Quantum Information and Computing (QIaC)

Prof. Simone Montangero

Tommaso Faorlin

Università degli Studi di Padova - Physics of Data

tommaso.faorlin@studenti.unipd.it

November 29, 2021

Time independent Schrödinger equation

The Hamiltonian operator of the **quantum harmonic oscillator** in one dimension is

$$\hat{H} = \frac{\hat{p}^2}{2m} + \frac{1}{2} m\omega^2 \hat{q}^2 \underset{m=\omega=1}{=} \frac{\hat{p}^2}{2} + \frac{\hat{q}^2}{2} \tag{1}$$

where m is the **mass** of the particle, and $\omega = \sqrt{k/m}$ the **angular frequency**. We work in **coordinate representation** $\psi_n(x) = \langle x|\psi_n\rangle$ and **natural units** $\hbar$=1. We want to solve the time-independent Schrödinger equation:

$$\hat{H}\,|\psi_n\rangle = E_n\,|\psi_n\rangle \rightarrow \frac{1}{2}\left(-\frac{d^2}{dx^2} + x^2\right)\psi_n(x) = E_n\psi_n(x) \tag{2}$$

- A class of solutions for the **eigenvectors** is that of Hermite functions:

$$\psi_n(x) = \frac{1}{\sqrt{2^n n!}}\left(\frac{m\omega}{\pi\hbar}\right)^{1/4} e^{-\frac{m\omega x^2}{2\hbar}} H_n\left(\frac{m\omega}{\hbar}x\right) \qquad n = 0, 1, 2, \ldots \tag{3}$$

- The energy levels corresponding to each $\psi_n(x)$, are specified by the **eigenvalues**:

$$E_n = \omega\left(n + \frac{1}{2}\right) \tag{4}$$

Code development

1. We discretized a space interval $[L_{min}, L_{max}]$, into N intervals of size $\Delta x = L/N$. We call $x_i = L_{min} + (ii - 1)\Delta x$ for $ii \in [1, N + 1]$ the i-th point of the discretized grid.
2. We use the **finite difference method** for the second derivative of the k-th eigenfunction:

$$\frac{\partial^2 \psi_k(x)}{\partial x^2} = \frac{\psi_{k+1} - 2\psi_k + \psi_{k-1}}{\Delta x^2} + \mathcal{O}(\Delta x^4) \tag{5}$$

   In this way we will end up with a **tridiagonal**, **real** and **symmetric** hamiltonian H^d

$$H^d_{ij} = H^d_{ji} = -\frac{\hbar^2}{2\Delta x^2} \quad H^d_{ii} = \frac{\hbar^2}{\Delta x^2} + \frac{1}{2} m\omega^2 {x_i}^2 \tag{6}$$

3. We **initialize** the hamiltonian operator $H^d_{ii,jj}$ for $ii, jj \in [1, N + 1]$

```
1      !filling the matrix
2              IF(ii.EQ.jj) THEN
3                  ham_op(ii,jj) = (h_bar**2)/(dx**2)
4              ELSE IF(ii.EQ.jj+1) THEN
5                  ham_op(ii,jj) = -(h_bar**2)/(2*dx**2)
6                  ham_op(jj,ii) = ham_op(ii,jj)
7      !adding the lattice-position dependent value
8      DO ii = 1, N+1
9          ham_op(ii,ii) = ham_op(ii,ii) + 0.5*m*omega**2*((Lmin+(ii-1)*dx)**2)
```

Code development

- The subroutine we exploit to diagonalize the matrix and find its eigenvalues λ_n and eigenfunctions is `DSTEV`. We implement it in the `diagonalize_H_op(ham_op, eigs)` subroutine inside the `qm_utils.f90` module.

```
1     !compute eigenvalues and eigenvectors
2     CALL DSTEV('V', N, diag_A, subd_A, ham_op, N, work, info)
```

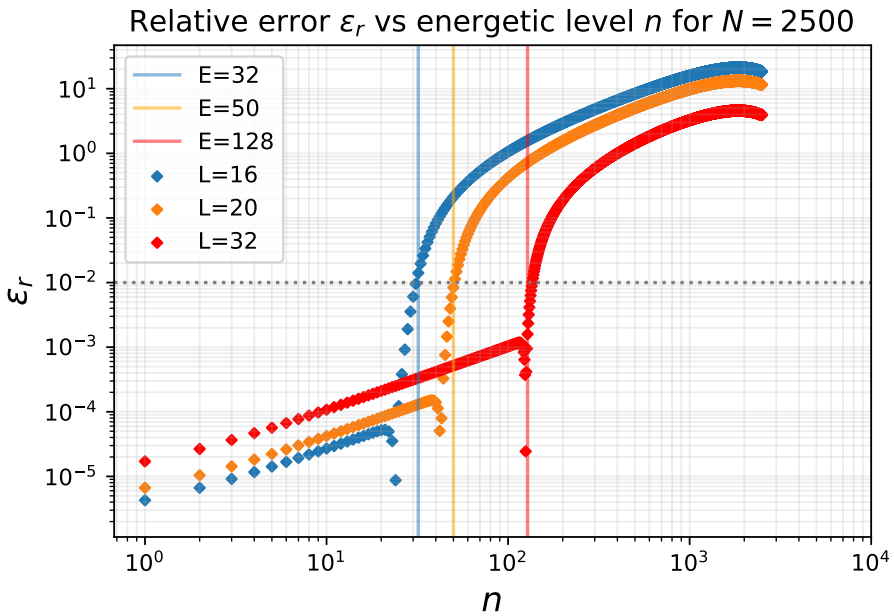
- We can run the code and specify simulation parameters:
  `./w4.out -custom 2500 -10 10 2500`
- We **format** output file name to increase code flexibility.

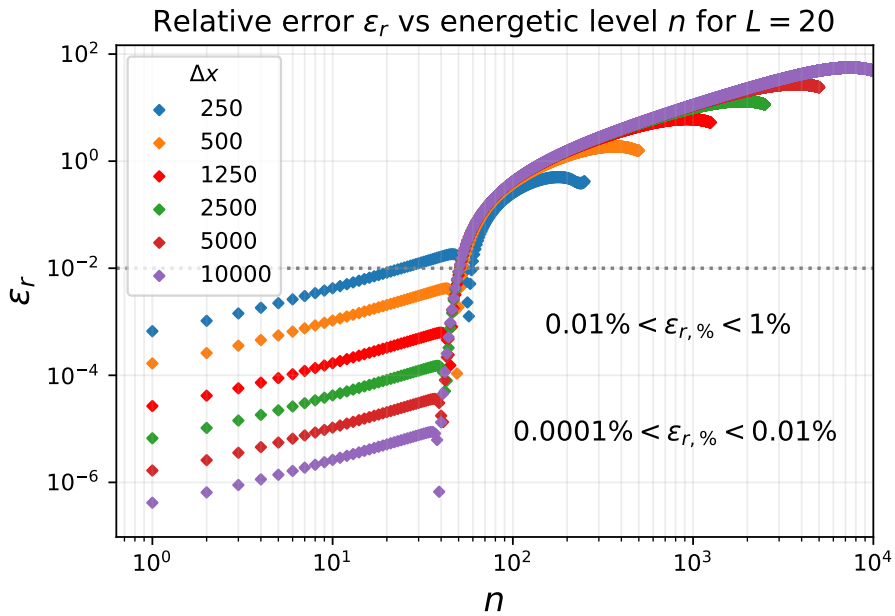
```
1  CHARACTER(len=1024) :: filename
2  ![...]
3  WRITE(filename, "(ai4af4.1af4.1a)") "data/eigenvalues_N", N,  "_Lmax", ...
```

Eigenvalues

We analyze the **relative error** between the numerical λ_n and theoretical eigenvalue λ_{th}

$$\varepsilon_r = \frac{\lambda_n - \lambda_{th}}{\lambda_{th}} \qquad \varepsilon_{r,\%} = 100 \cdot \varepsilon_r \tag{7}$$

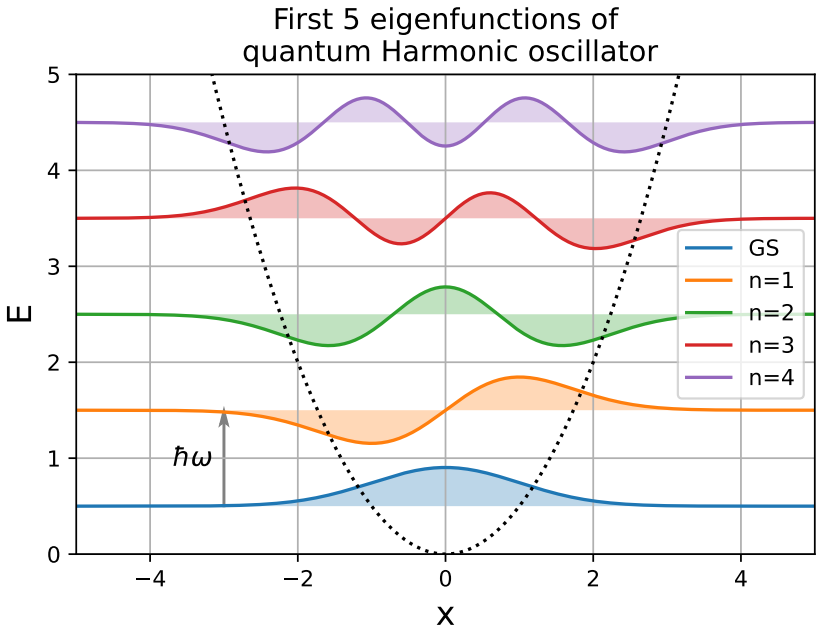




For a fixed number of discretization points $N = 2500$, we observe how by **increasing the width** of the space range we can describe systems with **higher energy**. We find that: $\lambda_{n,max} = L_{max}^2 \cdot 0.5$. For $\lambda > \lambda_{n,max}$ the problem becomes the *particle in a box*.

Once the **space range has been set** between $L_{min} = -10$ and $L_{max} = +10$, we can observe that as the number of discretization points increases (different colors), greater precision is achieved. For a fixed n: $\varepsilon_r \propto \mathcal{O}(\Delta x^2)$.

Considerations on the code



We report here the first 5 numerical eigenfunctions, with a spacing equal to $\hbar\omega = 1$ for the choice of the two parameters. The energetic levels E are the one computed in the simulation.

- **Correctness**. We have included some pre and post conditions. A special mention should be made of the one placed after the `DSTEV` subroutine, careful about the `INFO` variable ($\neq 0$ if errors are present).
- **Stability**. Variations in the input could result in inaccurate and nonphysical values, but not in code crashes (at least, up to now).
- **Accurate Discretization**. We can in principle achieve arbitrary accuracy up to the $k - th$ eigenvalue.
- **Flexibility**. The user can run the code with `-custom` flag and specify all the simulation parameters (m, ω, and the lattice quantities).
- **Efficiency**. The critical point of the simulation lies in the diagonalization of the matrix. By using the state-of-the-art `DSTEV` subroutine, we are able to calculate the eigenvalues and eigenfunctions of our system with good accuracy and in acceptable times. We didn't compute the algorithmic complexity.