Assignment 5

Quantum Information and Computing Course 2022/2023

Massimo Colombo

Department of Physics and Astronomy "Galileo Galilei"
DEGREE IN PHYSICS

December 10, 2022

Time Dependent Schrodinger Equation: Theory

The assignment's request is to solve the dependent Schrödinger equation with the given Hamiltonian:

$$\hat{H} = \frac{\hat{p}^2}{2m} + \frac{\omega^2(\hat{q} - \frac{t}{T})^2}{2m} \qquad t \in [0, T]$$

$$i\partial_t |\psi(x,t)\rangle = \hat{H} |\psi(x,t)\rangle \qquad |\psi(x,0)\rangle = \left(\frac{1}{\pi}\right)^{\frac{1}{4}} e^{-\frac{x^2}{2}} \qquad \text{with} \quad \hbar = \omega = m = 1$$

where, as the Initial state, the 1st eigenvector of the quantum harmonic oscillator is considered.
To deal with it numerically, we can consider a short time interval Δt where we can assume the Hamiltonian to be constant.
So, using the SPLIT OPERATOR METHOD, the time evolution of wavefunction results:

$$\begin{aligned} |\psi(x, t + \Delta t)\rangle &\simeq e^{-i\hat{V}\Delta t/2} e^{-i\hat{T}\Delta t} e^{-i\hat{V}\Delta t/2} |\psi(x,t)\rangle \\ &= U_V(x) \mathscr{F}^{-1} \mathscr{F} U_T(x) \mathscr{F}^{-1} \mathscr{F} U_V(x) |\psi(x,t)\rangle \end{aligned}$$

Where T and V are the kinetic and potential terms of the Hamiltonian H,
and the approximation is correct up to an error of order $\mathscr{O}(\Delta t^3)$
Discretizing the time interval of the system, it is then possible to iterate this procedure,
evolving the initial state from the time value 0 to the final T.

Code Development

The code requires LAPACK's and fftw3's routines. In the compilation those libraries have to be added:

```
massimocolombo$ gfortran Ex1-ColomboMassimo.f90 -o Ex1.out -llapack -lfftw3 -L/usr/local/lib
```

CODE STEPS:

DISCRETIZATION → DEFINING INITIAL STATE → EVOLUTION OPERATOR ALGORITHM → RESULTS

DISCRETIZATION: a "1 dim. lattice" type has been defined to simplify the procedure.

```
type lattice1
real*8, allocatable, dimension(:) :: grid
real*8, dimension(2) :: inf_sup
real*8 step
integer*4 :: step_amount
end type

function init_lattice(inf_sup, step_amount) result(lat)
    type(lattice1) lat
    lat%inf_sup = inf_sup
    lat%step_amount = step_amount
    lat%step = (inf_sup(2) - inf_sup(1)) / step_amount
    ALLOCATE(lat%grid(step_amount+1))
    DO ii = 1, step_amount + 1
        lat%grid(ii) = inf_sup(1) + (ii - 1)*lat%step
    END DO
```

DEFINING INITIAL STATE: the 1st eigenstate of the quantum harmonic oscillator is considered as initial state. Its "shape" is assigned to a derived type: *wavefunction*. that can be properly normalized thanks to the subroutine *normalize*().

```
type wave_function
complex*16, allocatable, dimension(:) :: comp
real*8                                :: norm
integer*4                             :: rank
end type
```

```
ALLOCATE(eigenvec%comp(dim))
eigenvec%comp = 0
DO ii = 1, dim
    eigenvec%comp(ii) = (exp(-(omega * x_lat%grid(ii)**2)/2))
END DO
norm_eigvec = normalize(eigenvec, x_lat%step)
```

Code Development 2

EVOLUTION OPERATOR ALGORITHM: Translating the theory into the code, we get that kinetic and potential evolution terms can be written as vectors (each being diagonal in the proper space). Moreover, they act on the state through the scalar product.
fftw3's routines take care of the Fourier transformations.

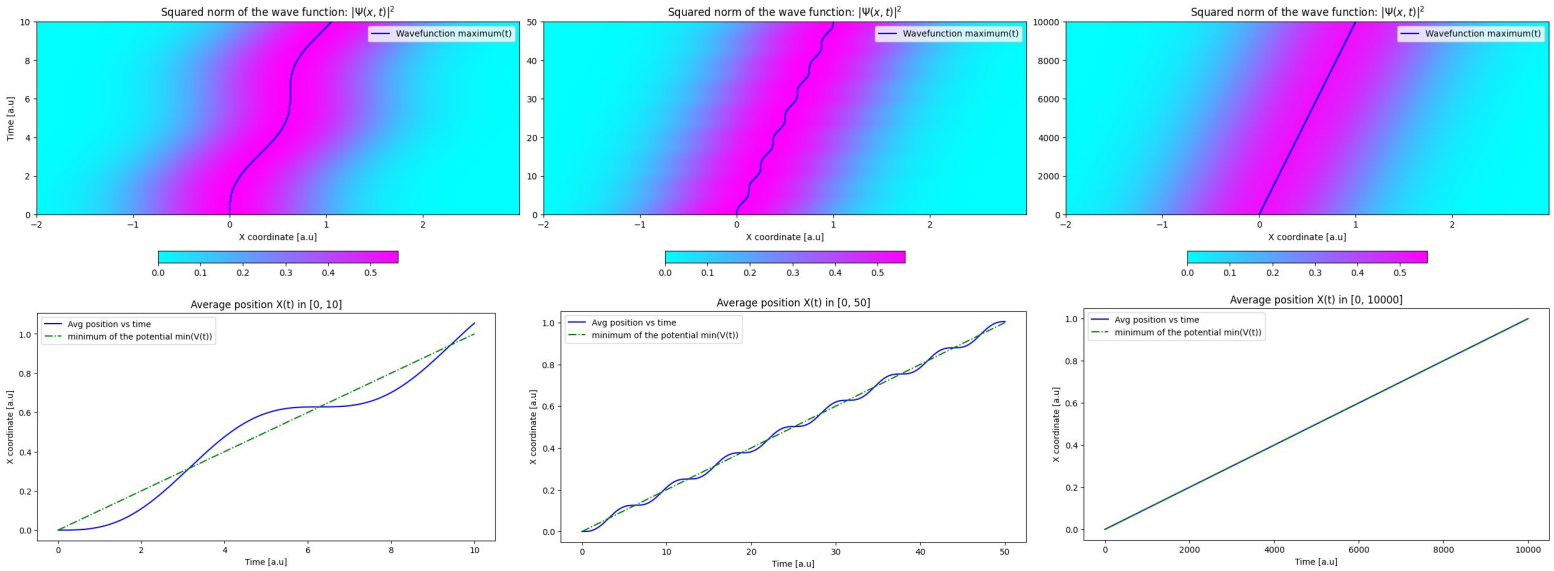
```
DO jj = 1,  t_lat%step_amount +1
    !First Potential term evolution.
        DO ii = 1,  dim
            Upot(jj, ii) = exp(- (i_math  * t_lat%step * omega**2 * (x_lat%grid(ii) - q0(jj))**2) /4 )
            wave_coor1%comp(ii) = Upot(jj, ii) * norm_eigvec%comp(ii)
        END DO
    !First fourier tfm. (from coordinate basis to momentum basis).
        CALL dfftw_plan_dft_1d(plan, dim, wave_coor1%comp, wave_mom1%comp, -1, 64)
        CALL dfftw_execute_dft(plan, wave_coor1%comp, wave_mom1%comp)
        CALL dfftw_destroy_plan(plan)
    !Kinetic term evolution
        DO ii = 1, dim
            Ukin(ii) = exp(- (i_math  * t_lat%step * p_arr(ii)**2 /2))
            wave_mom2%comp(ii) = Ukin(ii) * wave_mom1%comp(ii)
        END DO
    !Second fourier tfm. (from momentum basis to coordinate basis).
        CALL dfftw_plan_dft_1d(plan, dim, wave_mom2%comp, wave_coor2%comp, +1, 64)
        CALL dfftw_execute_dft(plan, wave_mom2%comp, wave_coor2%comp)
        CALL dfftw_destroy_plan(plan)
    !Second potential term evolution.
        DO ii = 1,  dim
            norm_eigvec%comp(ii) = Upot(jj, ii) * wave_coor2%comp(ii)
        END DO
        norm_eigvec = normalize(norm_eigvec, x_lat%step)
```

REMARK fftw3's Fourier transformation saves the momentum coefficients into an array in a **specific** order: first, the values associated with the momentum in $[0, \pi \setminus \Delta x)$, then those relative to the momentum in $[-\pi \setminus \Delta x, 0)$, where Δx is the coordinate-grid step. So, the kinetic evolution term has been created following the same order.

Results

The code allows saving the wavefunction and its squared value in .txt files. It was possible to analyze program results with different inputs through a python script.

Fixing the x range as [-10, 10]* and discretizing both x and time coordinate with 2000 points each, the value of T has been varied within values from 5 to 10000. T = 10, 50, 10000 plot are presented in the following.

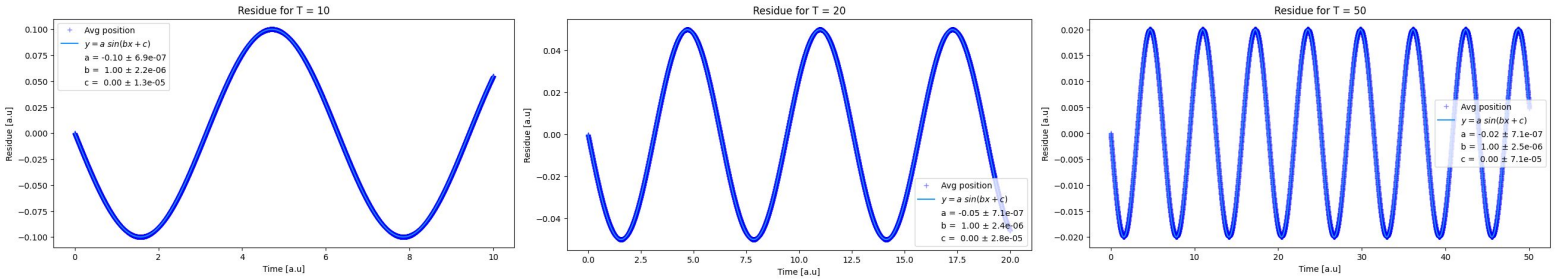


The first row of plots shows in color the value of the squared wavefunction and its maximum depending on the position and time. The second row shows the average position of the wavefunction and the minimum of the potential depending on time.
It is interesting to observe that the average position oscillates around its minimal potential coordinate with an amplitude that decreases and the number of oscillations that increases with increasing T.
The first result can be explained by noticing that a higher T means that the system changes slower, so we expect less “excitation”.
For instance, in T = 10000 plot, it becomes hard to notice oscillations.
The second result is just the consequence of having more time to observe oscillations.

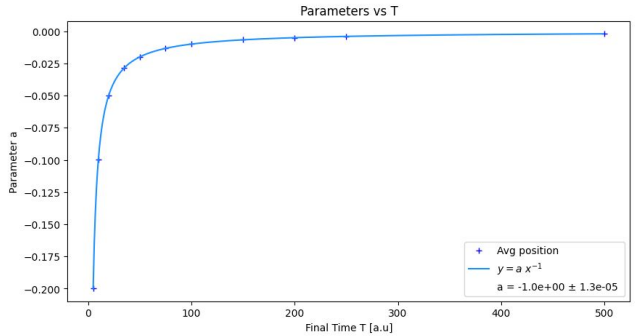
*This range is wider than the interesting interval [0, 1] to prevent computational issues close to borders.

Results 2

It is interesting to analyze the residue between the average position plot and the minimal potential coordinate:



In each plot, the shape of sin(t) appears. Moreover, its amplitude seems to be the reciprocal value of the final time T.
Fitting each plot with a sine function, y = a sin(bt + c), we get for each value of T: b = 1 ± O(10^(-5)) and c = 0 ± O(10^(-5)).
Meanwhile, the amplitude **a** shows the following T-dependence:



As expected amplitude parameters show an inverse proportionality with respect to T with coefficient -1. This allows us to predict that the average position of the solution of the T.D.S.E for the given Hamiltonian will be approximated by:

$$\overline{X}(t) = \frac{t}{T} - \frac{sin(t)}{T}$$