Hamiltonian, so it is a hermitian matrix with a negative spectrum. And we want to find the ground state which is the lowest eigenvalue of H. The lowest eigenvalue is $\lambda_0 = \langle\phi_0|H|\phi_0\rangle$. So we will stop the computation when :

$$||H|\phi_0\rangle - \langle\phi_0|H|\phi_0\rangle|\phi_0\rangle|| > \epsilon \tag{III.3}$$

It is when the lowest eigenvalue it is a solution of the eigenequation for the ground state. With that we can write a pseudo-code to find the ground state.

---

take a random vector ϕ_0
normalize ϕ_0
$H \leftarrow H - shift * id$
while $||H\phi_0 - \langle\phi_0|H|\phi 0\rangle\phi_0|| > \epsilon$ and $k \leq k_{max}$ do
$\phi_0 \leftarrow H\phi_0$
normalize ϕ_0
$k \leftarrow k + 1$
end while
$H \leftarrow H + shift * id$
$\lambda_0 \leftarrow \langle\phi_0|H|\phi_0\rangle$

$shift$ is a shift value to ensure a negative spectrum.
By using this method we can also find the first exited state by projecting onto an ortogonal component to avoid non-zero component from ϕ_0 appearig into ϕ_1. So we can write the following pseudo-code :

---

take a random vector ϕ_1
$\phi_1 \leftarrow \phi_1 - \langle\phi_0|\phi_1\rangle\phi_0$
normalize ϕ_1
$H \leftarrow H - shift * id$
while $||H\phi_0 - \langle\phi_1|H|\phi 1\rangle\phi_1|| > \epsilon$ and $k \leq k_{max}$ do
$\phi_1 \leftarrow H\phi_1$
$\phi_1 \leftarrow \phi_1 - \langle\phi_0|\phi_1\rangle\phi_0$
normalize ϕ_1
$k \leftarrow k + 1$
end while
$H \leftarrow H + shift * id$
$\lambda_1 \leftarrow \langle\phi_1|H|\phi_1\rangle$
