User-manual for TDSE numerical solver

Code by - MD ELIOUS ALI MONDAL

Supervisor - Dr. VARADHARAJAN SRINIVASAN
Associate Professor, IISER BHOPAL

Contents

1 Introduction

This code can be used to find the numerical solutions to both Time-independent as well as time-dependent Schrodinger equations for 1-D potentials. The code is divided into 7 modules and a run_file (script file) and each of these files can be individually modified as the user wants. The code is executed in the following order:

- **time_steps**
- **Global_constants** (Defining values which remain constant throughout the code)
- **Potential** (defining the time-dependent (or independent) potential)
- **TISE_solver** (Solver for time-independent schrodinger equation)
- **AB**
- **PSI_generation**(Generating $\Psi(x, t_i)$ for each t_i)
- **coefficients**
- **run_file**

2 Modules

2.1 time_steps

It requires three parameters:

- **ntp** - number of time periods
- **k** - gap between each time step
- **omega** - The frequency of oscillation of time-dependent potential

In cases in which the time-dependent potential is not oscillatory in time omega is to be set to 0.
When **time_steps.t_val(ntp,k,omega)** is called, it generates an array of all the time steps at which the wavefunction is to be evaluated

2.2 Global_constants

This contains all the constants that remain constant throughout the code.

The following are the constant parameters:

- **h** = step size for x-coordinate values
- **k** = step size for time-coordinate values
- **x_range** = range of x-coordinate values
- **n** = number of x-coordinates to be included
- **lam** = λ value for A and B matrices
- **omega** = The frequency of oscillation of time-dependent potential

Some arrays are also defined here:

- **x** = array to store the x-coordinate values
- **y** = array to store $\Psi(x_i)$ at each x_i
- **t_val** = array to store all time-coordinate values

2.3 Potential

Here we define the time-dependent potential **V(x,t)**. Here **x** implies position and **t** implies time. The constants dist, epsilon and gamma are given according to the potential as discussed in report. If a different potential is to be studied change **b** to the potential of interest.

2.4 TISE_solver

This module is used to solve the time-dependent schrodinger equation to obtain the stationary states and corresponding eigenvalues at any time t_i.

2.4.1 Numerov

This function applies Numerov method to obtain Ψ_{i+1} by **y1** (Ψ_{i-1}), **y2** (Ψ_i) (Here $\Psi_i \equiv \Psi(x_i)$) and energy **E**.

2.4.2 Psi

This function applies Numerov function (above) to evaluate $\Psi(x)$ for a given **E**. It returns a tuple **(m[-1],m)**:

- **m[-1]** = Value of Ψ corresponding to the last element of x.
- **m** = The array containing Ψ values for each x-value.

2.4.3 Eigen

This function calculates and returns an array containing all the possible eigenvalues within a given range of interest.

- **a** = array containing values within which the eigenvalues is to be searched.
- **P** = array containing **m[-1]** for all Ψ corresponding to each E_{trial} value in **a**.

2.4.4 Psi_gs

This is a dictionary which stores the ground state wavefunction (Ψ_{gs}) corresponding to each time step t_i.

2.5 AB

This module is used to generate the matrices **A** and **B** and to carry out the **LU** decomposition of **A**. The functions **a(i,j)** and **b(i,j)** are used to generate the **A** and **B** matrix elements. The dictionaries **A_val** and **B_val** are used to store the **A** and **B** matrices corresponding to each time step t_i. The dictionaries **L_val** and **U_val** store the **LU** decomposed matrices of **A** for each time step t_i

2.6 PSI_generation

This module generates $\Psi(t_{i+1})$ from $\Psi(t_i)$ by solving the time-dependent Schroedinger equation through Crank-Nicolson method as discussed in report. **PSI_t** is a dictionary used to store $\Psi(t_i)$ at each time step.

2.7 coefficients

Two dictionaries are used as storage here:

- **Coeff** = stores the coefficients ($c(t)$, contribution of $\Psi_{gs}(t_i)$ to the actual $\Psi(t_i)$).
- **Coeff2** = stores $c(t)c(t)^* = |c(t)|^2$ for each time step

3 How to use the code?

The **run_file** is the execution file which runs all the above module in the same sequence as described above.The parameters can be changed from Global_constants module according to the requirement. **C2_val** stores the same values as **Coeff2**. We can plot $|c(t)|^2$ vs **t** by un-commenting the region below as written in code.