Department of Physics and Astronomy
University of Heidelberg

Bachelor Thesis in Physics
submitted by

Caspar Groiseau

born in Berlin (Germany)

2015

Steering Random Walks for Detunings from Quantum Resonance

This Bachelor Thesis has been carried out by Caspar Groiseau at the
Institute for Theoretical Physics in Heidelberg
under the supervision of
PD Dr. Sandro Wimberger

Abstract

In quantum resonance, the atom optics kicked rotor, if it underlies a series of kicks, performs a random walk in momentum space. It is possible to steer the outcoming random walks by sampling random kick strengths from specific probability distributions for the kick sequence. The chosen distribution translates into the averaged momentum distribution. Particularly interesting, by taking power law distributions like stable distributions for the strength of the kicks one obtains a Lévy walk in momentum space with the identical power law. This thesis explores the stability of this phenomenon for a small detuning ϵ in the kick period τ from quantum resonance by modelling a theory for the vicinity of the quantum resonance using the ϵ-classical method and supporting it with numerically obtained results.

In der Quantenresonanz, vollführt die atom-optische Umsetzung des Kicked Rotors, wenn er von einer Reihe von Kicks gestoßen wird, einen Random Walk im Impulsraum. Es ist möglich, den resultierenden Random Walk durch das Auswählen von zufälligen Kickstärken aus bestimmten Zufallsverteilungen zu steuern. Die gewählte Verteilung überträgt sich auf die mittlere Impulsverteilung. Durch Wählen einer Power-law-Verteilung erhält man einen Lévy Walk im Impulsraum mit dem gleichem Power-law. Diese Arbeit erforscht die Erhaltung dieses Phänomens für kleine Verstimmungen ϵ in der Kickperiode τ von der Quantenresonanz durch entwickeln einer Theorie für die Umgebung der Quantenresonanz mittels der ϵ-klassischen Methode und numerischer Ergebnisse.

Contents

Introduction

The δ-kicked rotor, a particle that is subject to δ-like kicks, is a paradigm model of quantum chaos. Despite the simplicity of its description the δ-kicked rotor reveals chaotic dynamics and is therefore often a prime example of chaos theory. The importance of the kicked rotor comes as well from the fact that a lot of systems may be reduced to it locally in phase space. The classical version can be described by the so-called standard map or Chirikov-Taylor map, a Poincaré surface of section of the kicked rotor, first introduced by Chirikov [1].

Since our goal is to study quantum dynamics more precisely quantum chaos theory we switch by quantizing to the quantum version, the quantum kicked rotor. On the quantum level there exist two exclusive features to the quantum model which are quantum resonances [2] where energy increases according to ballistic motion and dynamical localization [2]. For dynamical localization energy, in contrast to quantum resonance, stops growing after a specific quantum break time. An experimental realization of the quantum kicked rotor, the atom optics kicked rotor, was realized for the first time by the Raizen group [3]. In this realization cold trapped atoms are exposed to a series of laser pulses corresponding to the kicks. Near the resonances the quantum kicked rotor may be dealt with through the framework of the ϵ-classical method that was designed by Guarneri, Rebuzzini and Fishman [4].

If we measure the resulting momentum distributions of the atoms in the atom optics rotor experiment we observe that in quantum resonance the kicked atoms perform a random walk in momentum space. This thesis focusses on creating a theory for the vicinity of the quantum resonances. A new idea arose, to control or steer the random walk in momentum space by drawing the kick strengths from specific probability distributions. By letting these probability distributions be stable distributions the random walks become Lévy processes.

Random walks in general are very important for statistical physics and so are the special cases of Lévy processes (or walks) in physical fields like fluid dynamics [5] or chaos studies. But these processes are also utilized in non-physical related topics such as finance [6] (Random-Walk-Theory) or quantitative biology [5] (DNA sequence Walk, Heartbeat and Human Walking).

We start by reviewing the background concepts in chapter 1. In chapter 2 we develop a theory for the steering of random walks outside of quantum resonance. Chapter 3 will explain how we performed our numerical simulations and finally, in chapter 4, we display the results and discuss them.

1. Preliminaries

In this chapter we want to introduce the background concepts that we require for our later work. We will start by explaining the experimental realization and theoretical description of the δ-kicked rotor including the pseudo-classical method. Then we will briefly review the idea of quantum resonance and quantum random walk.

1.1. The δ-kicked rotor

1.1.1. Experimental implementation

Most of the atom optics kicked rotor experiments with cold atoms function the same, here we describe how its realization is achieved [7].

The atom optics kicked rotor is experimentally realized by a set of non-interactive atoms, cooled via lasers, in a MOT (magneto-optical-trap). After their release from the trap the atoms underlie a sequence of periodic pulses called kicks from an optical standing wave. The atoms can expand for a short time in the order of several milliseconds before then being targeted by near-resonant light. This results in a fluorescence distribution which is captured by a CCD (charge couple device) and afterwards translated in the corresponding momentum distribution. Contemporary experiment that work with Bose-Einstein condensates are for instance described in [8].

1.1.2. Theoretical description

The quantum kicked rotor

If we consider a one-dimensional model for our kicked atoms we can describe the induced dynamics with this dimensionless ($\hbar = 1$) Hamilton operator [7]:

$$\hat{H}(t) = \frac{\hat{p}^2}{2} + k\cos(\hat{x}) \sum_{j\in\mathbb{Z}} \delta(t - j\tau) \tag{1.1}$$

where $\hat{x}$ and $\hat{p}$ depict the position and momentum operators, k the kicking strength, τ the kick period and j a discrete time variable indicating the number of kicks. The evolution of the state of the kicked atom, after a kick until the next one, is established by the unitary one-cycle Floquet operator [7]:

$$\hat{U}_k = \exp\left(-i\int\limits_t^{t+\tau}\hat{H}(t')dt'\right) = e^{-ik\cos(\hat{x})}e^{-\frac{i}{2}\tau\hat{p}^2} \tag{1.2}$$

The first part of the Floquet operator characterizes the effect of the kick and the second the free evolution between this kick and the next one. This models Hamiltonian (1.1) describes atoms that move on a line, while we normally describe atoms moving a circle. We can map one to the other because our Hamilton operator is periodic in $\hat{x}$ so as to the Bloch theorem holds. We introduce the spatial periodicity (boundary conditions) $\hat{\theta} = \hat{x} \mod (2\pi)$. $\hat{\theta}$ is our new (angular) position operator. Similarly we separate the momentum $p = n + \beta$ into an integer part n and a conserved non-integer part $\beta \in [0;1)$, that is called quasimomentum. β is conserved because transitions between states that are not different by an integer in momentum are forbidden. The n are the eigenvalues of the angular momentum operator $\hat{n} = -i\frac{d}{d\theta}$. The Floquet operator then takes the following form:

$$\hat{U}_{\beta,k} = e^{-ik\cos\left(\hat{\theta}\right)}e^{-\frac{i}{2}\tau(\hat{n}+\beta)^2} \tag{1.3}$$

We can write the wave packet of the atom $\psi(x)$ as a superposition of periodic plane (Bloch) waves $e^{i\beta x}\psi_\beta(x)$ where $\psi_\beta(x)$ is 2π-periodic:

$$\psi(x) = \langle x|\psi\rangle = \int\limits_0^1 d\beta\rho_\beta e^{i\beta x}\psi_\beta(x) \tag{1.4}$$

If the initial state of the atom is given by a plane wave with a discrete momentum $p_0 = n_0 + \beta_0$, the Bloch wave describing the system is:

$$\psi_\beta(\theta) = \langle\theta|\psi_\beta\rangle = \frac{1}{\sqrt{2\pi}}e^{in_0\theta} \tag{1.5}$$

$$\rho_\beta = \delta(\beta - \beta_0) \tag{1.6}$$

The pseudo-classical kicked rotor

The theoretical concept of the pseudo-classical rotor (see [9]) is very helpful for the description of the quantum kicked rotor experiment for cold non-interacting atoms in the proximity of quantum resonance. Near quantum resonance for $\tau = 2\pi l + \epsilon$ with $l \in \mathbb{N}$ and ϵ close to 0 we may rescale $\tilde{k} = |\epsilon|k$ and $\hat{I} = |\epsilon|\hat{n}$. The factor ϵ describes here the degree of "quantumness" (similar to $\hbar$) with $|\epsilon| \to 0$ being the classical limit. For the free evolution part we utilize $e^{-i\pi l\hat{n}^2} = e^{-i\pi l\hat{n}}$. This changes our Hamilton operator and our time evolution operator to:

$$\begin{aligned} \hat{U}_{\beta,\tilde{k}} &= e^{-\frac{i}{|\epsilon|}\tilde{k}\cos\left(\hat{\theta}\right)} e^{-\frac{i\tau\beta^2}{2}} e^{-\frac{i}{|\epsilon|}\left[\frac{\hat{I}^2}{2\,\mathrm{sgn}(\epsilon)}+\hat{I}(\pi l+\tau\beta)\right]} && (1.7) \\ \hat{H} &= \frac{\hat{I}^2}{2\,\mathrm{sgn}(\epsilon)} + \hat{I}(\pi l + \tau\beta) && (1.8) \end{aligned}$$

The ϵ-classical kicked rotor can be described by the following discrete map which relates the variables I and θ directly after the j-th kick to those immediately after the $(j+1)$-st kick.

$$\begin{aligned} \theta_{j+1} &= \theta_j + \mathrm{sgn}(\epsilon) I_j + \pi l + \tau\beta \qquad \mod (2\pi) && (1.9) \\ I_{j+1} &= I_j + \tilde{k}\sin(\theta_{j+1}) && (1.10) \end{aligned}$$

We change the variable $J = \mathrm{sgn}(\epsilon)I + (\pi l + \tau\beta)$ to get rid of the β-dependency.

$$\begin{aligned} \theta_{j+1} &= \theta_j + \mathrm{sgn}(\epsilon) J_j \qquad \mod (2\pi) && (1.11) \\ J_{j+1} &= J_j + \tilde{k}\sin(\theta_{j+1}) && (1.12) \end{aligned}$$

This last map known as the ϵ-classical map is in the end just a variant of the standard map. The momentum is incorporated in J with $p = \frac{J}{|\epsilon|}$. Due to the fact that the detuning ϵ is inherently small the phase space is for no too large k and $\tilde{k}$ nearly integrable, thus not chaotic and a good approximation to quantum motion [9].

1.2. Quantum resonances

Quantum resonance [2] is an effect specific to the quantum regime and cannot be observed in the classical regime. We call quantum resonance the maximal absorption of energy by the δ-kicked atoms from the kicking field. The momentum then grows linearly and the corresponding energy quadratically in time.

For the main resonances the first part of the Floquet operator, the one associated to the free evolution, vanishes ($e^{-\frac{i}{2}\tau\hat{n}^2} = 1$), so that we get a phase revival of the wave function in momentum space. The consequence of this is that the atoms directly after a kick and just before the next are in the same state. Then one cannot differentiate between T kicks of strength k and a kick of strength Tk. The main quantum resonances occur at specific τ-β couples:

$$\begin{aligned} &\tau_{resonance} = 2\pi l, && l \in \mathbb{N} && (1.13) \\ &\beta_{resonance} = \frac{1}{2} + \frac{i}{l} \mod (1), && i = 0, 1, ..., l-1 && (1.14) \end{aligned}$$

For our computations we will always work with the specific quantum resonance at $\tau = 4\pi$ and $\beta = 0$.

A speciality of quantum resonance is that we can express the momentum distribution P analytically [9]:

$$P(n,T|n_0,k) = J^2_{n-n_0}(kT) \tag{1.15}$$

Where $J_m(x) = \sum\limits_{i=0}^{\infty} \frac{(-1)^i(\frac{x}{2})^{2i+m}}{\Gamma(m+i+1)i!}$ is a Bessel function of first kind and order m.

1.3. Quantum random walks

A classical random walk is a stochastic process modelling a random motion. The simplest version, the one-dimensional "Drunkard Walk", with given probabilities to go forward or backwards leads after a certain amount of steps to a binomial distribution of the final positions which after the central limit theorem trends to a normal distribution.

The quantum random walk is the quantum mechanical analogue of the classical random walk and was introduced by Aharonov et al. [10]. For a comprehensive introduction see [11]. The main interest in quantum random walk lies in the probable enhancement of computation power by leading to more efficient algorithms than in classical computing. In the quantum case the walker is for example a particle with two degrees of freedom, the location and the spin.

If we now consider a particle with spin $\frac{1}{2}$ and a translation by l whether the displacement takes place in the positive or negative direction will be dependent on the spin of the particle. If the initial state is a superposition of the spin eigenstates the result of the time evolution will be superposition of the positions. This is called if iterated (measurement in the original spin base and reinitializing the spin to the initial spin state) a biased random walk. The spin corresponds to a biased coin which is flipped upon applying the time evolution and determines the direction of the step.

Alternatively to measuring in the original spin base one might measure in a rotated basis or rotate the spins before measuring. If the uncertainty in position (width of the wave packet) $\Delta x \ll l$ one of the outcoming displacements $l\delta_{\uparrow/\downarrow}$ can be much larger than l for special relations between rotation angle and probability amplitudes. This is unique to quantum random walks.

There are 2 models to describe quantum random walks, the discrete-time and the continuous-time quantum walk. The difference between the two is that the associated evolution operator for the first can only be applied at specific discrete times while for the second it may be applied at any time.

Quantum random walks are important in the context of the atom optics kicked rotor because the initial momentum state p of a kicked atom will be in superposition of momentum eigenstates after applying a sequence of kicks. From a quantum point of view the corresponding Floquet operators may lead to the emergence of interferences along the sequence so that the distribution of momenta will differ from the classical view like for example at quantum resonance. The atom optics kicked rotor fulfills a random walk in momentum space so to speak. To qualify for being a quantum random walk our kicked rotor setup lacks a second internal degree of freedom representing the coin that we could entangle with the momentum degree of freedom.

2. Theory

In this Chapter we want to establish a theory for steering random walks for finite detunings ϵ by using the concept of the pseudo-classical method. Like for quantum resonance by randomizing the kick strength at each kick we implement a classical randomness.

2.1. Randomizing the kick strength

Instead of considering kick sequences with constant kick strengths we shall vary the kick strength at each kick. Thus changing the Hamiltonian (1.1) to:

$$\hat{H}'(t) = \frac{\hat{p}^2}{2} + \cos(\hat{x}) \sum_{j \in \mathbb{Z}} k_j \delta(t - j\tau) \tag{2.1}$$

giving us the corresponding Floquet operator:

$$\hat{U}'_{\beta,k_j} = e^{-ik_j \cos\left(\hat{\theta}\right)} e^{-\frac{i}{2}\tau(\hat{n}+\beta)^2} \tag{2.2}$$

As we shall see, this allows us to steer the random walk. For every iteration of the kick sequence another operator is applied giving different results in momentum space depending on the chosen kick strength. By choosing k from different probability distributions we influence this dependency and may, for example enforce a Lévy walk by sampling from stable distribution where extremely large k are not that suppressed.

The previously explained method corresponds to integrate a classical random walk into a quantum random walk. While the momentum distribution still is a quantum mechanical result it appears to be classical because the quantity k controls the spread in momentum. Later we will form the classical mean over the momentum distribution, then the result is purely classical because of decoherence.

2.2. Deriving the momentum distribution

Since there is no known analytic solution, not even for fixed k, for the momentum distribution of the quantum kicked rotor outside of quantum resonance we have to compute the quantum dynamics numerically. Nonetheless we can for a small detuning ϵ in the kicking period τ use the ϵ-classics framework.

We use the previously mentioned ϵ-classical map:

$$\begin{aligned} \theta_{j+1} &= \theta_j + \mathrm{sgn}(\epsilon) J_j \qquad \mathrm{mod}\ (2\pi) && (2.3) \\ J_{j+1} &= J_j + \tilde{k}_j \sin(\theta_{j+1}) && (2.4) \end{aligned}$$

and take a look at how the momentum J evolves during a kick.

$$\begin{aligned} \delta J_{j+1} &= J_{j+1} - J_j && (2.5) \\ &= \tilde{k}_j \sin(\theta_{j+1}) && (2.6) \\ &= \tilde{k}_j \sin(\theta_j + \mathrm{sgn}(\epsilon) J_j) && (2.7) \\ &= \tilde{k}_j \sin(\theta_j + \phi_j) && (2.8) \\ &= \tilde{k}_j \sin\left(\tilde{\phi}_j\right) && (2.9) \end{aligned}$$

Here we made the approximation that the changes in momentum from a kick δJ_{j+1} are uncorrelated. The rough assumption here is that θ_j has no history. This will be represented by a random phase ϕ_j being added to the prior angle. Naturally $\tilde{\phi}_j = \theta_j + \phi_j$ is then just as well a random phase. The formula for the momentum J_T after T kicks is then:

$$\begin{aligned} J_T &= J_0 + \sum_{j=1}^{T} \tilde{k}_j \sin\left(\tilde{\phi}_j\right) && (2.10) \\ &= J_0 + \sum_{j=1}^{T} \tilde{k}_j \, \mathrm{Im}(e^{-i\tilde{\phi}_j}) && (2.11) \end{aligned}$$

In former works [12] spatial phase shift were considered as well, but we ignored them as they may be integrated in the kick strengths and fixing it at 0 resulted in the best results. There the averaged momentum distribution in resonance could be showed to directly depend on the distance from the origin from a classical random walk R_T in the complex plane. This can be done here as well:

$$\begin{aligned} R_T(\{\tilde{k}_j\}, \{\tilde{\phi}_j\}) &= \sum_{j=1}^{T} \tilde{k}_j e^{-i\tilde{\phi}_j} && (2.12) \\ &= \sum_{j=1}^{T} \tilde{k}_j (\cos\left(\tilde{\phi}_j\right) - i \sin\left(\tilde{\phi}_j\right)) && (2.13) \end{aligned}$$

Then the distance to the origin is:

$$|R_T|^2 = (\sum_{j=1}^{T} \tilde{k}_j \cos\left(\tilde{\phi}_j\right))^2 + (\sum_{j=1}^{T} \tilde{k}_j (\sin\left(\tilde{\phi}_j\right))^2 \tag{2.14}$$

$$= \sum_{i=1}^{T}\sum_{j=1}^{T} \tilde{k}_i\tilde{k}_j \cos\left(\tilde{\phi}_i\right)\cos\left(\tilde{\phi}_j\right) + \sum_{i=1}^{T}\sum_{j=1}^{T} \tilde{k}_i\tilde{k}_j \sin\left(\tilde{\phi}_i\right)\sin\left(\tilde{\phi}_j\right) \tag{2.15}$$

$$= \sum_{i,j}^{T} \tilde{k}_i\tilde{k}_j \cos\left(\tilde{\phi}_i - \tilde{\phi}_j\right) \tag{2.16}$$

$$|R_T| = \sqrt{\sum_{i,j}^{T} \tilde{k}_i\tilde{k}_j \cos\left(\tilde{\phi}_i - \tilde{\phi}_j\right)} \tag{2.17}$$

If we now go back to our problem in (2.11), (2.17) will show to be very helpful but first we have to use these standard trigonometrical formulas:

A sum of sines with different amplitudes and phases can be rewritten as a single sine with a single amplitude and phase (phasor addition):

$$\sum_i \tilde{k}_i \sin\left(\tilde{\phi}_i\right) = \tilde{k}\sin\left(\tilde{\phi}\right) \tag{2.18}$$

with

$$\tilde{k}^2 = \sum_{i,j} \tilde{k}_i\tilde{k}_j \cos\left(\tilde{\phi}_i - \tilde{\phi}_j\right) \tag{2.19}$$

and

$$\tan\left(\tilde{\phi}\right) = \frac{\sum\limits_i \tilde{k}_i \sin\left(\tilde{\phi}_i\right)}{\sum\limits_i \tilde{k}_i \cos\left(\tilde{\phi}_i\right)} \tag{2.20}$$

This applied on (2.11) gives us:

$$\sum_{j=1}^{T} \tilde{k}_j \sin\left(\tilde{\phi}_j\right) = \sqrt{\sum_{i,j}^{T} \tilde{k}_i\tilde{k}_j \cos\left(\tilde{\phi}_i - \tilde{\phi}_j\right)} \sin\left(\arctan\left(\frac{\sum\limits_j^T \tilde{k}_j \sin\left(\tilde{\phi}_j\right)}{\sum\limits_j^T \tilde{k}_j \cos\left(\tilde{\phi}_j\right)}\right)\right) \tag{2.21}$$

If we now fix the phase ϕ the change in momentum will be dominated by the sum of the effective kick strengths $\tilde{k}$. The variance of the sines is then assumed to be $\sigma^2 = \sigma^2_{\sin(\theta_j+\phi)} \approx \frac{1}{2\pi}\int\limits_0^{2\pi}(\sin^2(x) - \mu^2)dx = \frac{1}{2}$

with $\mu = \frac{1}{2\pi}\int\limits_0^{2\pi} \sin(x)dx = 0$.

$$\begin{aligned}
\sum_{j=1}^{T} \tilde{k}_j \sin(\theta_j + \phi) &\approx \sigma \sum_{j=1}^{T} \tilde{k}_j &(2.22)\\
&\approx \frac{1}{\sqrt{2}} \sum_{j=1}^{T} \tilde{k}_j &(2.23)
\end{aligned}$$

The same argumentation is valid for the cosines.

$$\sum_{j=1}^{T} \tilde{k}_j \cos(\theta_j + \phi) \approx \frac{1}{\sqrt{2}} \sum_{j=1}^{T} \tilde{k}_j \qquad (2.24)$$

This is inserted into (2.22) to give:

$$\begin{aligned}
\sum_{j=1}^{T} \tilde{k}_j \sin\left(\tilde{\phi}_j\right) &\approx |R_T| \sin\left(\arctan\left(\frac{\frac{1}{\sqrt{2}}\sum\limits_{j=1}^{T}\tilde{k}_j}{\frac{1}{\sqrt{2}}\sum\limits_{j=1}^{T}\tilde{k}_j}\right)\right) &(2.25)\\
&= |R_T| \sin(\arctan(1)) &(2.26)\\
&= \frac{1}{\sqrt{2}}|R_T| &(2.27)\\
&\propto |R_T| &(2.28)
\end{aligned}$$

This means that we expect the random walk in momentum space, for a detuning ϵ not to far from quantum resonances, to be steerable by the distribution of kick strengths ρ_k. We could reproduce these expectations in our numerical simulations that we will introduce next and shall discuss them further in the results.

$$P(n, T|n_0, \{\tilde{k}_j\}, \{\tilde{\phi}_j\}) \approx P(n, T|n_0, |R_T|) \qquad (2.29)$$

3. Numerical simulations

This chapter gives an overview on how our numerical data was generated and discusses the problems that came with the choice of the length of the momentum basis as a consequence of heavy tailed distribution of kick strengths.

3.1. Computation procedure

3.1.1. Quantum map

The averaged momentum distribution is computed via numerical simulations more precisely a Monte Carlo simulation that we will shortly describe in the following section [13]. The initial state in (angular) momentum space is given by:

$$\psi(n, j = 0) = \delta(n - n_0) \tag{3.1}$$

The kick-to-kick time evolution operator can be split into two parts:

$$\hat{U} = \hat{K}\hat{F} \tag{3.2}$$

Here $\hat{K} = e^{-ik\cos(\hat{\theta})}$ and $\hat{F} = e^{-\frac{i}{2}\tau\hat{n}^2}$. We work with a finite basis $N = 2^x$ (x is an integer so that we may use the Fast Fourier Transform) such that $\theta_i = \frac{2\pi}{N}i$ and $n = -\frac{N}{2}, -\frac{N}{2} + 1, ..., \frac{N}{2} - 1$. We start by computing $\hat{F}\psi(0)$ in (angular) momentum representation. Then we perform a Fast Fourier Transform into (angular) position space and there compute $\hat{K}\hat{F}\psi(0)$. Finally we Fast Fourier Transform back to (angular) momentum space and obtain $\hat{U}\psi(j = 0) = \psi(j = 1)$. This is iteratively done T times, the amount of kicks performed on the system. Whereby each kick is randomly drawn from a probability distribution. In the appendices A and C we explain how the Fast Fourier Transform was implemented and the pseudo random numbers were generated. We get the momentum distribution P_r of one realisation by computing:

$$P_r(n, T) = |\psi(n, j = T)|^2 \tag{3.3}$$

Since we are interested in the averaged momentum distribution $\bar{P}$, one has to average in a final step the set of momentum distributions where R is the number of realisations.

$$\bar{P} = \frac{1}{R}\sum_{r=1}^{R} P_r \tag{3.4}$$

The C++-code with which the computation were finally performed can be found in the appendix D.

3.1.2. Pseudo-classical map

For the pseudo-classical method the computation procedure consists basically in just iterating the previously mentioned pseudo-classical map. For each realization $r \in (1, R)$ we do this for a certain number Θ of realisations of θ where θ is each time drawn from a uniform distribution between 0 and 2π. After that we sort the resulting momentum in bins which result in the momentum distribution. The Matlab-code with which the computation were finally performed can be found in the appendix D.

3.2. Choice of the base length

As we stated in (3.1) we need to choose a finite basis for our numerical simulations. The base length is in addition restricted to be a power of two because of our Fast Fourier Transform routine (see appendix A). The choice is very important, since a basis that is too small will impair the result and a basis that is too large will take a lot of computation time. In the following we want to motivate the choice of our basis.

For this we will build a test series where we vary the length of the momentum base while picking the kick strengths k from a special case of stable distributions the Cauchy distribution with location parameter $\mu = 0$ and scale parameter $\gamma = 1$:

$$\rho_k = \frac{1}{\pi\gamma(1 + (\frac{k-\mu}{\gamma})^2)} \tag{3.5}$$

This distribution then decreases with x^{-2} in its tails and since the sum of M identical Cauchy distributions is again Cauchy distributed with a new scale $\gamma' = M\gamma$ and a new location parameter $\mu' = \mu M$ we expect for our resulting averaged momentum distribution again to observe a x^{-2} decay in the tails.

The base not being infinite results in events of higher momenta than the basis length being relocated to smaller momenta. Of course this causes the probability of the momenta to grow, ergo to raise the probability distribution. This rising is especially noticeable (in a logarithmic plot) in the tails were the probabilities are low and it seems to occur there first. We cannot prevent this from happening but by choosing bases that are large enough the effect will be shifted to larger momenta and therefore lower probabilities. Once it will be only appreciably affect probabilities $\lesssim \frac{1}{R}$, it is not relevant any more because it will be covered by statistical noise. For very small N there is an extreme case where the probability distribution starts to bend upward.

Bases where this behaviour could be observed with the naked eye we could exclude immediately for the use of our numerical simulations. Figure (3.1) shows the strong deviation of the averaged momentum distribution of small bases to those of large bases with the characteristic rearing up.

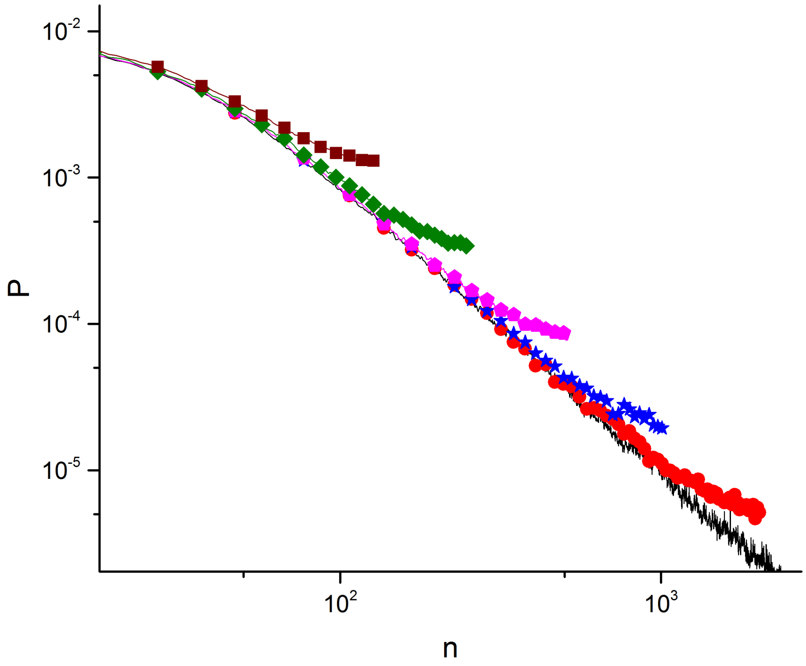
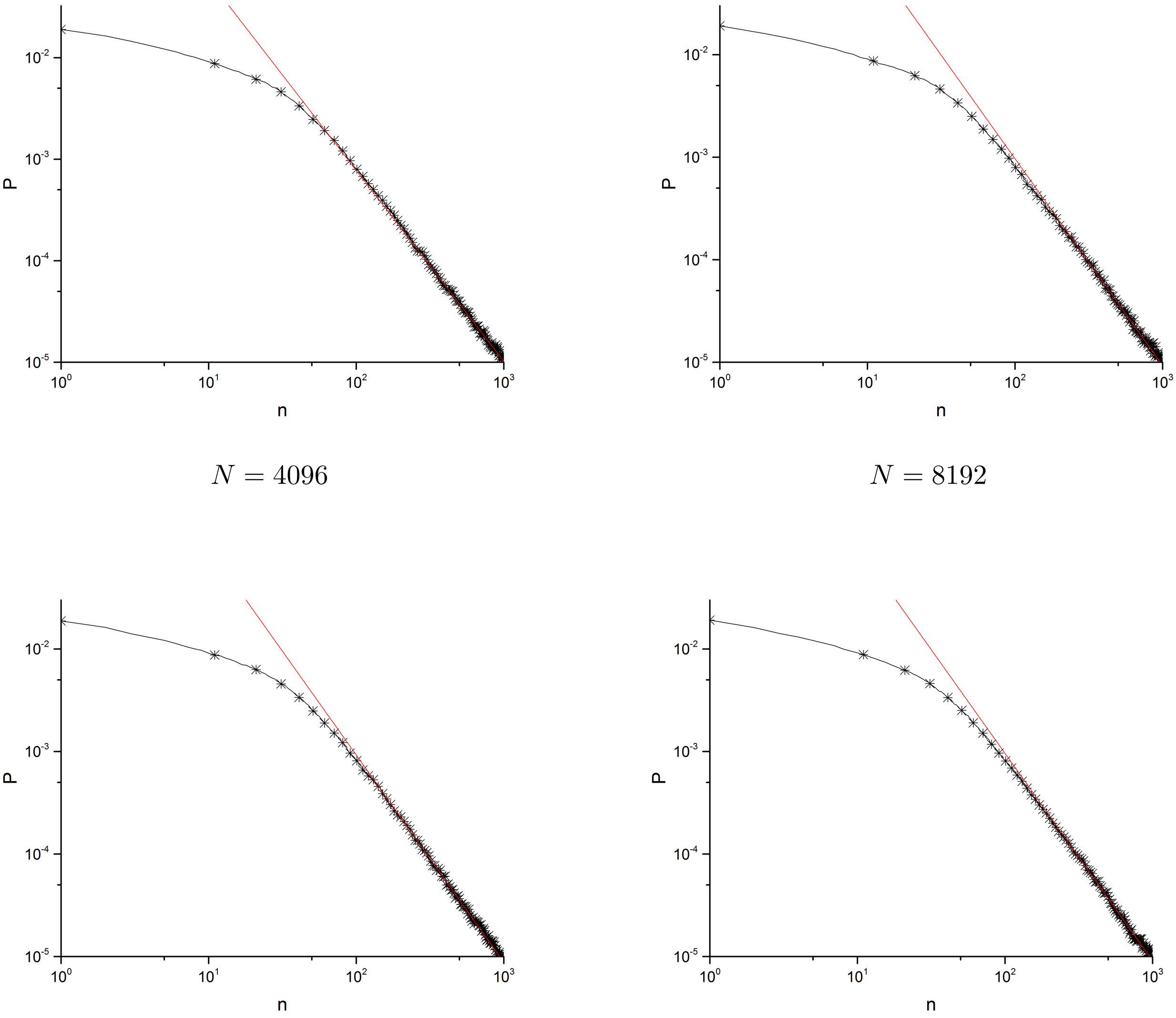


Figure 3.1.: Bending of the averaged momentum distribution for small N.

Double-logarithmic plot of the averaged momentum distributions obtained numerically with 20000 realisations and $\epsilon = 0.005$ for different small base lengths. The simulations with $N = 128$ (brown squares), $N = 256$ (green lozenges), $N = 512$ (pink pentagons), $N = 1024$ (blues stars), $N = 2048$ (red circles) show the characteristic bending. The black line shows a $N = 32768$ case to see the deviation from the expected tail.

The problem is now that for larger momentum base lengths N (from $N \approx 2048$ on) this effect is not that perceivable any more while still potentially largely impacting the power law property in the tails of the distribution. Which is why we employ the power law exponent as a criterion to determine whether the basis is flawless enough.



$N = 4096$

$N = 8192$

$N = 16384$

$N = 32768$

Figure 3.2.: Determining an appropriate base length.

> The double-logarithmic plots show the numerically ($R = 20000$, $T = 50$ and $\epsilon = 0.005$) obtained averaged momentum distributions (black crosses) and a power law function fitted to their tails (red line) for different greater base lengths: $N = 4096$, $N = 8192$, $N = 16384$ and $N = 32768$. The corresponding power law exponents are 0.88, 1.02, 1.02 and 1.02.

Figure (3.2) shows the resulting averaged momentum distributions for array lengths varying from 4096 to 32768 for a simulations with $T = 50$, $n_0 = 0$, $\epsilon = 0.005$. The curves with the array lengths 8192, 16384 and 32768 properly depict the expected power law of $\alpha = 1$ induced by the Cauchy distribution. The simulations for $N = 4096$ though falsely exhibits a power law behaviour of approximately $\alpha = 0.88$ and is interpreted as being to small. For our purposes base lengths exceeding $N = 8192$ showed to be ideal and were the only ones used for the remainder of this thesis.

4. Results

In this final chapter we will analyze the results of our numerics and discuss them in respect of our theory. We start by checking that our simulations reproduce the quantum resonance results in the $\epsilon \to 0$ limit. We proceed with sampling the kick strengths from special distributions, determining their averaged momentum distribution and testing them numerically. Here we produce Lévy walks in momentum space.

4.1. The $\epsilon \to 0$ limit

Although our interest lies in the ambiance of the quantum resonance we have to check whether our simulations match the results obtained for quantum resonance so that we can correlate our findings. After all we want to test the robustness of the steering in quantum resonance with respect to experimental inexactness. We approached the quantum resonance by changing the order of magnitude of ϵ and find that for $\epsilon < 0.001$ the resonance curve was not distinguishable from the detuned ones.

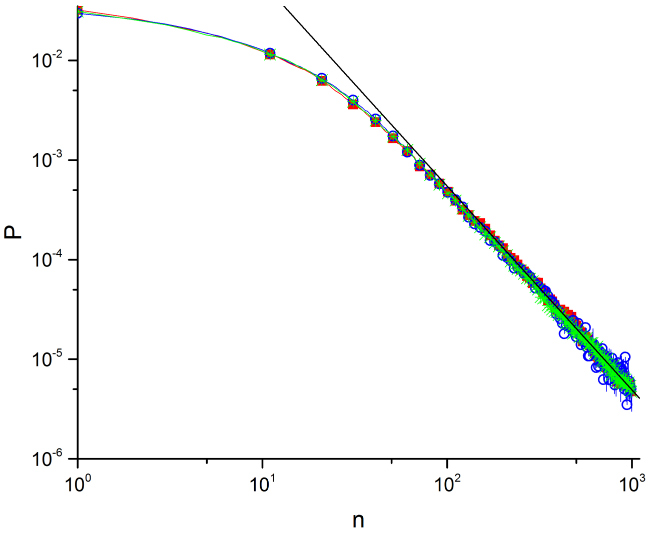


Figure 4.1.: The $\epsilon \to 0$ limit.

Numerically obtained averaged momentum distributions with 20000 realisations, 50 kicks. The first curve (red squares) shows the resonance (retrieved from [12]), the second (green crosses) shows the quantum mechanical result and the third the pseudo classical method (blue circles) in the $\epsilon \to 0$ limit with $\epsilon = 0.001$. Additionally the figure shows a power law fit to the tail with a power law exponent of $\alpha = 1.05$.

As we could see in (4.1) both numerics for the quantum version and the ϵ-classics reproduce for the $\epsilon \to 0$ limit the results of the quantum resonance. Now that we have verified that our work is properly linked to the resonant case we may tackle the main interest of the thesis: steering random walks with power law behaviour for detunings.

4.2. Power laws

In this section we will present how to obtain Lévy walks in momentum space so as to the averaged momentum distribution obeys to a power law in its tails similarly to what was done in quantum resonance [12] except here for finite detunings in the kick period. For this we sample the kick strengths k_j for each kick from a stable distribution with heavy tails $x^{-(1+\alpha)}$ for $0 < \alpha < 2$. This makes it so that extreme events, in our case extreme values for the kick strength, are more likely to happen in comparison to more common distributions like normal distributions. In part B and C and of the appendix we introduce the concept of stable distributions and explain how to generate pseudo random number distributed accordingly (for the $S(\alpha, \beta, 1, 0)$ case only).

Since we derived in (2.28) that the momentum distribution depends on the distance form the origin $|R_T|$. By choosing k form a heavy-tailed distribution R_T will be a Lévy walk in the complex plane. Figure (4.2) shows such a Lévy walk with the characteristic large jumps in position. The variance of the R_T will be infinite so that we may not use the central limit theorem, however we can use the generalized central limit theorem [14] which says that we have an attraction to stable distribution so that $|R_T|$ is stable distributed.

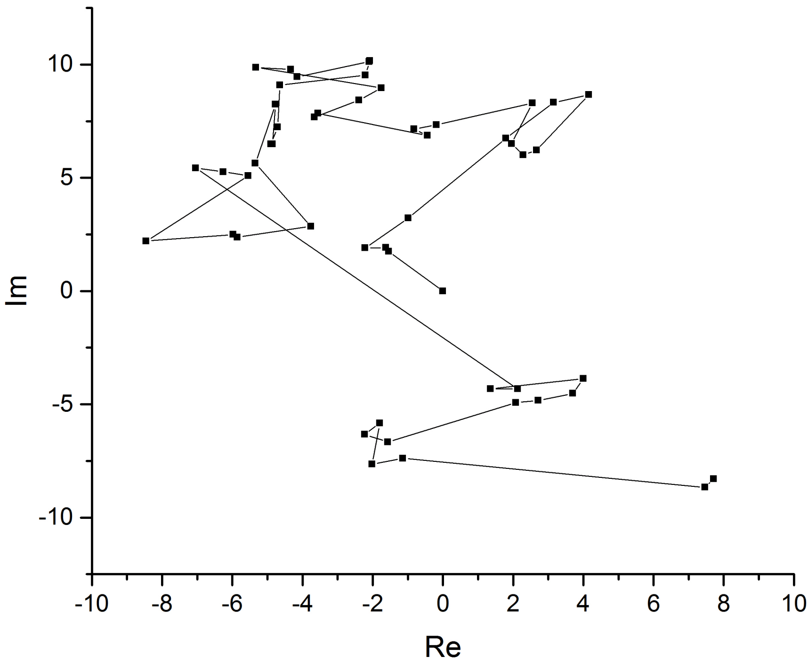
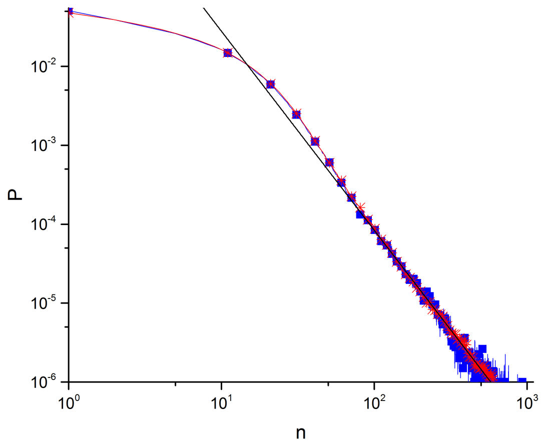


Figure 4.2.: Lévy flights.

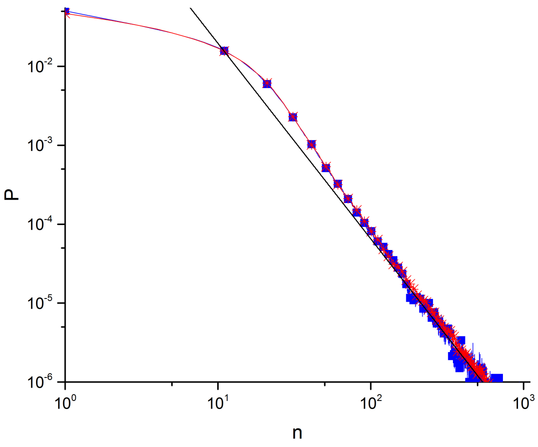
This plot shows a Lévy walk of R_T with Cauchy distributed ($\gamma = 1$ and $\mu = 0$) k and uniformly distributed $\tilde{\phi} \in [0, 2\pi)$ for $T = 50$. The walk starts at the location $[0, 0]$.

The sum of M independent identically symmetrically stable distributed random variables $X_i \propto S(\alpha, \gamma, \delta)$ will also be stable distributed, namely with $\sum_i X_i \propto S(\alpha, M^{\frac{1}{\alpha}}\gamma, M\delta)$ [18]. This means hat if we choose the k_j from a stable distribution $|R_T|$ will be distributed like $S(|R_T||\alpha, 0, T^{\frac{1}{\alpha}}, 0)$.

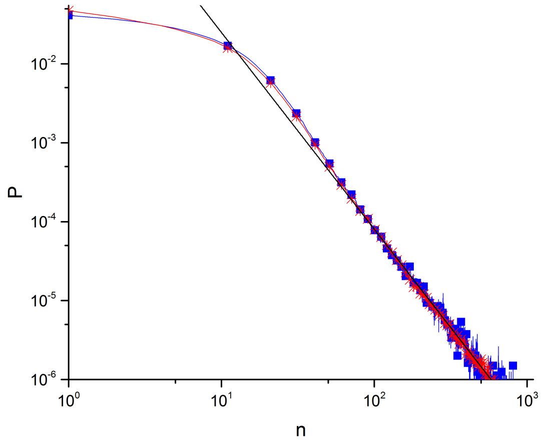
Representatively the figures (4.3) exhibit the averaged momentum distributions in double-logarithmic paper extracted from the quantum map and the pseudo-classical method. Both simulations were done for a sequence of $T = 50$ kicks, with the starting momentum $n_0 = 0$, varying detuning ϵ and k chosen from the stable distribution $S(\alpha = 1.5, \beta = 0, \gamma = 1, \delta = 0)$. For the quantum version we used $R = 100000$ and for the pseudo-classical variant $R = 40000$ and $\Theta = 100$. The plots show compliance of the pseudo-classical method and the true quantum computation also for greater detunings. And more important we clearly can observe the expected power law with $\alpha = 1.5$.



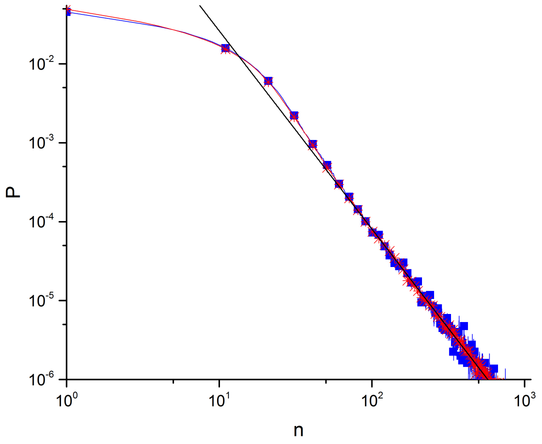
$\epsilon = 0.001$



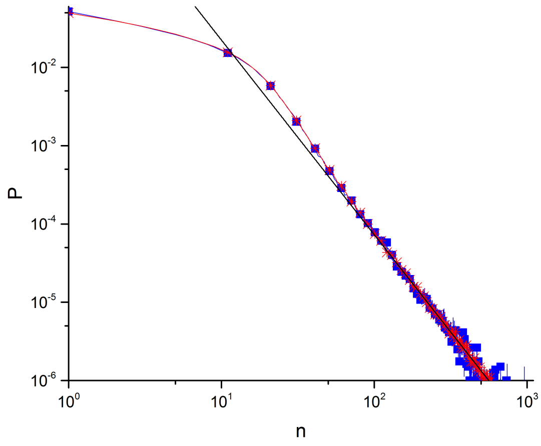
$\epsilon = 0.0025$



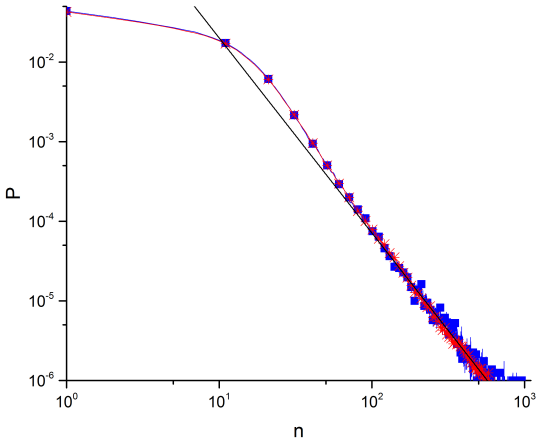
$\epsilon = 0.005$



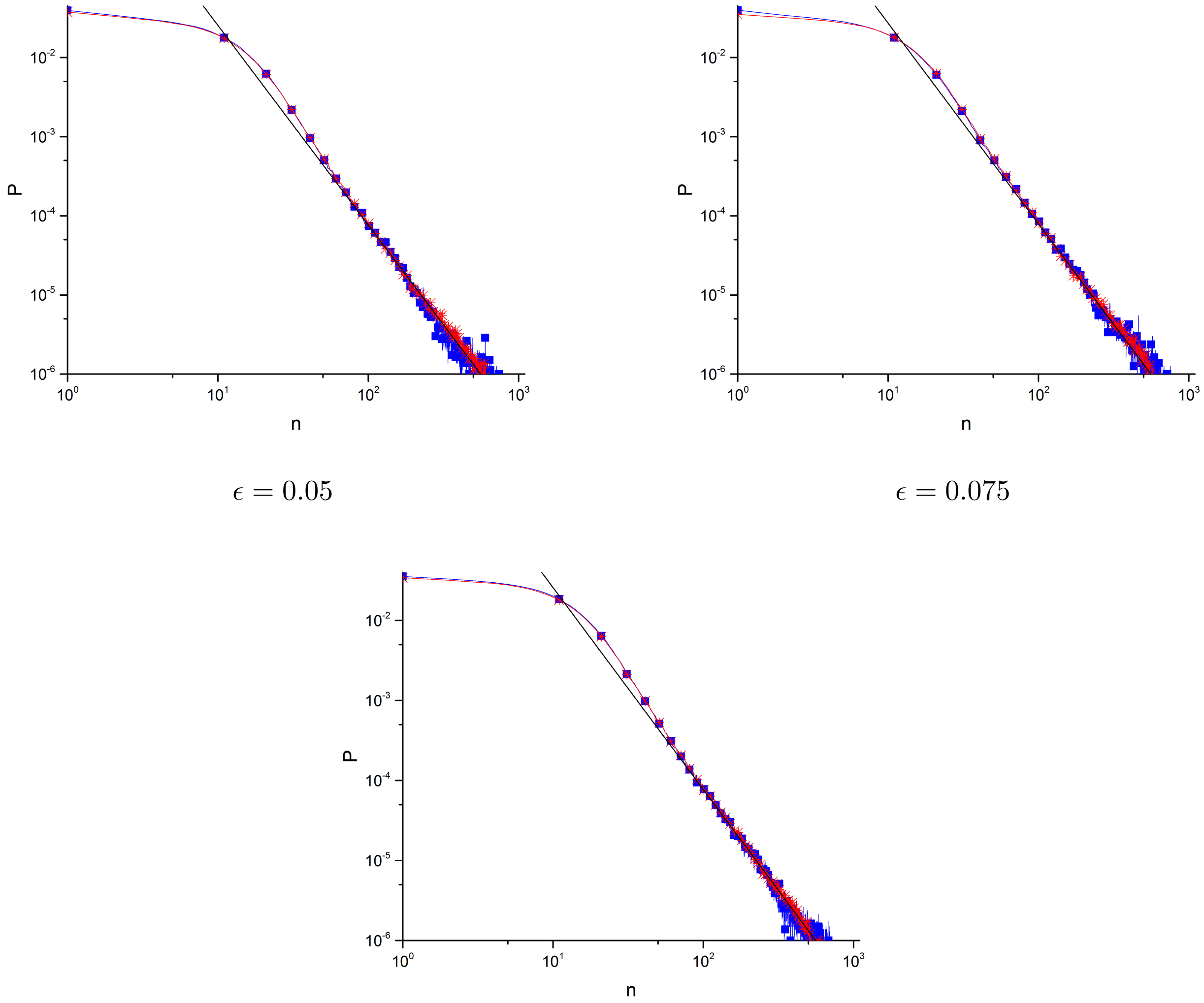
$\epsilon = 0.0075$



$\epsilon = 0.01$



$\epsilon = 0.025$



$\epsilon = 0.05$

$\epsilon = 0.075$

$\epsilon = 0.1$

Figure 4.3.: Power laws.

> Double logarithmic plots of the simulated averaged momentum distributions containing the quantum version (red stars) and the pseudo classical (blue squares) version and a power law fit (black line) with $\alpha = 1.5$ for a selection of detunings $\epsilon \in [0.001, 0.1]$.

To recapitulate, selecting the k_j from stable distributions resulted in a Lévy flight in momentum space with the same power law exponent α as the chosen stable distribution, therefore confirming our theory. But it was also observed that the distribution for the different detunings do not converge towards the same limiting distribution. This will covered in more detail in the next section.

4.3. Dissociation in function of ϵ

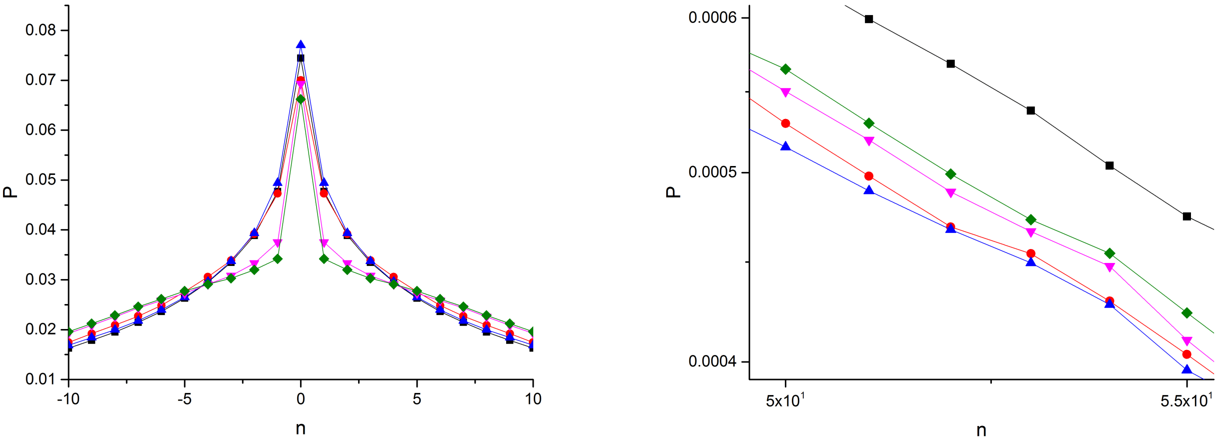
Although the different curves (resonant and detuned ones) seem to follow the same power law they differ in position. For simulations with a very great number of realisations R so that the averaged momentum distribution has converged not just for the small n but also for the n in the power law section of the distribution, we can see an interesting phenomena related to the non-resonant case.

The different distributions for the different detunings ϵ do not seem to tend towards the same limit distributions and overlap but rather seem to possess an order depending on the detuning in the kicking period τ from the resonant case. To explain this occurrence we take a closer look at the free evolution part of our Floquet operator since he is the component ϵ comes into play:

$$e^{-\frac{i}{2}\tau\hat{p}^2} = e^{-\frac{i}{2}\epsilon\hat{p}^2} \neq 1 \tag{4.1}$$

The free evolution term outside of quantum resonance not being equal to unity, makes it so that the phase revival we would have had in resonance is replaced with partial destructive interference whose strength depends on the detuning ϵ and the number of kicks T.

A great part of the momenta will be located near to the middle peak at $n = 0$. In the centre of the distribution we can observe a dependence of the main peak (height and width) on the detuning ϵ. We expect that curves with more weight in the centre (higher and broader peak) will show negative offsets to the resonance in their tails for reasons of normalization. The following graph (4.4) shows close-ups of the averaged momentum distributions from the centre and the power law part.



Close-up of centre.

Close-up of tail.

Figure 4.4.: The dissociation with ϵ.

Numerically obtained averaged momentum distributions with 100000 realisations. The different curves have detunings of 0.001 (black squares), here representing the quantum resonance, 0.005 (red circles), 0.01 (blue triangles), 0.05 (pink triangles) and 0.1 (green lozenges).

As expected the curves with the highest peaks were the ones with the largest offset from quantum resonance. But a greater detuning is apparently not equivalent to a greater offset in the tail.

4.4. Fixing the kick strength

To enable the experiment we might need to fix the kick strength during a realisation so here we will study the mean momentum distribution for such a scenario. Simply fixing the kick strength lead to numerical results where the obtained power laws had not the expected exponents, probably because the statistics were not sufficiently large. We tried to solve the problem by choosing random spatial potential shifts (phase shifts of the kick potential) in the measurements. Different forms of phase addition were chosen. We tried uniformly distributed phases for each realisation, for each kick and for each realisation in Θ for the ϵ-classics:

$$\hat{U}'_{\beta,k_j,\phi_j} = e^{-ik_j\cos\left(\hat{\theta}-\phi_j\right)}e^{-\frac{i}{2}\tau(\hat{n}+\beta)^2}$$

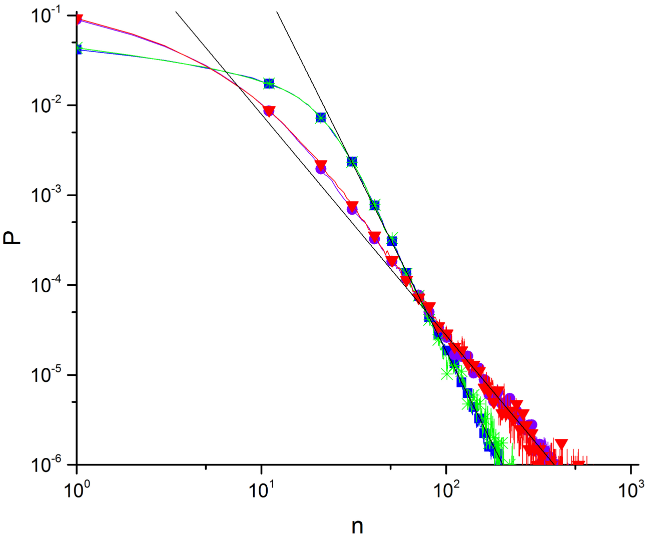


Figure 4.5.: Fixing the kick strength.

This double-logarithmic plot shows the mean momentum distribution for stable distributed kick strengths k ($\alpha = 1.5$) and uniformly distributed spatial potential shifts ϕ between 0 and 2π. The blue squares and green stars show the quantum and pseudo-classical result for constant phases along a realisation. The purple circles and red triangles show the quantum and pseudo-classical result for random phases at each kick. The two power law fits have exponents of $\alpha = 1.46$ and $\alpha = 3.12$.

The quantum map results were always equivalent to the one from the ϵ-classical map. Also the result from the test series with no phases and constant phases along the realisations lead to the same wrong power laws. Only the random phases at each kick lead to the power law corresponding to the chosen distribution of k. So if we want to fix the kick strength in the experiment we must randomly change the spatial potential shifts each kick.

Conclusion

In conclusion this thesis complements the theory of steering random walks in quantum resonance [12] (atom optics kicked rotor) and expands it to the adjacency of quantum resonance to strengthen it (see [16] for a complete theory). The theory: One can steer classical random walks in momentum space for ultracold atoms by choosing at each kick the kicking strength and the spatial potential shift from specific probability distributions. We have shown that the theory is stable with respect to small detunings ϵ in the kick period τ. This stability will show to be helpful for future experimental implementations since detuning constitute an experimental limitation.

From the steering of random walks inducing Lévy walks is especially interesting due to their large field of application like physics, economy and biology. This was achieved by making the probability distributions from which the kick strength were sampled a stable distribution. We demonstrated that like in resonance the averaged momentum distribution obey to the same power law behaviour in the tails like the chosen stable distribution.

Ideally the next step would be to find an analytical solution for the normal and averaged momentum distribution outside of quantum resonance. A more thorough investigation of the dissociation phenomenon and the dependency on ϵ of the offset might be interesting for this.

The theory has also shown to be stable in respect to finite quasimomentum distribution of the atoms [12]. Normally one would have to take into account that the atoms will be an ensemble of plane waves with diverse, partly non-resonant quasimomenta β. The non-resonant quasimomenta act as potential shifts and it was shown that by selecting uniformly distributed spatial potential shifts in $[0, 2\pi)$ their action could be averaged out [9]. This is good since in experiment the quasimomenta of the kicked atoms are able to change via weak spontaneous emission thus giving a broader distribution.

The next challenge is an experimental implementation for which we will give an outlook here and a summary of possible problems (see [16] for more detail). Like for the numerics a large amount of problems come with the heavy-tailed statistic of the kicking strengths. For example it is experimentally not possible to choose the kick strength from an infinite range. Instead one must use distributions with cut-offs thus preventing the proper representation of the tails. Other problems regarding the kick strength are that it is difficult to change the strength each kick which is why it will probably be fixed. As for the numerics to observe a ”nice” power law in the tails the process needs a lot of realisations which means a lot of time.

During this long measuring period the experimental parameters run the risk of drifting away. Also the different atoms in the trap perceive the kick strengths differently because they are not all at the same position. All these effects should be considered when modelling up a realistic experiment like an analogue simulation of classical random walks in Bose-Einstein-condensates.

Fixing the kick strength along a realisation yielded numerically not the expected results for the quantum and ϵ-classical map (too steep power laws). This is probably because by fixing the kick strength but keeping the number of realisations approximately the same the statistic in k is just not great enough. Increasing the number of realisations was not realistic since the computation time would be too high. The problem could be solved by considering random phase shifts at each kick.

A. Fast Fourier Transform

For the Fast Fourier Transform that we use in our simulations we used the four1-routine [17]. The FFT is a discrete (using a finite number of sampled points) Fourier transform called fast because of its algorithmic efficiency of O(nlog$_2$(n)) to the usual O(n^2), where n is the size of the input.

Danielson-Lanczos lemma [17]:
A discrete Fourier Transform of length N can be expressed as the sum of the Fourier transforms of the even numbered points and the one of the odd numbered points.

$$\begin{aligned}
F_k &= \sum_{j=0}^{N-1} e^{2\pi i \frac{jk}{N}} f_j \\
&= \sum_{j=0}^{\frac{N}{2}-1} e^{2\pi i \frac{(2j)k}{N}} f_{2j} + \sum_{j=0}^{\frac{N}{2}-1} e^{2\pi i \frac{(2j+1)k}{N}} f_{2j+1} \\
&= \sum_{j=0}^{\frac{N}{2}-1} e^{2\pi i \frac{jk}{\frac{N}{2}}} f_{2j} + W^k \sum_{j=0}^{\frac{N}{2}-1} e^{2\pi i \frac{jk}{\frac{N}{2}}} f_{2j+1} \\
&= F_k^e + W^k F_k^o
\end{aligned}$$

Where f_j, F_k, F_k^e and F_k^o are respectively the j-th component of the original input of length N, the k-th component of the Fourier transform of length N and the k-th components of the Fourier transform of length $\frac{N}{2}$ of the even and odd components. W is the complex number:

$$W = e^{\frac{2\pi i}{N}}$$

This can be used recursively until we have a sum of Fourier transforms of length one which are just the identity operations. To assign the different patterns A of o's and e's in $F_k^{A(o,e)} = f_j$ to the associated j, we reverse the pattern A and replace $e \to 0$ and $o \to 1$, which corresponds to a bit-reversal. The obtained binary pattern $A(0,1)$ is then the value j of the f_j.

For our purposes we have to reorder by switching the order of the halves of the outcoming array (FFT-shift) after the transformation from (angular) momentum space to (angular) position space and when transforming back because our momenta n are centred around 0 and our angles θ around π.

This is the code of four1:

```
/***************************************************************************
Fast Fourier transform program, four1, from "Numerical Recipes in C" (Cambridge
Univ. Press) by W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery
***************************************************************************/

#define SWAP(a,b) tempr=(a);(a)=(b);(b)=tempr

/**************************************************************************/
void four1(double data[], unsigned long nn, int isign)
/***************************************************************************
Replaces data[1..2*nn] by its discrete Fourier transform, if isign is input as
1; or replaces data[1..2*nn] by nn times its inverse discrete Fourier transform,
if isign is input as -1. data is a complex array of length nn or, equivalently,
a real array of length 2*nn. nn MUST be an integer power of 2 (this is not
checked for!).
**************************************************************************/
{
  unsigned long n,mmax,m,j,istep,i;
  double wtemp,wr,wpr,wpi,wi,theta;
  double tempr,tempi;

  n=nn << 1;
  j=1;
  for (i=1;i<n;i+=2) { /* This is the bit-reversal section of the routine. */
    if (j > i) {
      SWAP(data[j],data[i]); /* Exchange the two complex numbers. */
      SWAP(data[j+1],data[i+1]);
    }
    m=nn;
    while (m >= 2 && j > m) {
      j -= m;
      m >>= 1;
    }
    j += m;
  }

  mmax=2;
  while (n > mmax) { /* Outer loop executed log2 nn times. */
    istep=mmax << 1;
    theta=isign*(6.28318530717959/mmax); /* Initialize the trigonometric recurrence. */
    wtemp=sin(0.5*theta);
    wpr = -2.0*wtemp*wtemp;
    wpi=sin(theta);
    wr=1.0;
    wi=0.0;
    for (m=1;m<mmax;m+=2) { /* Here are the two nested inner loops. */
      for (i=m;i<=n;i+=istep) {
        j=i+mmax; /* This is the Danielson-Lanczos formula. */
        tempr=wr*data[j]-wi*data[j+1];
        tempi=wr*data[j+1]+wi*data[j];
        data[j]=data[i]-tempr;
        data[j+1]=data[i+1]-tempi;
        data[i] += tempr;
        data[i+1] += tempi;
      }
      wr=(wtemp=wr)*wpr-wi*wpi+wr; /* Trigonometric recurrence. */
      wi=wi*wpr+wtemp*wpi+wi;
    }
    mmax=istep;
  }
}
```

B. Stable distributions

Alpha-stable or stable distributions are a type of probability distributions which are special due to their heavy tails and invariance under convolution. Their theory was first shaped by Paul Lévy in the early 20th century. Here is a short summary of the most important aspects (for this thesis) of stable distributions [18, 19]:

Definition of stable :
A random variable X is called stable (in the broad sense) if independent copies X_1 and X_2 of X with constant parameters $a, b > 0$ satisfy the following relation

$$aX_1 + bX_2 \overset{\mathcal{D}}{=} cX + d$$

for some $c > 0$ and $d \in \mathbb{R}$. Here ”$\overset{\mathcal{D}}{=}$” stands for equal in distribution meaning that both sides follow the same probability law. The distribution is stable symmetric if in addition it is symmetrically distributed around 0.

The stable distribution $S(\alpha, \beta, \gamma, \delta)$ depends on four real parameters: the stability parameter $\alpha \in (0, 2]$ that characterizes the decrease in the tails, the skewness parameter $\beta \in [-1, 1]$, the scale parameter $\gamma > 0$ and finally the location parameter δ. A great inconvenience of the stable distributions is that they may generally not be expressed analytically except for the 3 following special cases:

1. The Gaussian distribution $G(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{(x-\mu)^2}{2\sigma^2}\Big)$ with $S(2, \beta, \frac{\sigma}{\sqrt{2}}, \mu)$
2. The Cauchy distribution $C(x; \mu, \gamma) = \frac{1}{\pi\gamma(1+(\frac{x-\mu}{\gamma})^2)}$ with $S(1, 0, \gamma, \mu)$
3. The Lévy distribution $L(x; \mu, \gamma) = \sqrt{\frac{\gamma}{2\pi}} \frac{\exp\big(-\frac{\gamma}{2(x-\mu)}\big)}{(x-\mu)^{\frac{3}{2}}}$ with $S(\frac{1}{2}, 1, \gamma, \mu)$

With exception of the Gaussian ($\alpha = 2$) stable distributions $p(x)$ possess inverse power law-tails:

$$p(x) \underset{|x|\to\infty}{\propto} \frac{1}{|x|^{1+\alpha}} \quad 0 < \alpha < 2$$

The sum X of two stable distributed variables $X_1 \propto S(\alpha, \beta_1, \gamma_1, \delta_1)$ and $X_2 \propto S(\alpha, \beta_2, \gamma_2, \delta_2)$ is again stable distributed, $X_1 + X_2 = X \propto S(\alpha, \beta, \gamma, \delta)$ with:

$$\beta = \frac{\beta_1\gamma_1^\alpha + \beta_2\gamma_2^\alpha}{\gamma_1^\alpha + \gamma_2^\alpha}, \quad \gamma = (\gamma_1^\alpha + \gamma_2^\alpha)^{\frac{1}{\alpha}}, \quad \delta = \delta_1 + \delta_2$$

C. Random Number Generation

The random numbers from ”common” distributions like uniform and normal are generated with the <random> header of the Standard Library (C++11-standard [20]) which offers a selection of generators and distributions. The generators or random number engines produce pseudo random values after being given an initializing randomness in form of a seed. We chose to base the seed creation on time. For this we included <chrono> and generated seeds from the number of seconds passed since ”Wed Dec 31 19:00:00 1969” to the moment of execution of the program. The generated random numbers can then be mapped on an interval following the various distributions. The following code shows the production of normal distributed numbers:

```
unsigned seed = std::chrono::system_clock::now().time_since_epoch().count();
std::default_random_engine generator (seed);
normal_distribution<double> distribution(0.0,1.0);
distribution(generator);
```

To create stable random numbers we make use of the following theorem.

Theorem: Simulating stable random variables [18, 21]
Let θ and W be independent with θ uniformly distributed on $(-\frac{\pi}{2},\frac{\pi}{2})$ and W exponentially distributed with mean 1. Also let $0 < \alpha \leq 2$. Then it holds that:
(a) The symmetric random variable

$$Z = \begin{cases} \frac{\sin(\alpha\theta)}{\cos(\theta)^{\frac{1}{\alpha}}} \left[\frac{\cos((\alpha-1)\theta)}{W}\right]^{\frac{1-\alpha}{\alpha}} & \alpha \neq 1 \\ \tan(\theta) & \alpha = 1 \end{cases}$$

has a $S(\alpha, 0, 1, 0)$ distribution.
(b) In the non symmetric case, for any $-1 \leq \beta \leq 1$, define $\theta_0 = \arctan\left(\frac{\beta\tan(\frac{\pi\alpha}{2}))}{\alpha}\right)$ when $\alpha \neq 1$.

$$Z = \begin{cases} \frac{\sin(\alpha(\theta+\theta_0))}{(\cos(\alpha\theta_0)\cos(\theta))^{\frac{1}{\alpha}}} \left[\frac{\cos((\alpha-1)\theta+\alpha\theta_0)}{W}\right]^{\frac{1-\alpha}{\alpha}} & \alpha \neq 1 \\ \frac{2}{\pi}\left[\left(\frac{\pi}{2}+\beta\theta\right)\tan(\theta) - \beta\ln\left(\frac{\frac{\pi}{2}W\cos(\theta)}{\frac{\pi}{2}+\beta\theta}\right)\right] & \alpha = 1 \end{cases}$$

has a $S(\alpha, \beta, 1, 0)$ distribution.

θ and W can easily be achieved by computing:

$$\begin{aligned} \theta &= \pi(x - 0.5) \\ W &= -\ln(y) \end{aligned}$$

whereby x and y represent random numbers drawn from a uniform distribution of real numbers between 0 and 1. This translates into the following simple code:

```
unsigned seed = std::chrono::system_clock::now().time_since_epoch().count();
std::default_random_engine generator (seed);
uniform_real_distribution<double> distribution(0.0,1.0);

double stable(float alpha){

  double theta;
  double W;
  double result;

  theta=3.14159265358979323846*(distribution(generator)-0.5);
 W=-log(distribution(generator));

  result=(sin(alpha*theta)/pow(cos(theta),1.0/alpha))
  *pow(cos(theta*(1.0-alpha))/W,((1.0-alpha)/alpha));
  return(result);
}
```

D. Example code

The subsequent C++-code executes the in (3.1) described Monte Carlo method to compute the averaged momentum distribution P, here for symmetric stable distributed k with $\alpha = 1.5$, $\epsilon = 0.01$, $T = 50$ and $R = 1000000$.

```cpp
#include <iostream>
#include <math.h>
#include <fstream>
#include <random>
#include <chrono>
using namespace std;

int T;
double PSI[65536][2][100];
double P[65536][100];
double psy[131072];
double psyy[131072];
double MeanP[65536];
double R;
double k;
double tau;
double epsilon

int main(){

ofstream fout(”MeanP.txt”);

epsilon=0.01;
T=50;
R=1000000.0;
PSI[32768][0][0]=1.0;
tau=4*3.14159265358979323846+epsilon;

for(int r=0;r<R+1;r++){

  for(int t=0;t<T+1;t++){
    k=stable(1.5);

//apply free evolution
    for(int j=0; j<65536; j++ ){
      psy[2*j]=cos(-0.5*tau*(j-32768)*(j-32768))*PSI[j][0][t]
      -sin(-0.5*tau*(j-32768)*(j-32768))*PSI[j][1][t];
      psy[2*j+1]=sin(-0.5*tau*(j-32768)*(j-32768))*PSI[j][0][t]
      +cos(-0.5*tau*(j-32768)*(j-32768))*PSI[j][1][t];
    }

//Fourier Transform to angular space
    four1(psy-1,65536,1);

//Reordering the array (FFT-shift)
    for(int i=0; i<65536; i++ ){
      psyy[65536+i]=psy[i];
      psyy[i]=psy[65536+i];
    }

//Computing the the action of the kick
    for(int i=0; i<65536; i++ ){
      psy[2*i]=cos(-k*cos(i*2.0*3.14159265358979323846/65536.0))*psyy[2*i]-sin(-k*cos(i*2.0*3.14159265358979323846/65536.0))*psyy[2*i+1];
      psy[2*i+1]=sin(-k*cos(i*2.0*3.14159265358979323846/65536.0))*psyy[2*i]+cos(-k*cos(i*2.0*3.14159265358979323846/65536.0))*psyy[2*i+1];
    }

//Ordering the array back (FFT-shift)
    for(int i=0; i<65536; i++ ){
      psyy[65536+i]=psy[i];
      psyy[i]=psy[65536+i];
    }

    for(int i=0; i<131072; i++ ){
      psy[i]=psyy[i];
    }

//Going back to angular momentum space
    four1(psy-1,65536,-1);
    for(int j=0; j<65536; j++ ){
      PSI[j][0][t+1]=psy[2*j]/65536.0;
      PSI[j][1][t+1]=psy[2*j+1]/65536.0;
    }
  }
//Computing distribution of momenta
  for(int j=0; j<65536; j++ ){
    P[j][50]=PSI[j][0][50]*PSI[j][0][50]+PSI[j][1][50]*PSI[j][1][50];
  }

  for(int j=0; j<65536; j++ ){
    MeanP[j]+=P[j][50];
  }
}

fout<<"n"<<"   "<<"MeanP"<<endl;
for(int j=0; j<65536; j++ ){
  MeanP[j]=MeanP[j]/R;
  fout<<j-32768<<"   "<<MeanP[j]<<endl;
}

return 0;
}
```

And here the corrresponding MATLAB-code for the ϵ-classical simulation with $\Theta = 100$:

```
clear all
%clear figure

eps = 0.01;
ell=2.;
tau = 2*pi*ell + eps;
p_00=0.;
%%% Bin number for log-log plot:
Nbin=30;
NR=1000000;
N=100;
NT=50;
alpha=1.5;
mean=0.;
gamma=0.5;

%%% phase randomization:
% phi=random('unif',0,2*pi,NR,NT);
%%%% Approach to QR is tested and ok for eps -> 0:
% sk = ones(NR);
%%% k=u;
%sk = stblrnd(alpha,0,gamma,0,NR,NT);

%%% Starting momentum distribution in the first Brillouinzone:
% beta = random('normal',0,0.01,NR,1);
%%% or:
% beta = 0.;

th_0 = random('unif',0,2*pi,N,1);

% sk=abs(sk*eps);
u=random('unif',0,1,NR,NT);
sk=gamma * tan(pi*(u-0.5)); % +mean;
sk=sk*eps;
```

```

If = zeros(N*NR,1);

for i=1:NR

  for l=1:N

    %%% equidistant values of theta:

    % th_0 = l*2*pi/N;

    th=th_0(l); p=p_00;

    for j=1:NT
      p = p + sk(i,j) * sin(th);
      th = th + sign(eps)*p; %+ tau*beta(i);
      %%% superfluous for ell=2:
      % + pi*ell
      th=mod(th,2*pi);
      % + sk(i,j) * sin(th+phi(i,j));
    end
    ii = l+(i-1)*N;
    %%%% test approach QR:
    % If(ii) = p/abs(eps);
    If(ii) = abs(p/eps);

  end

end

fileID=fopen('file.txt','w');
fprintf(fileID, '%f \n',[If]');
fclose(fileID);

%hist(If,35);set(gca,'Xscale','log','Yscale','log');
%axis([2.^1 2.^Nbin 10^(-6) 1000.05])
%figure(2)
%x=linspace(0,10,Nbin);Xexp=2.^x; C=hist(If,Xexp);
%sum=0;
%for i=1:Nbin-1
%sum = sum + C(i)*(Xexp(i+1)-Xexp(i));
%end
%plot(Xexp,C/sum);
%set(gca,'Xscale','log','Yscale','log'); axis([Xexp(1) Xexp(Nbin) 10^(-6) 0.1]);
%hist(If,25);
hold on
```

Bibliography

[1] B. V. Chirikov, *Research concerning the theory of nonlinear resonance and stochasticity.* Preprint N 267, Institute of Nuclear Physics, Novosibirsk, 1969.

[2] F. Izrailev, *Simple models of quantum chaos: spectrum and eigenfunctions.* Phys. Rep. 196, 299, 1990.

[3] Moore F. L., Robinson J. C., Bharucha C. F., Williams P. E. and Raizen, *Observation of dynamical localization in atomic momentum transfer: a new testing ground for quantum chaos.* Phys. Rev. Lett. 73 2974, 1994.

[4] I. Guarneri, L. Rebuzzini, and S. Fishman, *A Theory for Quantum Accelerator Modes in Atom Optics.* Journal of Statistical Physics, Vol. 110, 2003.

[5] Michael F. Shlesinger, George M. Zaslavsky, Uriel Frisch, *Lévy Flights and Related Topics in Physics.* Springer Verlag, Berlin-Heidelberg, 1994.

[6] Wolfgang Paul, Jörg Baschnagel, *Stochastic Processes - From Physics to Finance.* Springer Verlag, Berlin-Heidelberg, 2000.

[7] Mark Sadgrove and Sandro Wimberger, *A Pseudoclassical Method for the Atom-Optics Kicked Rotor: from Theory to Experiment and Back.* Adv. At. Mol. Phys. 60, p. 315-369, 2011.

[8] C. Ryu, M. F. Andersen, A. Vaziri, M. B. d'Arcy, J. M. Grossman, K. Helmerson, W. D. Phillips, *High-order quantum resonances observed in a periodically kicked bose-einstein condensate.* Phys. Rev. Lett. 96, 160403, 2006.

[9] S. Wimberger, I. Guarneri, and S. Fishman *Quantum resonances and decoherence for delta-kicked atoms.* Nonlinearity 16, p 1381-1420, 2003.

[10] Y. Aharonov, L. Davidovich and N. Zagury, *Quantum random walks.* Phys. Rev. A, 48, 1993.

[11] J. Kempe, *Quantum random walks - an introductory overview.* Contemporary Physics 44 (4), p. 307-327, 2003.

[12] Marcel Weiß, *Steering Random Walks through Quantum Resonance.* Bachelor Thesis, 2014.

[13] Sandro Wimberger, *Nonlinear Dynamics and Quantum Chaos - An Introduction.* Springer, 2014.

[14] B. V. Gnedenko and A. N. Kolmogorov *Limit distributions for sums of independent random variables.* Addison-Wesley, 1954.

[15] Sandro Wimberger, *Chaos and Localisation: Quantum Transport in Periodically Driven Atomic Systems.* Ph.D. Thesis, 2004.

[16] Marcel Weiß, Caspar Groiseau, W. K. Lam, Raffaella Burioni, Alessandro Vezzani, Gil S. Summy, Sandro Wimberger *Steering random walks with kicked ultracold atoms.* Phys. Rev. A 92, 033606, 2015.

[17] William. H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, *NUMERICAL RECIPES - The Art of Scientific Computing, Third Edition.* Cambridge University Press, New York, 2007.

[18] J. P. Nolan, *Stable Distributions - Models for Heavy Tailed Data.* In progress, Chapter 1 online at academic2.american.edu/∼jpnolan, 2015.

[19] A. Klemke, *Wahrscheinlichkeitstheorie.* Springer-Verlag Berlin Heidelberg, 2013.

[20] B. Stroustrup, *Die C++-Programmierssprache - Aktuell zum C++11-Standard.* Hanser, München, 2015.

[21] J. M. Chambers, C. L. Mallows and B. W. Stuck, *A Method for Simulating Stable Random Variables.* Journal of the American Statistical Association, Volume 71, Number 354, 1976.

Erklärung

Ich versichere, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, den 10. Juli 2015,