# Quantum Information and Computing 2021-2022

$4^{th}$ week assignment

November 29, 2021

Campesan Giulia


Università
degli Studi
di Padova

**1-D quantum harmonic oscillator-Eigenvalues problem**

$$\hat{H}\,|\psi\rangle = \left[\frac{\hat{p}}{2m} + \frac{1}{2m}\omega^2 q^2\right]|\psi\rangle = E\,|\psi\rangle$$

- good approximation of any 1-D system around stable equilibrium point
- analytical solution
- **time independent Shrodinger equation** for harmonic potential

Representation: **coordinate basis**

$$\hat{q} \to \hat{x}, \quad \hat{p} \to -i\hbar\frac{\partial}{\partial\hat{x}}$$

**Natural units** for both energy and distance

$$\hbar\omega x \to x, \quad \sqrt{\frac{\hbar}{m\omega}}E \to E$$

$$\hat{H}\,|\psi\rangle = \left[\frac{-i}{2}\frac{\partial}{\partial\hat{x}^2} + \frac{1}{2}\omega^2 x^2\right]|\psi\rangle = E\,|\psi\rangle$$

- Eigenvalues $\mathbf{E_n} = \mathbf{n} + \frac{\mathbf{1}}{\mathbf{2}}, \quad n = 0, 1, 2, ...$
- Eigenvectors: $\psi_n(x) = \frac{1}{\sqrt{2^n n!}}\pi^{-\frac{1}{4}}\exp\left(-\frac{-x^2}{2}\right)H_n(x)$, where $H_n(x)$ are the Hermite polynomials

Computational representation: **discretization**

$$x \to x_k, \quad x_k = L1 + k * \Delta x$$

in which we are building a grid of $N+1$ points in the interval $[L1, L2]$, with a spacing

$$\Delta x = \frac{L2-L1}{N}$$

The discretized Shrodinger equation is then, using finite differences method:

$$[\hat{H}\,|\psi\rangle]_{x=x_k} = \frac{1}{2}\omega^2 x_k^2 + \frac{\psi\,(x = x_{k+1}) - 2\psi\,(x = x_k) + \psi\,(x = x_{k-1})}{\Delta x^2} + O(\Delta x^2)$$

The representation of the system Hamiltonian in the computational basis ends up to be **tridiagonal** and **symmetric**:

$$H_{i,i+1} = H_{i,i-1} = -\frac{1}{2\Delta x^2}, \qquad H_{i,i} = \frac{1}{2\Delta x^2} + \frac{1}{2}\omega^2 x_i^2$$

$$\mathrm{H}= \begin{bmatrix} \frac{1}{2\Delta x^2} + \frac{1}{2}\omega^2 x_1^2 & -\frac{1}{2\Delta x^2} & & & \\ -\frac{1}{2\Delta x^2} & \frac{1}{2\Delta x^2} + \frac{1}{2}\omega^2 x_2^2 & -\frac{1}{2\Delta x^2} & & \\ \vdots & -\frac{1}{2\Delta x^2} & \frac{1}{2\Delta x^2} + \frac{1}{2}\omega^2 x_3^2 & -\frac{1}{2\Delta x^2} & \\ & & & \ddots & \\ & 0 & & -\frac{1}{2\Delta x^2} & \frac{1}{2\Delta x^2} + \frac{1}{2}\omega^2 x_{N+1}^2 \end{bmatrix}$$

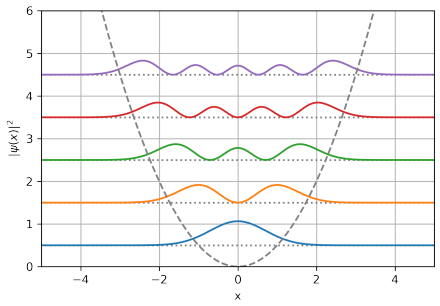
The compilation and execution of the Fortran script and module exploited are controlled by running:

*python execution.py L1 L2 dx*

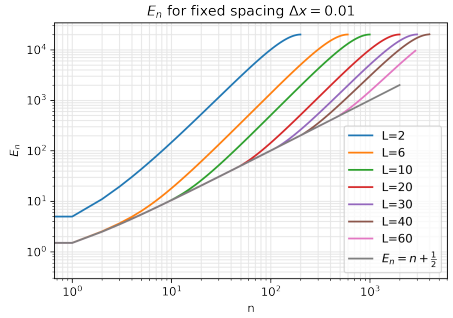
where L1, L2 are the range extrema and dx is the discretization spacing.
In the *Hermitian.f90* we exploit the DSTEV fortran subroutine for real, symmetric, tridiagonal matrices diagonalization to solve our eigenvalues problem.

```
N = size(h, 1)
LDZ = MAX(1, N)

allocate(eigvalues(N)); allocate(UD(N-1)); allocate(WORK(MAX(1, 2*N-2))); allocate(eigvectors(LDZ, N))

do ii=1, N
    eigvalues(ii) = h(ii, ii)
enddo

do ii=1, N-1
    UD(ii) = h(ii, ii+1)
enddo

call DSTEV( 'V', N, eigvalues, UD, eigvectors, LDZ, WORK, INFO )

call check( condition= (INFO.NE.0), &
            msg_type = 'ERROR', &
            msg = 'something went wrong in dstev')
```

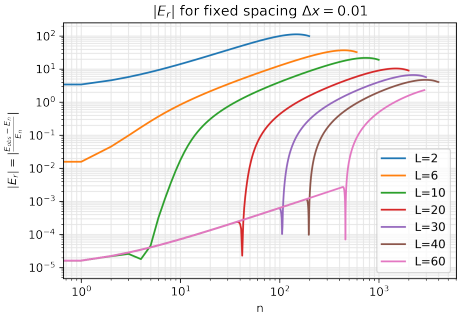
Figure: DSTEV subroutine call



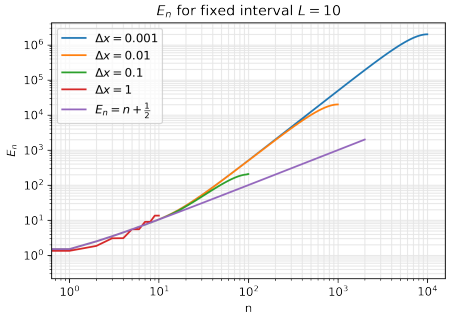

(a)



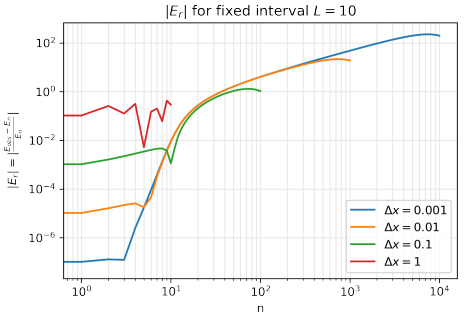

(b)




(c)




(d)




(e)

Figure: Eigenvalues and eigenvectors (2a). Eigenvalues (2b) and their relative error (2c) for fixed spacing $\Delta x$ and varying L. Eigenvalues (2d) and their relative error (2e) for fixed spacing $\Delta x$ and varying L

- **Correctness**
  The curves reporting the absolute value of the relative error keeping respectively $\Delta x$ and $L$ fixed are shown in the slide before. We can see a strong dependence both on the spacing (finite differences method related error) and both on the interval range (higher energies eigenvalues see the potential as infinite-well shaped)
- **Stability**
  The program returns the same results (up to finite precision) for different runs
- **Discretization**
  There is a strong dependence on the chosen discretization spacing.
- **Flexibility**
  The program responds well to different input values of both $\Delta x$ and $L$
- **Efficiency**
  The program exploits the ad-hoc STDEV subroutine for real, tridiagonal, symmetric matrices. Still, there is a strong dependence of the execution time on the number of points in the lattice.