

On Spectral Analysis of Time-Dependent Schrödinger Equation

A PH4401 Quantum Mechanics III project

JEREMY LIAN ANG KO HONG

Division of Physics & Applied Physics
School of Physical & Mathematical Sciences
Nanyang Technological University

May 2019

Abstract

Classic computational methods for determining the eigenvalues and eigenfunctions of the time-independent Schrödinger equation involves either a diagonalization of a Hamiltonian matrix or an iterative numerical integration of a 2^{nd} order partial differential equation. In this report, an approach using spectral analysis of solutions to time-**dependent** Schrödinger equation is discussed. A correlation function $\langle \psi(\mathbf{r}, 0) | \psi(\mathbf{r}, t) \rangle$ from a numerical solution $\psi(\mathbf{r}, t)$ is first computed and Fourier analysis of this correlation function then gives a set of resonant peaks that correspond to the stationary states of the system. Lineshape fitting analysis of the location of these peaks further reveals the eigenvalues. Subsequent Fourier transforms of $\psi(\mathbf{r}, t)$ with respect to time will generate the respective eigenfunctions. The effectiveness of the method is demonstrated for a one-dimensional quantum harmonic oscillator.

Contents

1	Motivation	1
2	Adventures with Fourier Transform	2
3	On correlation function and eigenvalue determination	3
4	Escapades in known territory: the case of a simple harmonic oscillator	4
5	Conclusion	7

1 Motivation

Consider the simple time-independent Schrödinger equation (TISE):

$$\hat{\mathbf{H}}|\psi(\mathbf{r})\rangle = \mathbf{E}|\psi(\mathbf{r})\rangle \quad (1)$$

Conventional methods to solve such eigenvalue equation involve diagonalization of the Hamiltonian matrix \mathbf{H} . In the position representation, the wavefunction $\psi(\mathbf{r}, t)$ is decomposed into a linear combination of a finite set of basis functions. These orthonormal basis functions are chosen for specific properties such as symmetry and spatial behaviours. The diagonalization of the Hamiltonian matrix constructed from these basis functions will produce the desired eigenvalues and eigenfunctions[1].

Matrix diagonalization is limited in its effectiveness to TISE consisting of Hamiltonians that do not differ greatly from those with known eigenfunctions and have potentials with simple analytic forms – allowing the Hamiltonian matrix to be constructed without excessive computation. This method becomes less effective when a large amount of computation is required to set up the Hamiltonian matrix, i.e. the required Hamiltonian matrix is of a high order.

The TISE from Eq. (1) can be rewritten in the position representation as a 2^{nd} order partial differential

equation:

$$\left[-\frac{\hbar^2}{2m} \nabla^2 + V(\vec{r}) \right] \psi(\vec{r}) = E\psi(\vec{r}) \quad (2)$$

which, in 1-D, will take the form:

$$\left[-\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x) \right] \psi(x) = E\psi(x) \quad (3)$$

A different approach to solving for the eigenvalues (energies) involves repeated numerical integrations of Eq. (3), sometimes known as shooting method[1]. Typically, an initial estimate for the individual eigenvalue is approximated and the corresponding eigenfunction is computed by numerical integration. Knowledge of the eigenfunction enables the computation of an improved eigenvalue, which provides the input to the next iteration and this process repeats until a desired accuracy for the eigenvalue is achieved.

Despite the awkward trial-and-error approach, iterative numerical integration gives very accurate results for 1-D TISE. To successfully apply this method, it is necessary for the initial estimate of each eigenvalue to be close to the exact value before the cycle of iterations is begun, both to reduce the number of iterations and to assure that no individual eigenvalues are missed. Algorithm efficiency gets reduced when information on the eigenvalue spectrum is limited. There is also no practical iterative numerical integration scheme developed for TISE in higher dimensions.

Spectral analysis of numerical solutions to time-**dependent** Schrödinger equation (TDSE) allows one to solve the Schrödinger eigenvalue problem[2]. The spectral method is applicable to any linear eigenvalue problem involving any number of dimensions. The only requirement is an available numerical solution for the time-dependent wavefunction specified on a suitable coordinate grid. The split operator FFT method provides highly accurate numerical solutions to TDSE and is thus, particularly attractive for application with the spectral method to the Schrödinger equation in Cartesian coordinates.

2 Adventures with Fourier Transform

Now, let us consider the time-dependent Schrödinger equation (TDSE) for a time-independent Hamiltonian in position representation:

$$i\hbar \frac{\partial}{\partial t} \psi(\vec{r}, t) = \hat{\mathbf{H}} \psi(\vec{r}, t) \quad (4)$$

The wavefunction has a general solution:

$$\psi(\vec{r}, t) = \exp\left[-\frac{i}{\hbar} H(t - t_0)\right] \psi(\vec{r}, t_0) \quad (5)$$

Taking infinitesimal timestep dt , the wavefunction at time $t_0 + dt$ becomes

$$\begin{aligned} \psi(\vec{r}, t_0 + dt) &= \exp\left[-\frac{i}{\hbar} \hat{\mathbf{H}} dt\right] \psi(\vec{r}, t_0) \\ &= \exp\left[-\frac{i(\hat{\mathbf{H}}_k + \hat{\mathbf{H}}_r) dt}{\hbar}\right] \psi(\vec{r}, t_0) \end{aligned} \quad (6)$$

where we have separated the Hamiltonian into its momentum and position space components.

$\hat{\mathbf{H}}_k$ and $\hat{\mathbf{H}}_r$ do not commute, i.e.

$$\exp\left[-\frac{i(\hat{\mathbf{H}}_k + \hat{\mathbf{H}}_r) dt}{\hbar}\right] \neq \exp\left(-\frac{i\hat{\mathbf{H}}_k dt}{\hbar}\right) \exp\left(-\frac{i\hat{\mathbf{H}}_r dt}{\hbar}\right) \quad (7)$$

Fortunately, by expanding $\exp\left[-\frac{i(\hat{\mathbf{H}}_k + \hat{\mathbf{H}}_r) dt}{\hbar}\right]$ in a Taylor series while rearranging and keeping the terms to second order, one can show that:

$$\exp\left[-\frac{i(\hat{\mathbf{H}}_k + \hat{\mathbf{H}}_r) dt}{\hbar}\right] = \exp\left(-\frac{i\hat{\mathbf{H}}_k dt}{2\hbar}\right) \exp\left(-\frac{i\hat{\mathbf{H}}_r dt}{\hbar}\right) \exp\left(-\frac{i\hat{\mathbf{H}}_k dt}{2\hbar}\right) + \mathcal{O}(dt^3) \quad (8)$$

It turns out that the error of order dt^3 comes from the noncommutation relations between each component of the Hamiltonian with $[\hat{\mathbf{H}}_k, \hat{\mathbf{H}}_r]$. This can be shown explicitly if one applies the Baker–Campbell–Hausdorff formula in Eq. (8).

In position representation, $\hat{\mathbf{H}}_k = -\frac{\hbar^2}{2m}\nabla^2$ and $\hat{\mathbf{H}}_r = V(\mathbf{r})$. One can thus rewrite Eq. (8) as:

$$\exp\left[-\frac{i(\frac{\hbar^2}{2m}\nabla^2 - V(\mathbf{r}))dt}{\hbar}\right] = \exp\left(\frac{i\hbar dt}{4m}\nabla^2\right)\exp\left(-\frac{idtV(\mathbf{r})}{\hbar}\right)\exp\left(\frac{i\hbar dt}{4m}\nabla^2\right) + \mathcal{O}(dt^3) \quad (9)$$

Let us concern ourselves with a 2-D wavefunction in cartesian coordinates $\psi(x, y, t)$. Extension to higher dimensions can be made without loss of generality. It is easy to see that

$$\exp\left(\frac{i\hbar dt}{4m}\nabla^2\right)\psi(x, y, t_0) \quad (10)$$

is the solution to the free particle TDSE

$$i\hbar\frac{\partial}{\partial t}\psi(x, y, t) = -\frac{\hbar^2}{2m}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\psi(x, y, t) \quad (11)$$

over time $dt/2$ with the initial wavefunction at time $t = t_0$.

Using the results of Eq. (9), the right-hand-side of expression (6) is thus equivalent to a free particle propagation over a half-time increment, followed by a phase change from the action of the potential applied over a whole-time increment, and an additional free particle propagation over another half-time increment. If many factors of the form (9) are applied in sequence, corresponding to a series of timesteps dt s making up t , pairs of half-step free particle propagations combine into full-step propagations. The computation then proceeds as a succession of full-step propagations, applied in momentum space, alternating with phase changes of the wave function executed in position space. The only exceptions are the half-step propagations applied at the beginning and end of the calculation.

The awkwardly-named split operator FFT method becomes clear in its application if we consider Eq. (8) with Eq. (6). We note that the operator $\hat{\mathbf{H}}_k$ matrix is diagonal in momentum space while the $\hat{\mathbf{H}}_r$ matrix is diagonal in position space. Eigenvalues can then be found without the need for matrix diagonalization if we ensure both operators are acting in spaces where their matrices are diagonal. This can be achieved with the split operator FFT, using Fourier transformations to switch between position and momentum spaces. The wavefunction solution to the TDSE in Eq. (4) can thus be found for a timestep dt :

$$\psi(\mathbf{r}, t_0 + dt) = \mathcal{F}^{-1}\left[\exp\left(-\frac{i\hat{\mathbf{H}}_k dt}{2\hbar}\right)\mathcal{F}\left[\exp\left(-\frac{i\hat{\mathbf{H}}_r dt}{\hbar}\right)\mathcal{F}^{-1}\left[\exp\left(-\frac{i\hat{\mathbf{H}}_k dt}{2\hbar}\right)\mathcal{F}\psi(\mathbf{r}, t_0)\right]\right]\right] \quad (12)$$

where \mathcal{F} represents Fourier transformation from position space to momentum space and \mathcal{F}^{-1} represents inverse Fourier transformation from momentum space to position space. Generalising this solution for time t can then be done by breaking down t into a series of timesteps dt .

3 On correlation function and eigenvalue determination

One is able to show that solution to the TDSE in Eq. (4) can be written as a linear superposition of eigenfunctions:

$$\psi(x, y, t) = \sum_{n,j} A_{n,j} u_{n,j}(x, y) \exp(-iE_n t) \quad (13)$$

where the index j denotes states within a degenerate set and $u_{n,j}(x, y)$ satisfies

$$-\frac{\hbar^2}{2m}\nabla^2 u_{n,j} + V u_{n,j} = E_n u_{n,j} \quad (14)$$

The correlation function $\mathcal{C}(t)$ is then defined as

$$\begin{aligned}\mathcal{C}(t) &= \langle \psi(x, y, 0) | \psi(x, y, t) \rangle \\ &= \iint \psi^*(x, y, 0) \psi(x, y, t) dx dy \\ &= \sum_{n,j} |A_{nj}|^2 \exp(-iE_n t)\end{aligned}\tag{15}$$

where we have made use of the expression from Eq. (13) in arriving at the last line.

We sample the correlation function by its multiplication with a normalised Hanning window function $w(t)$:

$$w(t) = \frac{1 - \cos(2\pi t/T)}{T}\tag{16}$$

Doing so allows us to account for the fact that only a finite length of time is observed (from experiments/computations). It turns out that employing this window function greatly reduces the amplitudes of “sidelobes” that may be confused with or overlap other resonance peaks when Fourier transformation is applied to the correlation function later on.

Taking the Fourier transform of the sampled function would yield:

$$\tilde{\mathcal{C}}(E) = \sum_n W_n \mathcal{L}(E - E_n)\tag{17}$$

where $W_n = \sum_j |A_{nj}|^2$ and lineshape function $\mathcal{L}(E - E_n)$ is

$$\begin{aligned}\mathcal{L}(E - E_n) &= \frac{1}{T} \int_0^T \exp[i(E - E_n)t] w(t) dt \\ &= -\frac{1}{2} \left[\frac{\exp(i[(E - E_n)T + 2\pi]) - 1}{i[(E - E_n)T + 2\pi]} + \frac{\exp(i[(E - E_n)T - 2\pi]) - 1}{i[(E - E_n)T - 2\pi]} \right] \\ &\quad + \frac{\exp[i(E - E_n)T] - 1}{i(E - E_n)T}\end{aligned}\tag{18}$$

It is easy to see that the eigen-energy values for a TISE can now be determined from the TDSE “analogue”. The correlation function is first computed numerically for each propagation step, followed by evaluating the Fourier transform $\tilde{\mathcal{C}}(E)$ from the sampled value of each correlation function. Finally, the data set for each $\tilde{\mathcal{C}}(E)$ is fitted to the form in Eq. (17) by a least-squares computational routine, yielding an energy spectrum from which the eigen-energy value appear as a peak in this spectrum.

4 Escapades in known territory: the case of a simple harmonic oscillator

We begin the demonstration of the robustness of this spectral technique with a simple quantum system which eigen-energy values can be calculated analytically.

Consider the 1-D quantum harmonic oscillator, whose eigen-energy values can be easily determined to be:

$$E_n = \left(n + \frac{1}{2}\right) \hbar \omega\tag{19}$$

where $n = 0, 1, 2 \dots$ represents each discrete energy state.

The potential of this system has the typical harmonic form:

$$V(x) = \frac{1}{2} m \omega^2 x^2\tag{20}$$

where ω is the angular frequency of the classical oscillator related to the spring constant k in Hooke’s law via $\omega = \sqrt{\frac{k}{m}}$.

We adopt computational units where $\hbar = m = \omega = 1$ and set up our computation with 1000 spatial “grid” points and 1000 temporal steps. We then implement the potential function using `python` in the following code:

```

1 # Simple Harmonic Oscillator (SHO) potential
2 def V(x):
3     return 0.5 * x**2

```

Listing 1: Defining the potential function

The spectral technique is neat in its application because the eigen-energy values can be determined from any arbitrary initial wavefunction. All we require is the time evolution of this wavefunction under the desired potential “barrier”. With this, we define a fully arbitrary gaussian wave that approaches 0 at $|x| \rightarrow \infty$.

```

1 # Wavefunction at t = 0
2 def psi_0(x):
3     psi = ( exp(-1j*((x-4.)/2)) * exp(-1j*((x-2.)/2)) * exp(-1j*((x+1.)/2)) * exp(-1
           ↪ j*((x+3.)/2)) ) * (exp(-((x/sqrt(5))**2))/2 * sqrt(2*pi*5))
4     return psi

```

Listing 2: Defining the initial wavefunction

Next, we implement the split step operator FFT routine as laid out in Section 2, and evolve the initial wavefunction for 1000s.

```

1 # Split step operator FFT
2 def split_step(time, dt, N, L):
3     dx = L/N
4     x = linspace(-L/2, L/2-dx, N)
5     psi = psi_0(x)
6
7     # Output array
8     t_out = zeros(time + 1)
9     psi_out = zeros((time + 1, N), dtype=complex)
10    t_out[0], psi_out[0] = 0, copy(psi)
11
12    # Potential operator
13    def potential_step(psi):
14        psi = exp(-1j * dt * V(x)) * psi
15        return psi
16
17    # For the kinetic step
18    n = arange(N) # oscillatory factor
19    p = (-1)**n
20
21    def kinetic_step(psi):
22        # Kinetic operator
23        expK = exp(-0.25j * dt * (-pi/dx + n * 2*pi/L)**2)
24
25        psi = p * ifft(expK * fft(psi * p))
26        return psi
27
28    # split step run
29    for i in range(time):
30        psi = kinetic_step(psi)
31        psi = potential_step(psi)
32        psi = kinetic_step(psi)
33
34        t_out[i+1] = i * dt
35        psi_out[i+1] = copy(psi)
36

```

```

37     return x, t_out, psi_out
38
39 # Evolving the wavefunction
40 x, t, psi = split_step(time, dt, N, L)

```

Listing 3: Split step operator FFT routine

We proceed to numerical calculation of the correlation function and sample it with a normalised Hanning window, before Fourier transforming the resulting function.

```

1 # Hanning window function
2 def win_fn(tstep, time_arr):
3     return 1 - cos(2*pi*time_arr[tstep]/time_arr[-1])
4
5 # Correlation function
6 t = delete(t,0)
7 corr_arr = zeros(len(t), dtype = complex)
8 for s in range(len(t)):
9     corr = conj(psi[0]) * psi[s]
10    corr_arr[s] = trapz(corr,dx=L/N)
11    corr_arr[s] *= win_fn(s, t)/t[-1]
12
13 # Fourier transform of correlation function
14 fcorr_arr = fft(corr_arr)

```

Listing 4: Correlation function in Fourier space

In the endgame now ¹, we fit the results from the Fourier transformation to the form in Eq. (17), employing a least-squares routine from `scipy.optimize` package.

```

1 # Line shape fitting
2 def Line(E, W, En):
3     a = 1j * (E - En) * time
4     b = (exp(a)-1)/a + 0.5*((exp(a+2*pi)-1)/(a+2*pi)) + 0.5*((exp(a-2*pi)-1)/(a-2*pi
5         ↪ ))
6     return W*b
7
8 # Line fit for Real and Imag parts
9 def LineBoth(x, W, En):
10    n = len(x)
11    x_real = x[:n//2]
12    x_imag = x[n//2:]
13    y_real = real(Line(x_real, W, En))
14    y_imag = imag(Line(x_imag, W, En))
15    return hstack([y_real, y_imag])
16
17 # Data for the fitting
18 freq_arr = linspace(0,2*pi/dt, time)
19 yReal = fcorr_arr.real
20 yImag = fcorr_arr.imag
21 xdata = hstack([freq_arr, freq_arr])
22 ydata = hstack([yReal, yImag])
23
24 # Curve fitting according to no. of peaks
25 def BestFit(x, W1,W2,W3,W4,W5,W6, E1,E2,E3,E4,E5,E6):
26     fn = LineBoth(x, W1, E1)+LineBoth(x, W2, E2)+LineBoth(x, W3, E3)+LineBoth(x, W4,
27         ↪ E4)+LineBoth(x, W5, E5)+LineBoth(x, W6, E6)
28     return fn

```

¹Explicit Avengers reference ☺, in addition to the very obvious peppering of “spectral analysis” and “eigenvalues” throughout this article. Unfortunately, the authors of this paper are doubtful an “inverted” Mobius strip holds the key to time travelling as perpetuated by Tony Stark.

```

28 | errfunc = lambda p, x, y: (BestFit(x, *p)-y)**2
29 | guess = [.5,.5,.5,.5,.5,.5,.5,1.5,2.5,3.5,4.5,5.5]
30 |
31 | optim, success = leastsq(errfunc, guess[:], args=(xdata,ydata))

```

Listing 5: Peak function fitting

The resulting energy spectrum is displayed in Figure 1 below, together with a graph of the initial wavefunction and the harmonic potential. The energy eigenvalues show up as peaks in the graph, whose positions 0.5, 1.5, 2.5, 3.5, 4.5... are exactly as we would expect of the energy levels of a quantum harmonic oscillator in Eq. (19).

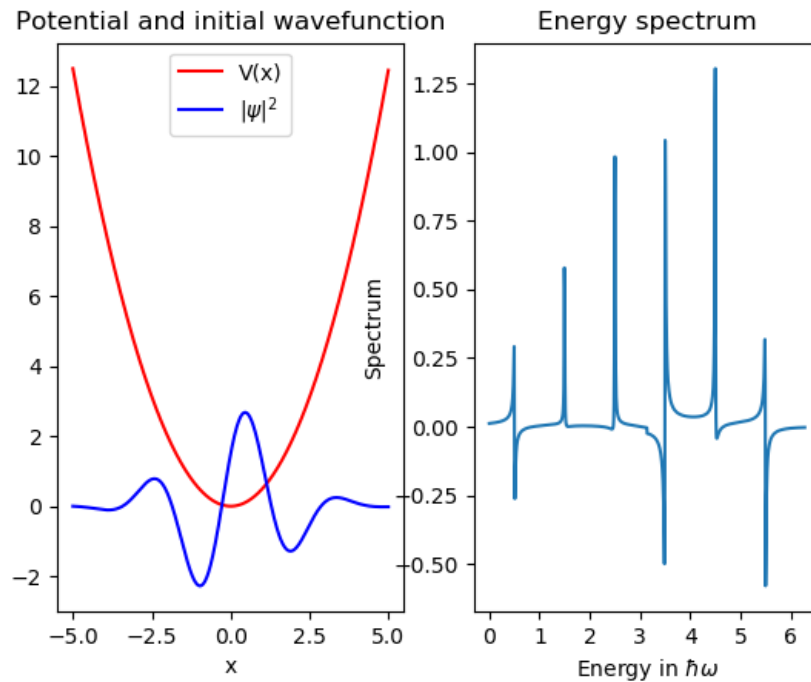


Figure 1: Graphs from Spectral method implementation

5 Conclusion

An attractive feature of the spectral analysis method presented in this paper[2] is the graphical representation of the energy values as a spectrum and eigen-energy levels can be identified visually as with an experimentally determined spectrum. The beauty of the spectral method lies in its simple, straightforward application. There is no special selection of basis functions or a requirement for the potential to have a unique analytical form. The only concern for accuracy is the specification of adequate sampling rates in space and time for the numerical solution of a time-evolved wavefunction. Ultimately, the spectral method provides a more robust alternative to matrix diagonalization and iterative numerical integration methods.

References

- [1] Daniel V. Schroeder. Numerical solutions of the tise, 2015. URL <https://physics.weber.edu/schroeder/quantum/Numerical.pdf>.
- [2] M. D. FEIT, J. A. FLECK JR., and A. STEIGER. Solution of the schrodinger equation by a spectral method. *JOURNAL OF COMPUTATIONAL PHYSICS*, 47:412–433, feb 1982.