**Floquet phase function in coding**

Mahbub Rahaman

19 August 2020

I have discussed the coding here in Python , sotry to stick with it, else readers can use any language while keeping the algorithm unaltered.

1. Start with a unitary matrix with the dimension (N x N).
2. Take each coumn of the matrix as wave function of the Schrodinger equation and solve it using "odeintw" in scipy in Python lanugage.
3. Try to keep the wave function in complex number format, cause this will help us to get the result more accurate while in normalisaton and getting expactation value for the obsevables of our interest.
4. After solving "odeintw" we will get another wave function which is basically the wavefunction corresponding to the Hamiltonian of the quantum system.
5. We will get one wave function for each inital wavefunction. Each if these final wave functions is one of N coulmn of the (N x N) matrix which is called "Evolution matrix".
6. Now use "eig" to get the eigen value and eigen function from the evolution matrix. The eiven value of the evolution matrix is $e^{-\Omega t}$. And $\Omega$ is called as "phase function" or "quasi energy".

Our goal is to obtain the variation of the quasi energies over frequency of the external transverse field. Now, using the above step, we can easily obtain N number of quasi energies for each frequency as we are using each column of the unitary matrix of (N x N ) dimension. If we plot the N quasienergy for each frequency then to observe **quantum freezing effect** we can expect to quasienergies to vanishes at some particular frequencies which i anticipate from **Arnab Das’**s paper that those points should be at JH, J is the Bessels's roots of first order.