Quantum Kicked Rotor

Aditya Chincholi

May 13, 2021

1 Introduction

2 Questions

1. What does kicking the rotor periodically have anything to do with a random walk?

2. What is the analogy with Anderson localization? After all, Anderson localisation is about a diffusing wavefunction which encounters disorder in the form of passive scatterers and gets reflected/transmitted with certain probability. This transmission amplitude goes down exponentially with length of the sample. In contrast, in the kicked rotor we have an initial condition of uniform distribution in the position space. We have active "kicks" which pump energy into the system and these kicks strengths are pseudo-random apparently.

3. If we take a quantum rotor and kick it with a truly random kick strength like Ka_j at $t = j$ where $a_j = 1$ with probability p and $a_j = -1$ with probability $1 - p$, would it also show initial diffusion and subsequent localisation?

4. In the Anderson localisation (at least for d=1 case), we see the "transport" being matter transport i.e. we comment on the chances that the matter particle is transmitted across the sample. One could also say that the energy of the particle (which is constant) gets transported to position states far from the origin. But in the case of the kicked rotor, what exactly is being transported? Energy is being pumped into and taken out of this system at all levels, so what is being transported?

5. In the simulation, given a particular dimension of the fourier space of $|\psi\rangle$, how do we get a bound on the maximum timestep?

3 Partial Answers

1. Let us try and answer the question 3. First consider a hamiltonian given by:

$$H = \frac{L^2}{2} + \hbar K \sum_{j=0}^{\infty} a_j \delta(t-j) \tag{1}$$

where $P(a_j = 1) = p, P(a_j = -1) = 1 - p$. We can take $p = \frac{1}{2}$ for simplicity. We then have the unitary operator U_j to evolve the state from after the (j-1)th kick to after the jth kick.

$$U_j = exp(-iKa_j)exp(-i\frac{L^2}{2\hbar}) \tag{2}$$

$$U_j |m\rangle = exp(-i(Ka_j + \frac{\hbar m^2}{2})) |m\rangle \tag{3}$$

where $|m\rangle$ is the eigenstate of angular momentum operator L. We can see clearly here that this random "kick" actually does nothing. It doesn't project our system from one angular momentum eigenstate to another. So this is just a phase shift of each existing eigenstate. No new $|m\rangle$ states can be occupied which weren't occupied before. Clearly, the disorder $\leftrightarrow$ random kick strength analogy fails in this respect. Lets try the following general hamiltonian and find the problem.

$$H = \frac{L^2}{2} + \hbar K V(\theta) \sum_{j=0}^{\infty} a_j \delta(t-j) \tag{4}$$

Then we get

$$U_j = exp(-iKa_jV(\theta))exp(-i\frac{L^2}{2\hbar}) \tag{5}$$

$$U_j \left|m\right\rangle = \frac{e^{-i\hbar m^2/2}}{\sqrt{2\pi}} \int exp(-iKa_jV(\theta))exp(-im\theta) \left|\theta\right\rangle d\theta \tag{6}$$

$$= \frac{e^{-i\hbar m^2/2}}{\sqrt{2\pi}} \int exp(-i(Ka_j\frac{V(\theta)}{\theta} + m)\theta) \left|\theta\right\rangle d\theta \tag{7}$$

$$[\text{And if we take } V(\theta) = \theta] = e^{-i\hbar m^2/2} \left|m + Ka_j\right\rangle \tag{8}$$

So clearly, $V(\theta)$ needs at least a θ term in order to kick the system into other states. The problem is that without theta dependence, the initial m term will never break into pieces which hop to other states. ~~The issue is one cannot think of the state hopping from $|m\rangle$ to $|n\rangle$, rather one must look at it from the θ space perspective.~~

2. A naive answer to question 5 might be to do the following energy calculation:

$$Tk = \frac{1}{2}\hbar^2 L_{max}^2 \tag{9}$$

$$T \leq \frac{\hbar^2 L_{max}^2}{2k} \tag{10}$$

$$[\text{If we take } \hbar = 1, L_{max} = 1000, k = 5]$$

$$T \leq 10^5 \tag{11}$$

which undoubtedly seems like an overestimate.

4 Week 7

Deadline: Next Saturday (10 April 2021)

4.1 Objectives

1. Work out the a_j model analytically and computationally both.
2. Add noise to kick period in the kicked rotor: Kick the rotor at $\tau \pm \delta\tau$ where $\delta\tau$ is drawn from a uniform distribution. Loss of localization is expected.

3. Add noise to kick strength in the kicked rotor: Kick the rotor with strength $k \pm \delta k$ where δk is from uniform distribution. Loss of localization is expected.

4. Read up on the quasi-periodic kicked rotor and the metal-insulator transition in it.

5 Week 8

Deadline: Saturday 17 April 2021 3:30 PM

5.1 Problems with the results

1. In order to get the same results as them, we need to run till $t = 10^6$ steps, and for at least 1000 initial conditions of (ϕ_2, ϕ_3). That comes to 10^9 evolutions, which is a lot. And for $t = 10^6$, we'd have to account for diffusion in momentum space by taking as high a dimensionality for our simulation space. Time required for each time step goes over 2s per step for as low as 2001 dimensional space. For context, the run with 1000 timesteps, 201 dimensional space and 100 initial conditions of (ϕ_2, ϕ_3) took around 2 hrs.

2.

6 Week 9

6.1 Driving Perturbation

Consider the following Hamiltonian:

$$H = \frac{p^2}{2} + K(1 + \epsilon cos(\omega_2 t + \phi_2) cos(\omega_3 t + \phi_3)) \sum_{n \in \mathbb{Z}} \delta(t - n) + \lambda F_0 cos(\Omega t) \tag{12}$$

Switching to the 3d rotor with quasimomenta. (13)

$$H = \frac{p_1^2}{2} + p_2\omega_2 + p_3\omega_3 + K(1 + \epsilon\ cos\theta_2\ cos\theta_3) \sum_{n \in \mathbb{Z}} \delta(t - n) + \lambda F_0 cos(\Omega t) \tag{14}$$

The time evolution from $n-1$ to $n - \Delta t$ is:

$$U_{\Delta t} = exp(\frac{-i}{\hbar} \int_{n-1}^{n-\Delta t} dt H) \tag{15}$$

$$= exp(\frac{-i}{\hbar} \int_{n-1}^{n-\Delta t} dt (H_0 + \lambda H')) \tag{16}$$

$$= e^{-i(p_1^2/2 + p_2\omega_2 + p_3\omega_3)(1-\Delta t)/\hbar}\ e^{\frac{-i\lambda F_0}{\hbar\Omega}(sin(\Omega(n-\Delta t)) - sin(\Omega(n-1)))} \tag{17}$$

$$F = e^{-ik(1+\epsilon cos\theta_2 cos\theta_3)/\hbar} \lim_{\Delta t \to 0} U_{\Delta t} \tag{18}$$

$$F = e^{-ik(1+\epsilon cos(\omega_2 n + \phi_2) cos(\omega_3 n + \phi_3))/\hbar}\ e^{-ip^2/2\hbar}\ e^{-i\lambda F_0 (sin(n\Omega) - sin((n-1)\Omega))/\hbar\Omega} \tag{19}$$

~~This is under the approximation that the hamiltonian H is dominated by the kick term during the interval $n - \Delta t$ to n. That implies that $|cos(n\Omega)\Delta t| << k(1 - \epsilon)$ where Δt is the width of the kick pulse in a physical setting.~~

7 Week 10

7.1 Goals

- To find out a way to calculate the bipartite entanglement between the $|p_1\rangle$ states and the $|p_2, p_3\rangle$ states in the quasiperiodic kicked rotor.
- To better express the phase driving perturbation from last time.
- To try out the driving term as a perturbation to the kicked rotor.

7.2 Bipartite Entanglement Calculation

- First of all, we want to stick to simulating the 1d quasiperiodic kicked rotor. In order to simulate the 3d form, our floquet operator would require $O(N^6)$ entries where N is the size of the momentum spaces along each angle. This is not feasible for accuracy in a simulation. There is a possibility, however, that we use sparse matrices and reduce the size of the stored operator as it need only be calculated once but the density operator will also be of the same size and that is not sparse.

- The first issue I tried to tackle was to find a good way to represent the F_{3d} operator. However, that involved a fourier transform of the following which I have no clue how to solve. The 1d case came from an vague Bessel function relation which I couldn't really apply here. In any case, the $\theta's$ would definitely get mixed up in the final result as entanglement is happening so there's no straightforward way to do this:

$$\frac{1}{(2\pi)^3}\int d^3\theta\ e^{-ikcos(\theta_1)[1+\epsilon cos(\theta_2)cos(\theta_3)]/\hbar}e^{-i(\mathbf{m}-\mathbf{n})\cdot\boldsymbol{\theta}} \tag{20}$$

  So I resorted to numerically evaluating it. It took a while to understand the discrete fourier transform code, but I wrote it.

- The main theoretical considerations I tried was trying to get a relation between the 1d rotor and the 3d rotor. But despite trying to manipulate the kets for a long time, I couldn't really do much. I tried taking the trace of F_{3d} and that did yield a relation to F_{1d} but the bipartite entanglement requires $\rho_1 = tr_{2,3}(\rho)$ which cannot be reduced to just the trace of F_{3d}.

$$tr_{2,3}(F_{3d}) = \sum_{m_1,n_1}\sum_{m_2,m_3}|m_1\rangle\langle n_1|\, e^{-i(m_2\omega_2+m_3\omega_3)}\int\int\frac{d\theta_2 d\theta_3}{(2\pi)^2}(F_{1d})_{nm}) \tag{21}$$

- Last night's work:

$$\mathbb{P}_{\varphi_2+\omega_2,\varphi_3+\omega_3}F_{3d}\mathbb{P}_{\varphi_2,\varphi_3} = F_{1d}\otimes\mathbb{I}\otimes\mathbb{I} \tag{22}$$

8 Week 11

8.1 Goals

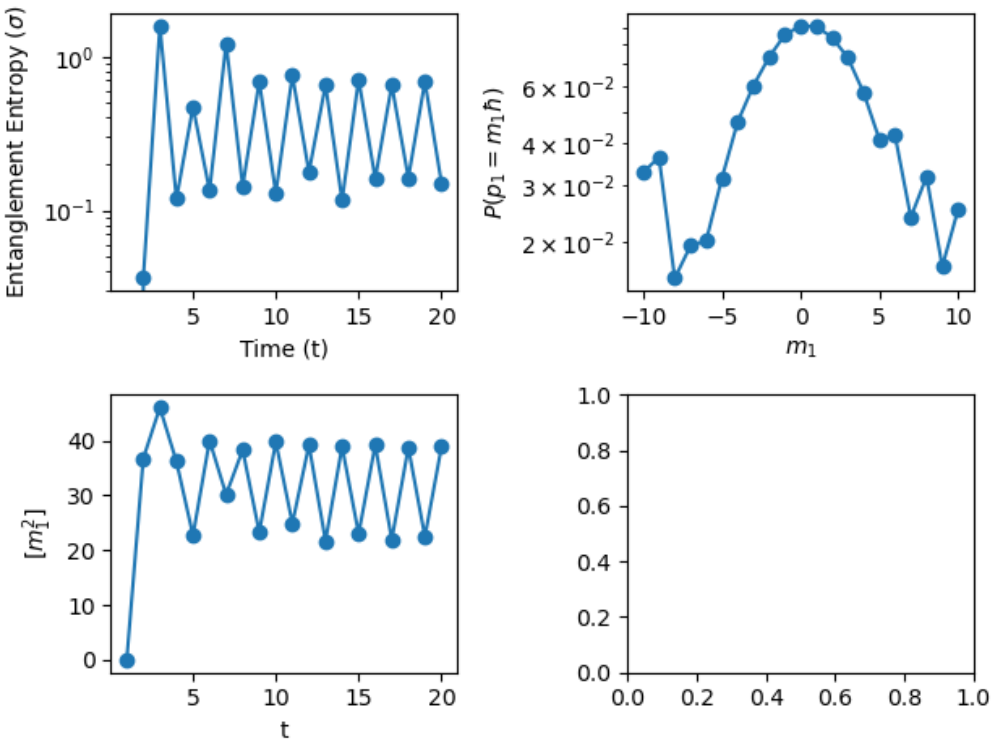
- Program and calculate the bipartite entanglement by simulating the 3d kicked rotor.
- Read and understand the paper on "Multifractility ..." and compare with my results.
- Read and work out the critical point dynamics from Muller, Delande's paper.

8.2 Floquet Expression Derivation

$$
\begin{aligned}
F_{3d} &= e^{-iKcos\theta_1(1+\alpha cos\theta_2 cos\theta_3)/\hbar} e^{-i(p_1^2/2+p_2\omega_2+p_3\omega_3)/\hbar} \\
\langle \mathbf{m}| F_{3d} |\mathbf{n}\rangle &= \langle m_1, m_2, m_3| F_{3d} |n_1, n_2, n_3\rangle \\
&= \langle \mathbf{m}| e^{-iKcos\theta_1(1+\alpha cos\theta_2 cos\theta_3)/\hbar} |\mathbf{n}\rangle e^{-i(\hbar n_1^2/2+n_2\omega_2+n_3\omega_3)} \\
&= \langle \mathbf{m}| \int_{[0,2\pi]^3} d^3\theta \; e^{-iKcos\theta_1(1+\alpha cos\theta_2 cos\theta_3)/\hbar} |\boldsymbol{\theta}\rangle \langle \boldsymbol{\theta}|\mathbf{n}\rangle e^{-i(\hbar n_1^2/2+n_2\omega_2+n_3\omega_3)} \\
&= \int_{[0,2\pi]^3} d^3\theta \; \langle \mathbf{m}|\boldsymbol{\theta}\rangle e^{-iKcos\theta_1(1+\alpha cos\theta_2 cos\theta_3)/\hbar} \langle \boldsymbol{\theta}|\mathbf{n}\rangle e^{-i(\hbar n_1^2/2+n_2\omega_2+n_3\omega_3)} \\
&= e^{-i(\hbar n_1^2/2+n_2\omega_2+n_3\omega_3)} \frac{1}{(2\pi)^3} \int_{[0,2\pi]^3} d^3\theta \; e^{-i(\mathbf{m}-\mathbf{n})\cdot\boldsymbol{\theta}} e^{-iKcos\theta_1(1+\alpha cos\theta_2 cos\theta_3)/\hbar} \\
&\approx e^{-i(\hbar n_1^2/2+n_2\omega_2+n_3\omega_3)} DFT_N(f(\boldsymbol{\theta}))[\mathbf{m}-\mathbf{n}] \\
\text{where } f(\boldsymbol{\theta}) &= e^{-iKcos\theta_1(1+\alpha cos\theta_2 cos\theta_3)/\hbar}
\end{aligned}
$$

9 Bipartite Entanglement Simulation

9.1 Results



1. Bipartite entanglement entropy between the two subspaces $\mathcal{H}_1$ and $\mathcal{H}_{2,3}$ is non-zero and is roughly $O(10^{-1})$. It seems to saturate or at least oscillate about a mean. But there are issues.

2. The momentum p_1 distribution looks sort of correct. But it is clear that we need more basis states based on the rise at the edges of the distribution.

3. The momentum square p_1^2 is increasing properly like a diffusive regime and then saturating out due to lack of basis states.

9.2 Issues

1. First of all, the matrix is not unitary. It's eigenvalue spectrum is not composed of phases. For a $21^3 \times 21^3$ case, 1331 eigenvalues were calculated for the matrix and they had a decent amount of spread. Calculating the determinant, through logdet, also yields very small quantities $e^{-O(100)}$.

   But this is just with the dense form of the matrix. In order to decrease memory consumption, we use sparse matrices. So we zero out any terms with absolute value less than 10^{-6}. Should be a reasonable thing to do. Apparently not, because the eigenvalues amplitudes shot up to 10-12 when I tried finding eigenvalues of the sparse matrix. But I could not repeat this computation. Every time I tried again, ARPACK (used by scipy) said all the eigenvectors failed to converge.

   It is notable that this is happening despite various tests telling me that the floquet operator expression is correct. I have run tests wherein one of the elements is computed through quad instead of the fft and it matches it to a good degree.

2. The other issue, due to the first one, is that the trace of the density matrix is not conserved to be unity. For lower dimensions, the difference is very small, it goes from 0.99 to 0.97 to 0.89 to 0.6. In this case, one can possibly justify a simple correction to divide ρ by the $tr(\rho)$ or by $tr(\rho_1)$. But this is harder for higher dimensions because the fluctuations tend to increase. I have to correct by dividing by $tr(\rho)$ here, $tr(\rho_1)$ also fails here.

3. The calculation of the floquet matrix has been parallelized, and it's sparse form takes relatively low amount of space in memory. But the density matrices are, well, dense. They cannot be sparse. So the evolution takes place with a Sparse $\times$ Dense $\times$ Sparse dot product operation. But sparse linalg is single-threaded. And that's the case in pretty much all libraries available across major languages (MATLAB included). So the evolution is slow. And by slow I mean quite slow. It took over an hour to do 20 timesteps in the 21 basis case.

4. I cannot afford to use dense forms of the floquet operator on my system. The system already uses 3.7 GB of RAM and 2 GB of swap space on my

system with 21 basis states for each subspace. When I tried increasing this to 24 basis states per subspace, the program segfaulted. So having a dense floquet operator is not an option on my system at least.

I have also tried using a smaller datatype (complex32 over complex64), but that started giving me underflow errors and NaNs. I tried mitigating them, but eventually I had to move to the larger datatype as the processing became very problematic.

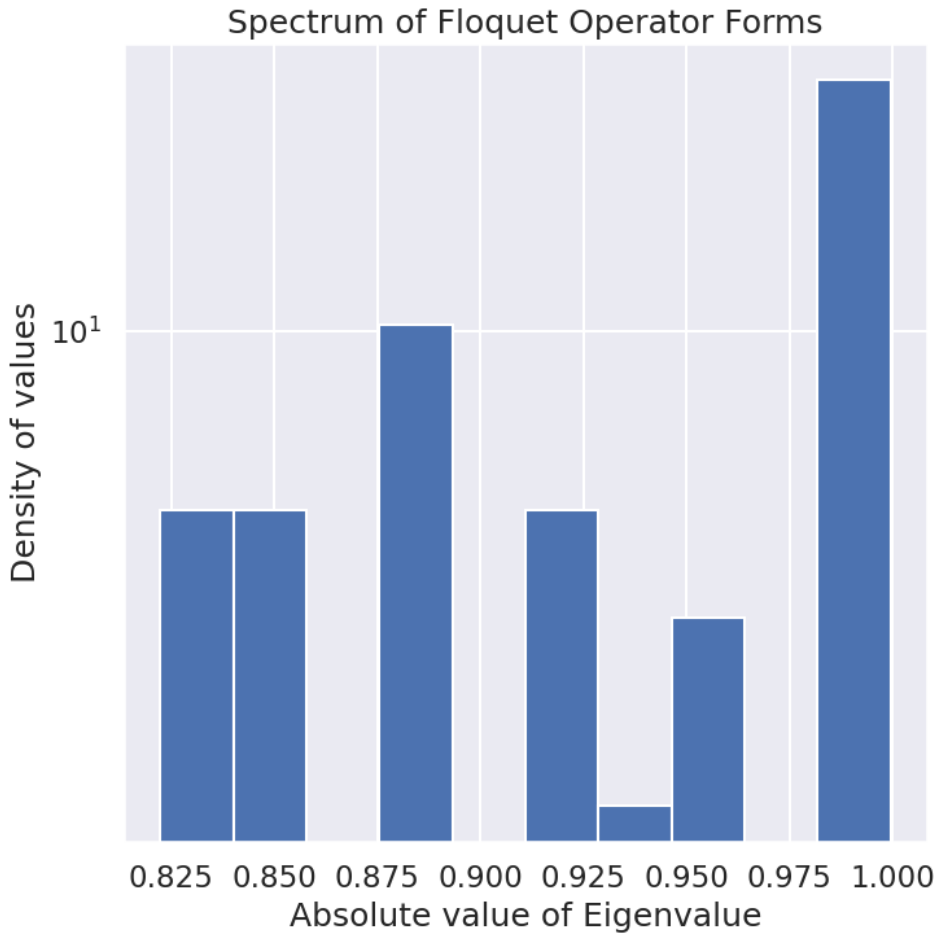
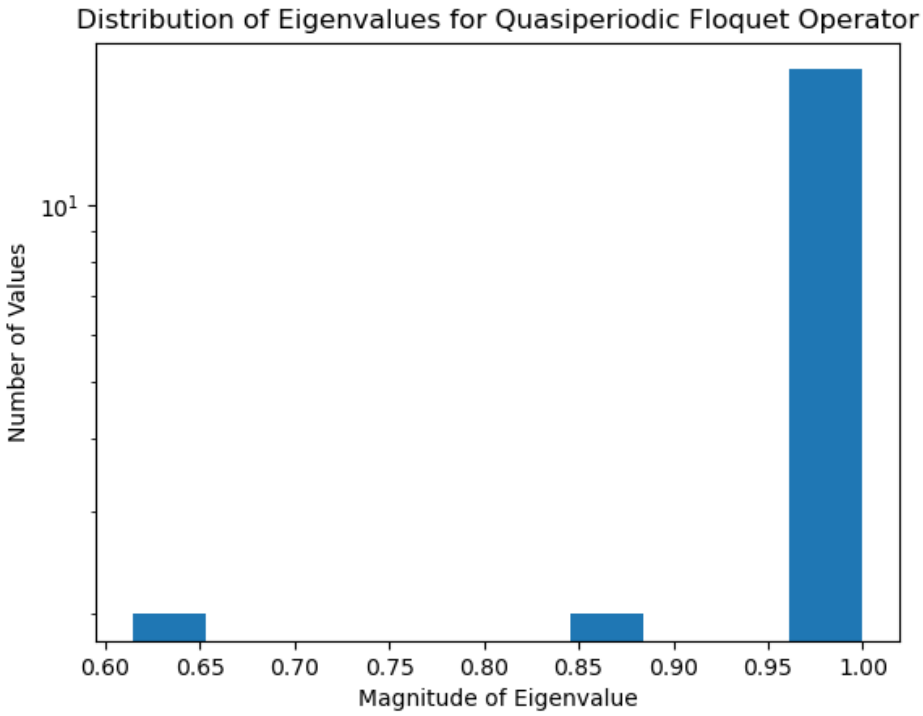


Figure 1: This plot was made from 1331 eigenvalues. Also it should be number of values not density.


Distribution of Eigenvalues for Quasiperiodic Floquet Operator
Number of Values
10^1
0.60
0.65
0.70
0.75
0.80
0.85
0.90
0.95
1.00
Magnitude of Eigenvalue