Report: Week 4

Quantum Information and Computing (2021/22)
Prof. Simone Montangero

Samuele Piccinelli

Università degli Studi di Padova

30 November 2021

Continuous time-indipendent Schrödinger equation

We consider the **one-dimensional quantum harmonic oscillator**, defined by the Hamiltonian

$$H\psi_n(x) = \left(-\frac{1}{2m}\frac{\mathrm{d}^2}{\mathrm{d}x^2} + \frac{1}{2}m\omega^2x^2\right)\psi_n(x) = \mathcal{E}_n\psi_n(x) \qquad (n \in \mathbb{N}).$$

We make use of natural units ($\hbar = 1$) and set $m = 1$.

- the **eigenvectors** $\psi_n(x)$ are related to the **Hermite functions** $H_n(x)$,

$$\psi_n(x) = \frac{1}{\sqrt{2^n n!}} \cdot \left(\frac{\omega}{\pi}\right)^{\frac{1}{4}} \cdot \exp\left(-\frac{\omega x^2}{2}\right) \cdot H_n(\sqrt{\omega}x);$$

- the **eigenvalues** $\mathcal{E}_n = \left(n + \frac{1}{2}\right)\omega$ are the corresponding **energy levels**.

Numerical solution

Given a **finite** and **symmetric** interval $[-x_{\max}, x_{\max}]$, $x_{\max} \geq 0$ we can discretize it in N intervals of width Δx,

$$\Delta x = \frac{2 \cdot x_{\max}}{N} \qquad\qquad x_i = -x_{\max} + i\Delta x,\ i = 0, \dots, N.$$

We apply the **finite differences method** to the second derivative of $\psi(x)$,

$$\frac{\mathrm{d}^2\psi}{\mathrm{d}x^2} = \frac{\psi(x_{i+1}) - 2\psi(x_i) + \psi(x_{i-1})}{(\Delta x)^2} + \mathcal{O}(x^4).$$

TASK: find eigenvalues and eigenvectors of the discretized Hamiltonian H represented by the following **tri-diagonal matrix**:

$$H_{ij} = \begin{cases} (\Delta x)^{-2} + \frac{\omega^2 x_i^2}{2} & \text{if } i = j \\ -1/2 & \text{if } i = j \pm 1 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

We implement the **oscillator** module to **initialize** and **diagonalize** H.

User input: **N**, **x_max**, **omega** ⟶ H initialization ⟶ dstev subroutine

```fortran
delta_x = (2 * ABS(x_max)) / (N-1)
H = 0.d0

do ii=1,N
    H(ii,ii) = (1.0d0/(delta_x**2)) + (0.5d0*omega**2) *\
               (-ABS(x_max) + delta_x*(ii-1))**2
end do

do ii=1,N-1
    H(ii,ii+1) = - (1.0d0/(2.0d0*delta_x**2))
    H(ii+1,ii) = H(ii,ii+1)
end do
```

Listing 1: Implementation of the initialization of the discretized Hamiltonian as in Eq. (1).

```fortran
lwork = max(1, 2*N-2)

allocate(work(lwork))
call DSTEV('V', N, D, E, H, N, work, info)

deallocate(work)
eigs = D
```

Listing 2: Implementation of the `dstev` subroutine.

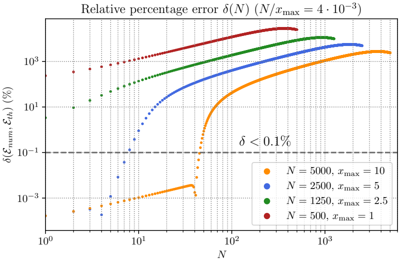
We compare the first N eigenvalues with the theoretical expectations.
The **perentage relative error** between the numerical $\mathcal{E}_n^{\text{num}}$ and theoretical $\mathcal{E}_n^{\text{th}}$ value is computed for $0 < n \leq N$,

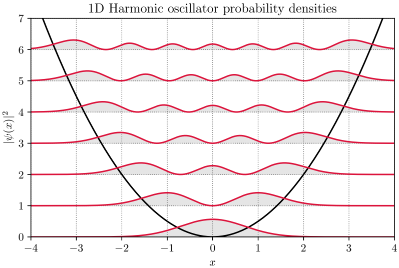
$$\delta_n = \frac{\left|\mathcal{E}_n^{\text{num}} - \mathcal{E}_n^{\text{th}}\right|}{\mathcal{E}_n^{\text{th}}} \%.$$

For a fixed **resolution** $\Delta x = N/x_{\max}$, the **accuracy** increases for

- larger ranges $L = 2 \cdot x_{\max}$;
- higher angular frequencies ω.

The numerical solution has to take **physical boundary effects** into account.





Self-evaluation

- **Correctness**. The code produces results in accordance with the theoretical expectations, but is limited to the adopted **discretization** and **range**.
- **Stability**. The code returns the same result for the same parameters and has been run for different values of N, $x_{\max}$, ω.
- **Accurate discretization**. The discretization can be improved by modifying the input parameters to obtain n eigenvalues under a fixed threshold δ_n.
- **Flexibility**. The program can be adapted to different values of the parameters: the next step would be to consider a $m \neq 1$ mass particle.
- **Efficiency**. We exploit the fact that H is tridiagonal to employ adapted **Lapack routines**.