Discrete Quantum Walks on 1D and 2D Lattices

Sudarsan Balakrishnan

May 1, 2017

Abstract

This report discusses my study of discrete-time quantum walks on a line(DQWL)[1], and on a rectangular 2D lattice. Simulations using Python helped track the probability distribution of the walker for walks with various initial states. Analytical verification of the results using directly derived expressions was carried out for at least one case with the simplest initial conditions. The simulations closely matched published results in all the considered cases, and were ensured to be numerically stable.

1 Classical Random Walk

In its most common version, the classical random walk(CRW) has a moving particle(called 'walker') on a discrete 1-d lattice, take a step forward(with probability p) or backward(with probability q) in the lattice according to the outcome of a coin-toss at every time step. At the continuum limit, the walk reduces to the diffusion equation. The probability after 'n' time-steps that the walker is found at lattice position $Z_n = k$, after starting at $Z_0 = 0$ is given by the following equation[2]

$$Prob(Z_n = k | Z_0 = 0) = \binom{n}{\frac{1}{2}(k+n)} p^{\frac{1}{2}(k+n)} q^{\frac{1}{2}(n-k)} \tag{1}$$

which is subject to the condition that $\frac{1}{2}(k+n)$ is a non-negative integer. In all other cases involving other lattice points, the probability is zero. This means that in odd(even) time steps, the probability of finding the walker in even(odd) lattice positions is zero. This arises from the walker having only a binary choice, both of which leads to his movement to the next lattice site of opposite parity. The mean position of the walker after n time-steps is given by $n(2p-1)$, and the variance by $4npq$. For the $p = 1/2$ case, this represents a binomial distribution centered at $k = 0$, whose variance increases with time as $\mathbb{O}(n)$, or $\mathbb{O}(t)$. This implies that the RMS position of the walker increases with time as $\mathbb{O}(\sqrt{t})$, the mean position for the unbiased classical random walk being the origin.

2 Quantum Random Walk in 1-D

In the quantum analog of the random walk in one dimension, the walker 'throws' a quantum coin at every time-step, whose outcome determines his direction of movement. Since quantum systems support non-classical superposition states and because quantum coherence is maintained throughout the evolution, the outcome of the quantum random walk is very different qualitatively from the classical picture. To model DQWL on a 1-D lattice whose positions are labeled from $-n$ to n, we define the following:

- The state-space of the system lies in the direct product space of two subspaces - one for the **coin**($\mathcal{H}_c$), and one for the **lattice**($\mathcal{H}_x$).

- A general state $|\psi\rangle$ of the system can be written as $\sum_{c=0}^{1}\sum_{x=-n}^{n}\alpha_{c,x}|c\rangle|x\rangle$, where the coefficients satisfy

$$\sum_{c=0}^{1}\sum_{x=-n}^{n}|\alpha_{c,x}|^2 = 1 \tag{2}$$

- During a time-step, two operations are performed: 1)Toss of a coin, performed by a Hadamard operator which acts only on the coin-space, keeping the lattice states unchanged, leading to an operator of the form $\hat{T} = (\hat{C_c^H} \otimes \hat{I_x})$ and 2)Shift of the walker's position in the lattice, conditional to the outcome of the coin toss - this is performed by the 'shift' operator $\hat{S}$.

- The unitary time-evolution operator $\hat{U} = (\hat{S})\dot{(}\hat{C_c^H} \otimes \hat{I_x})$ represents a time-step, giving $|\psi_{t+1}\rangle = \hat{U}|\psi_t\rangle$ for the time-evolution of the system during each step. The large-time behavior is found by repeating the unitary operation over many steps.

- The expanded form of $(\hat{C_c^H})$ and $(\hat{S})$ is written as follows -

$$\hat{C_c^H} = \frac{1}{\sqrt{2}}(|0\rangle_c\langle 0|+|0\rangle_c\langle 1|+|1\rangle_c\langle 0|-|1\rangle_c\langle 1|) \tag{3}$$

$$\hat{S} = |0\rangle_c\langle 0|\otimes \sum_{x'=-n}^{n}|x'-1\rangle_x\langle x'|+|1\rangle_c\langle 1|\otimes \sum_{x'=-n}^{n}|x'+1\rangle_x\langle x'| \tag{4}$$

As seen during the time evolution of the system after a sufficiently large number of steps, the probability of finding a particle at lattice-point 'x' is low near $x = 0$, and is peaked about the periphery. The outcome is seen to depend on the initial state of the coin used, and the nature of the coin thrown(the Hadamard coin, is one of many possible quantum two-level systems suitable for our application). If the initial state of the coin is a classical state($|0\rangle_c$ or $|1\rangle_c$), then the long-time outcome of the walk is singly peaked in that direction. Upon using a symmetric initial state for the coin(a quantum superposition state of the form $\frac{|0\rangle_c+i|1\rangle_c}{\sqrt{2}}$), a roughly bimodal distribution is observed, peaking at two spots symmetrically placed with respect to the origin[1, 2].

3 Numerical Implementation of 1-D Quantum Walk

3.1 Design considerations

The state of the system was represented in Python using complex vectors, the operators being complex matrices. A given state at time 't' could be represented as

$$|\Psi(\vec{x},t)\rangle = \begin{bmatrix}\vec{\Psi_L}(\vec{x},t)\\ \vec{\Psi_R}(\vec{x},t)\end{bmatrix} \tag{5}$$

where

$$\vec{\Psi_L}(\vec{x},t) = \begin{bmatrix}\alpha_{c=0,x=-n}(t)\\ \alpha_{c=0,x=-n+1}(t)\\ \vdots\\ \alpha_{c=0,x=n-1}(t)\\ \alpha_{c=0,x=n}(t)\end{bmatrix}, \vec{\Psi_R}(\vec{x},t) = \begin{bmatrix}\alpha_{c=1,x=-n}(t)\\ \alpha_{c=1,x=-n+1}(t)\\ \vdots\\ \alpha_{c=1,x=n-1}(t)\\ \alpha_{c=1,x=n}(t)\end{bmatrix} \tag{6}$$

are $N = 2n + 1$ dimensional vectors on the lattice, storing the probability amplitudes. Without losing generality, we can choose the toss-outcome $c = 0(1)$ to correspond to a leftward(rightward) movement of the walker.

Each operator($\hat{T}, \hat{S}$) acting on the state has $2N \times 2N$ elements for the 1D case, for which we can write closed expressions as given below.

$$\hat{T} = (\hat{C_c^H} \otimes \hat{I}_x) = \frac{1}{\sqrt{2}} \begin{bmatrix} \hat{I}_{n\times n} & \hat{I}_{n\times n} \\ \hat{I}_{n\times n} & -\hat{I}_{n\times n} \end{bmatrix}, \hat{S} = \begin{bmatrix} |i-1\rangle_x\langle i| & \hat{0} \\ \hat{0} & |i+1\rangle_x\langle i| \end{bmatrix} \tag{7}$$

where $\hat{I}_{n\times n}$ and $\hat{0}$ are the identity and zero matrices of dimension $n \times n$, and $|i \pm 1\rangle_x\langle i|$ stands for the corresponding shift of all indices in the lattice-space in the specified direction. The time-step operator $\hat{U}$ can be obtained by taking the matrix product of $\hat{S}$ and $\hat{T}$.

Since for our purposes, we use a finite basis to represent an infinite-dimensional lattice, we need to be careful to not let the system evolve beyond a point where $|i \pm 1\rangle_x\langle i|$ in $\hat{S}$ leads to states outside the finite, N-dimensional lattice Hilbert-space used. In all the simulations performed, this was ensured by keeping the classical 'worst-case' scenario(the walker advances by one lattice-step in a fixed direction in every step of 't') in mind, which restricted the time-steps from $t = 0$ to $t = n$. Verifying completeness 2 at each time-step was performed to guard against numerical errors.

The most relevant observable for the system is the probability of finding the particle at lattice-site 'x' in each time-step, which can be computed from the general state as

$$Prob(x,t) = |\alpha_{0,x}(t)|^2 + |\alpha_{1,x}(t)|^2 \tag{8}$$

The probability distribution thus obtained was dynamically plotted over the lattice-position at each time-step to study the time-evolution of the system. The following initial states were chosen in this regard - $|\Psi(0)\rangle = |0\rangle_c \otimes |0\rangle_x, |1\rangle_c \otimes |0\rangle_x, \frac{|0\rangle_c + i|1\rangle_c}{\sqrt{2}} \otimes |0\rangle_x$. The walker starts at the zero-position of the lattice with 100% probability in each case, with the coin initially being in classical left or right states(the first two cases), or on a uniform superposition(the third case).

To implement these details in Python, an $2N \times 1$ numpy array of complex type was used to store the initial state of the system. The operators $\hat{T}$ and $\hat{S}$ were $2N \times 2N$ complex numpy arrays, storing matrices. In each time-step, $\hat{S}\hat{T}$ was multiplied to the state, following which its magnitude was checked and verified to be 1. Subsequently, the occupation probabilities were extracted and stored in a real $N \times 1$ numpy array which was plotted in each step.

The unitariness of the operators used(seen via their determinants), and the completeness of the states employed, were persistent checks used to look out for possible numerical errors. Numpy and Matplotlib were used to implement the arrays required, as well as to visualize the outcomes.

3.2 Output and analysis

- For the initial state $|\Psi(0)\rangle = |0\rangle_c \otimes |0\rangle_x$ with the coin in a 'left' state at time-step 0, the output is peaked about the left, with a nearly uniform distribution near $x = 0$. For the initial state $|\Psi(0)\rangle = |1\rangle_c \otimes |0\rangle_x$ with the coin in the 'right' state initially, the output is peaked about the right, with an identical profile as the previous case upto a reflection about the origin - consistent with the symmetry in the system.

- For the symmetric initial state $\frac{|0\rangle_c + i|1\rangle_c}{\sqrt{2}} \otimes |0\rangle_x$ with the coin in a uniform superposition, the output is peaked symmetrically about both sides of the origin, covering an identical distance over time as both the cases considered previously, peaking at $x = \pm 68$ for time-step $t = 100$.

- In every odd(even) time-step, the occupation probability of even(odd) lattice sites goes to zero, as expected. In the plots shown, only the non-zero probabilities are plotted for a lattice of size $n = 100$. The results are in agreement with published results, and the sum of occupation probabilities is verified to be 1 in every time-step. The simulation is run until time-step 't' becomes 100, in keeping with the consideration of the classical worst-case scenario.
- In the plots shown, I compare the results of my own simulation with the results published by Venegas[2] - the two are in agreement, as anticipated from all the numerical consistency checks being met.

4 Deriving an analytical expression for the 1D random walk

- In order to derive an analytical expression that verifies the simulations are in order, we use a two-component state(borrowing the notation in 5) $\Psi(x,t) = \begin{bmatrix}\Psi_L(x,t)\\ \Psi_R(x,t)\end{bmatrix}$ associated with a single lattice-position $|x\rangle$.(This analysis is originally due to Nayak and Vishwanath[3], and is also repeated in [2, 4] and other sources.)
- During every coin toss, the two components are mixed by the Hadamard operator 3. This behavior is identical for *each* position x, which means we can describe the evolution of the entire system's distribution using this formalism.
- We can now write

$$\Psi(x,t+1) = \mathbf{M}_+\Psi(x-1,t) + \mathbf{M}_-\Psi(x+1,t). \tag{9}$$

- Here, $\mathbf{M}_+ = \begin{bmatrix}0 & 0\\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}}\end{bmatrix}$ and $\mathbf{M}_- = \begin{bmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ 0 & 0\end{bmatrix}$ are derived from the Hadamard operator $\hat{C}_c^H$. A physical interpretation could be that the walker gets to lattice-site x in time-step $t+1$ by either - a *left-step* from lattice-site $x+1$, or a *right-step* from $x-1$.
- To continue the analysis, we move to the Fourier transform space for the discrete spatial lattice, writing $\tilde{\Psi}(k,t) = \sum_x \Psi(x,t)e^{ikx}$. This transform is derived from the Discrete-time Fourier Transform(DTFT), and maps a discrete x-space to a continuous k-space.
- The inverse Fourier transform for the above operation takes the form

$$\Psi(x,t) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \tilde{\Psi}(k,t)e^{-ikx}dk \tag{10}$$

  where we retain the analogy to DTFT.
- We consider the solution for one of the three initial conditions chosen, with $|\Psi(0)\rangle = |0\rangle_c \otimes |0\rangle_x$. In this case, the initial state vector takes the form $\Psi(x,t=0) = \begin{bmatrix}\delta_{0x}\\ 0\end{bmatrix}$, which fixes $\tilde{\Psi}(k,0) = \begin{bmatrix}1\\ 0\end{bmatrix}$ $\forall k \in \mathbb{R}$ as the initial state in the k-space.

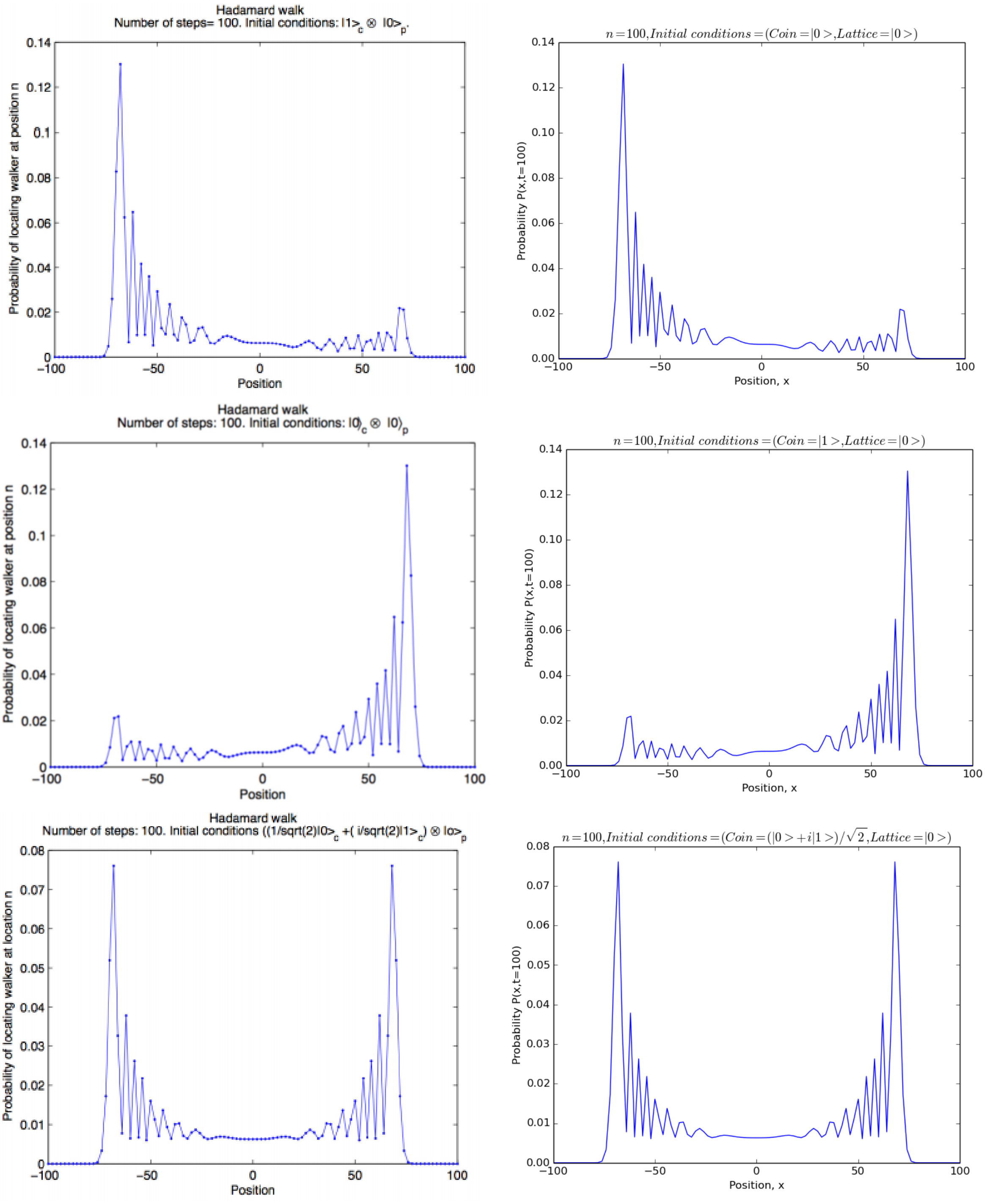

Figure 1: Comparison of final outputs for simulation of a quantum walk on a 1-D lattice, having lattice-parameters $n = 100$ and $N = 201$ at time-step $t = 100$. My plots are on the right side, the plots on the left are due to Venegas[2], who uses a different convention to map the coin-space with the direction of the walk.

- Applying Fourier transform in equation 9, we obtain

$$\tilde{\Psi}(k,t+1) = e^{ik}\mathbf{M}_{+}\sum_{x}\Psi(x-1,t)e^{ik(x-1)} + e^{-ik}\mathbf{M}_{-}\sum_{x}\Psi(x+1,t)e^{ik(x+1)} \quad (11)$$

$$\tilde{\Psi}(k,t+1) = (e^{ik}\mathbf{M}_{+} + e^{-ik}\mathbf{M}_{-})\tilde{\Psi}(k,t) \quad (12)$$

which is a single-step recursion relation in the t-space.

- Thus, we obtain a closed expression for the transformation in one time-step, as $\tilde{\Psi}(k,t+1) = \mathbf{M}_k\tilde{\Psi}(k,t)$, where $\mathbf{M}_k = \frac{1}{\sqrt{2}}\begin{bmatrix} e^{-ik} & e^{-ik} \\ e^{ik} & -e^{ik} \end{bmatrix}$.
  $\implies \tilde{\Psi}(k,t) = {\mathbf{M}_k}^t\tilde{\Psi}(k,0)$.

- We can compute ${\mathbf{M}_k}^t$ by diagonalizing $\mathbf{M}_k$, and thus calculate the two components $\tilde{\Psi}_L(k,t), \tilde{\Psi}_R(k,t)$.

- The eigenvalues of matrix $\mathbf{M}_k$ are found by solving the secular equation, which leads to the quadratic form

$$\lambda^2 + \sqrt{2}i\lambda sin(k) - 1 = 0 \quad (13)$$

  which upon solution leads to the eigenvalues

$$\lambda_1 = e^{-i\omega_k}, \lambda_2 = e^{i\omega_k+\pi} \quad (14)$$

  for $\mathbf{M}_k$, and λ_1^t, λ_2^t for ${\mathbf{M}_k}^t$, where $sin(\omega_k) = \frac{sin(k)}{\sqrt{2}}$ $\forall k \in \mathbb{R}$.

- The eigenvectors of ${\mathbf{M}_k}^t$ and $\mathbf{M}_k$ come out to be of the form

$$|\tilde{\Psi}_1\rangle = N\begin{bmatrix} e^{-ik} \\ \sqrt{2}e^{-i\omega_k} - e^{-ik} \end{bmatrix}, |\tilde{\Psi}_2\rangle = M\begin{bmatrix} e^{-ik} \\ -\sqrt{2}e^{i\omega_k} - e^{-ik} \end{bmatrix} \quad (15)$$

  with $N = \frac{1}{\sqrt{2((1+cos^2k) - cos(k)\sqrt{1+cos^2(k)})}}$ and $M = \frac{1}{\sqrt{2((1+cos^2k) + cos(k)\sqrt{1+cos^2(k)})}}$

- The action of ${\mathbf{M}_k}^t$ on $|\tilde{\Psi}(k,0)\rangle$ can now be simplified by writing ${\mathbf{M}_k}^t$ in its normal form as

$${\mathbf{M}_k}^t = e^{-i\omega_k t}|\tilde{\Psi}_1\rangle\langle\tilde{\Psi}_1| + e^{i(\omega_k+\pi)t}|\tilde{\Psi}(k,0)\rangle\langle\tilde{\Psi}_2| \quad (16)$$

- We can now write the state of the system at any time-step ‘t’ in k-space as

$$\tilde{\Psi}(k,t) = e^{-i\omega_k t}\langle\tilde{\Psi}_1|\tilde{\Psi}(k,0)\rangle|\tilde{\Psi}_1\rangle + e^{i(\omega_k+\pi)t}\langle\tilde{\Psi}_2|\tilde{\Psi}(k,0)\rangle|\tilde{\Psi}_2\rangle \quad (17)$$

- Upon evaluation, this leads to the closed expressions given below:

$$\tilde{\Psi}_L(k,t) = \frac{1}{2}\Big(\Big(1 + \frac{cos(k)}{\sqrt{1+cos^2(k)}}\Big)e^{-i\omega_k t} + (-1)^t\Big(1 - \frac{cos(k)}{\sqrt{1+cos^2(k)}}\Big)e^{i\omega_k t}\Big) \quad (18)$$

$$\tilde{\Psi}_R(k,t) = \frac{e^{ik}}{2}\Big(\frac{e^{-i\omega_k t} - (-1)^t e^{i\omega_k t}}{\sqrt{1+cos^2(k)}}\Big) \quad (19)$$

- Analytic expressions in position-space are recovered by performing the inverse Fourier transform10, giving the final, closed expressions for the wavefunction components along the lattice as follows -

$$\begin{aligned}\Psi_L(x,t) &= \frac{1}{2\pi}\int_{-\pi}^{\pi}\tilde{\Psi}_L(k,t)dk \\ &= (\frac{1+(-1)^{x+t}}{4\pi})\int_{-\pi}^{\pi}(1+\frac{cos(k)}{\sqrt{1+cos^2(k)}})e^{-i(\omega_k t+kx)}dk\end{aligned} \tag{20}$$

$$\begin{aligned}\Psi_R(x,t) &= \frac{1}{2\pi}\int_{-\pi}^{\pi}\tilde{\Psi}_R(k,t)dk \\ &= (\frac{1+(-1)^{x+t}}{4\pi})\int_{-\pi}^{\pi}\frac{e^{ik}}{\sqrt{1+cos^2(k)}}e^{-i(\omega_k t+kx)}dk\end{aligned} \tag{21}$$

  with $\omega_k \in [-\frac{\pi}{2},\frac{\pi}{2}]$, such that $sin(\omega_k) = \frac{sin(k)}{\sqrt{2}}$.

- The integrals described above were split into their real and imaginary components, and were integrated using `scipy.integrate.quad()` to check for correspondence with the outcomes of the simulation. The squared magnitudes of numerical error outputs after each integration were added for each lattice site, and verified to be within machine-precision($\approx 10^{-15}$). Similarly, the deviation between the simulation results and the analytical expressions were calculated in each case, and the sum of this deviation was also ensured to lie within machine-precision. The equations used in this regard are as follows:

$$\epsilon_{integrationerror} = \sum_x(|\Delta\Psi_L(x,t)|^2 + |\Delta\Psi_R(x,t)|^2) \tag{22}$$

$$\delta_{simulation-analytical} = \sum_x(|\Psi_L(x,t)-\alpha_{c=0,x}(t)|^2 + |\Psi_R(x,t)-\alpha_{c=1,x}(t)|^2) \tag{23}$$

  where $\Delta\Psi_L(x,t)$, $\Delta\Psi_R(x,t)$ are integration errors from `scipy.integrate.quad()`, and $\alpha_{c,x}$ are components of the state vector found by simulation.

5 Quantum walks in 2D

5.1 Setup and design considerations

In considering quantum random walks in 2 dimensions, we now have one quantum coin tossed for each dimension. In the simplest case, the walker moves in a rectangular lattice, tossing *two* coins at each step. The first coin dictates his direction of movement in the x-axis(say), and the second coin determines his movement along the y-direction.

We see that the coin Hilbert-space ($\mathcal{H}_c$)now has $2 \times 2 = 4$ dimensions, having a basis $\{|j,k\rangle_c, j,k \in \{0,1\}\}$. Similarly, the lattice Hilbert-space($\mathcal{H}_r$) has $N \times N = (2n+1)^2$ dimensions, with a basis $\{|x,y\rangle_r, x,y \in \{-n,-n+1,...n-1,n\}\}$ making the total problem $4(2n+1)^2$-dimensional.

A general state of the system at time 't' can be written as

$$|\Psi(t)\rangle = \sum_{x,y}\sum_{j,k}A_{j,k;x,y}|j,k\rangle_c|x,y\rangle_r \tag{24}$$

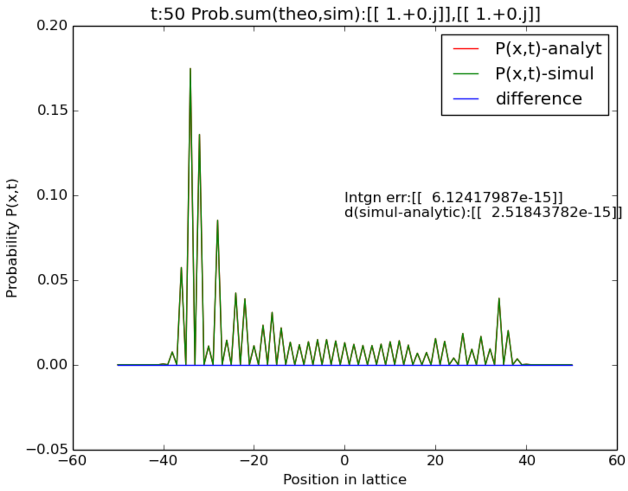

Figure 2: Plot of results of a quantum walk on a 1-D lattice comparing the simulations to the analytical results. Lattice-parameters used are $n = 50$ and $N = 101$ at time-step $t = 50$. The error-parameters described in the analysis section are both indicated in the plot. The initial conditions used are $|0\rangle_c|0\rangle_x$. It is apparent that all the odd lattice sites have zero occupation probability, since the time-step is even.

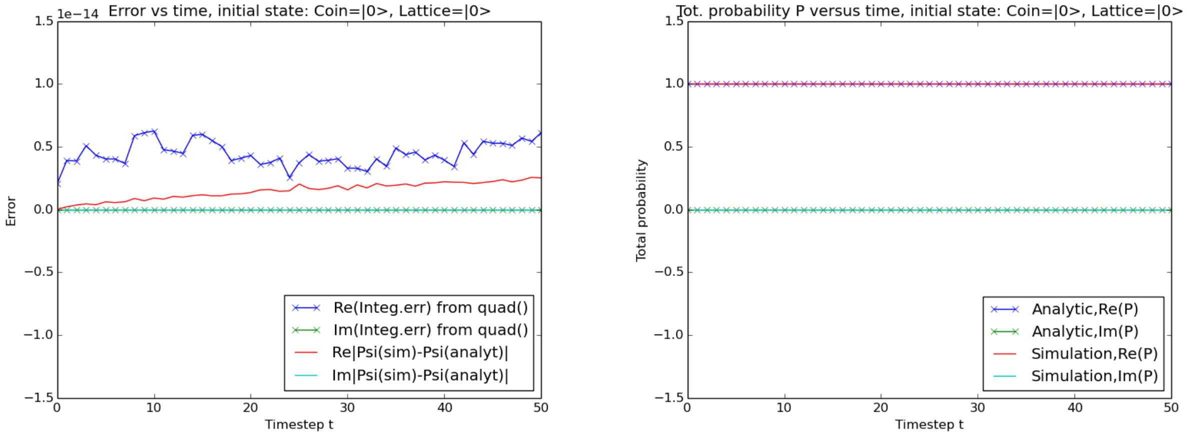

Figure 3: (Left)Plot of numerical error-parameters $\epsilon_{integrationerror}(crossed\ \ line)$ and $\delta_{simulation-analytical}(line)$ versus time, for a 1D quantum walk with $n = 50$ and the initial conditions $|0\rangle_c|0\rangle_x$, showing their bounded nature. (Right) The plot of total probability(see equation 2) versus time for both the analytical expressions(eq 20,21) and the simulation results. The plots of total probability follow identical profiles for all the simulations considered in this program, and are not repeated in this report.

The coin-toss operation, as will be seen, can depend on multiple possible ways of synthesizing a quantum coin in 2D. In general, if we denote the coin operator by $\hat{C}$, we can write its expansion as

$$\hat{C} = \sum_{j,k=0}^{1} \sum_{j',k'=0}^{1} C_{j,k;j',k'} |j,k\rangle_c \langle j',k'| \tag{25}$$

The coin-toss would now be represented by an operator of the form $\hat{T} = \hat{C} \otimes \hat{I}_r$, following which the walker moves according to a *step* operator $\hat{S}$, having the action

$$\hat{S}|j,k\rangle_c |x,y\rangle_r = \hat{S}|j,k\rangle_c |x+(-1)^j, y+(-1)^k\rangle_r \tag{26}$$

on the state $|\Psi(t)\rangle$ of the system.

Since a time-step is characterized by the evolution $|\Psi(t+1)\rangle = \hat{S}\hat{T}|\Psi(t)\rangle$, we can write the state components at a general time-step '$t+1'$ according to the recursion relation

$$A_{j,k,x,y}(t+1) = \sum_{j',k'=0}^{1} C_{j,k;j',k'} A_{j',k';x-(-1)^j, y-(-1)^k}(t) \tag{27}$$

Now, the state evolution of the system seems to depend only on the choice of the form of the coin operator $\hat{C}$. The observable of interest, once again, is the occupation probability of the walker on a lattice site (x,y), which is calculated as

$$P(x,y,t) = \sum_{j,k=0}^{1} |A_{j,k;x,y}(t)|^2 \tag{28}$$

We require completeness to hold in this case as well, which forces a constraint

$$P_{net}(t) = \sum_{x,y=-n}^{n} \sum_{j,k=0}^{1} |A_{j,k;x,y}(t)|^2 = 1 \tag{29}$$

to be satisfied at all time-steps by the system. We consider the classical worst-case scenario in this set of simulations too, and run the simulation only upto a maximum time of $t = n$.

To implement the state evolution of the system, the system's state was stored in a complex numpy array of dimensions $(2, 2, 2n+1, 2n+1)$. The first two indices represented the coin indices, and the final two were the lattice indices. The various coin operators used were stored in complex numpy arrays of dimensions $(2, 2, 2, 2)$, from which the equations 27, 28 were directly computed. This change of approach from the 1D case was beneficial, since defining tensor products appeared to be computationally costly. The initial conditions depended on the choice of coin operator used, and would be discussed in the subsequent section. The constraint equation 29 was ensured to hold in all time-steps.

5.2 Output and analysis of results

- The most obvious choice for the 2D coin operator $\hat{C}$ turns out to be the direct product of two 1D Hadamard coins, which leads to a 2D Hadamard coin of the form $\hat{C}_{H,2d} = \hat{C}_{H,1d} \otimes \hat{C}_{H,1d}$, giving

$$\hat{C}_{H,2d} = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \tag{30}$$

(In the matrix given, the basis cycles through the states $|0,0\rangle_c$, $|0,1\rangle_c$, $|1,0\rangle_c$, $|1,1\rangle_c$ in that order.)

- The Hadamard matrix described above, since it is separable into two 1D Hadamard coins, leads to a system which has the two lattice-dimensions completely unentangled to each other. This would mean that the output of the quantum walk on the lattice would be a superposition of two uncorrelated 1D random walks along the two directions.

- This result is borne out fairly well when we consider the asymmetric initial states $|\Phi(t = 0)\rangle = |0,0\rangle_c|0,0\rangle_r$, and $|1,1\rangle_c|0,0\rangle_r$ for the system, for which the outputs are plotted in figure 4. The probability distribution along the periphery of the walk along each axis is similar to the profile of a 1D random walk in that direction.

  The probability distribution is peaked at the coordinates $(x = \pm 26, y = \mp 26)$, well within the boundary of the lattice used at $n = 40$. This is in keeping with the slow-spreading behavior of the 1D Hadamard walk seen before.

- For the 1D quantum walk, it has been shown[5] that the full range of possible unbiased evolutions of the walk can be generated by only using the 1D Hadamard coin($\hat{C}_{H,1d}$), and by choosing different coin initial states. However, in the 2D walk, the full range of possibilities is determined by the SU(4) group nature of the unitary coin operator. Choosing only unbiased coins, and restricting the leading diagonal entry to be $+1/2$ would lead to a set of 640 possible unitary operators, which contain a lot of subsets whose the final outcomes are identical upto a rotation or a reflection. Upon restricting the set further this way, the coin operators in 2D just fall into 10 types, with the 2D Hadamard operator being just one of them. Two other operators having a different type are the Grover coin and the Fourier coin[5].

- We first look at the outcome of evolution of a symmetric initial state under the Hadamard coin, where

$$|\Psi(t = 0)_H\rangle = \frac{1}{2}(|0\rangle_{cx} + i|1\rangle_{cx}) \otimes (|0\rangle_{cy} + i|1\rangle_{cy}) \otimes |0,0\rangle_r. \tag{31}$$

  As expected, the outcome has four peaks, corresponding to the two independent, symmetric 1D-Hadamard walks along the two coordinates. The peaks are observed at $(|x|, |y|) = (26, 26)$, like in the asymmetric case. The overall distribution has rectangular symmetry, since the 2D Hadamard coin, being factorable, does not mix the two lattice directions in any way.

- In addition to the Hadamard coin, two other possible coins of interest are the Fourier coin and the Grover coin. The Fourier coin derives its name from the use of Discrete Fourier Transform in its definition, and leads to a coin operator having the following form:

$$\hat{C}_{F,2d} = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{bmatrix} \tag{32}$$

- Although the Fourier coin can lead to a lot of interesting behavior as far as the evolution of the system goes, we are most interested in the maximally symmetric evolution of the system, which uses the initial state given below[5]:

$$|\Psi(t = 0)_F\rangle = \frac{1}{2}\Big(|0,0\rangle_c + \frac{1-i}{2}|0,1\rangle_c + |1,0\rangle_c - \frac{1-i}{2}|1,1\rangle_c\Big) \otimes |0,0\rangle_r. \tag{33}$$

- As seen in figure 6, the probability distribution of the Fourier walk has a rotation symmetry by 180^o on the lattice. The walker seems to move faster along one of the two coordinates, and a tad slower along the other. These properties are maintained throughout the evolution of the system at each time-step, and most importantly, the walker *hits* the boundary of the lattice along the faster coordinate!

- This speedup compared to the Hadamard walk(which in itself is faster than the classical random walk) makes the Fourier walk interesting in the context of a quantum search algorithm along a 2D lattice, reaching lattice position at $\mathbb{O}(t)$ along the faster axis.

- Finally, we consider the Grover walk, using the Grover coin $\hat{C}_{G,2d}$ having the structure

$$\hat{C}_{G,2d} = \frac{1}{2}\begin{bmatrix} 1- & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix} \tag{34}$$

- The maximally symmetric evolution under a Grover coin is made possible by the initial state

$$|\Psi(t=0)_G\rangle = \frac{1}{2}\Big(|0,0\rangle_c - |0,1\rangle_c - |1,0\rangle_c + |1,1\rangle_c\Big) \otimes |0,0\rangle_r. \tag{35}$$

- The Grover walk extends the speedup due to the Fourier walk further, moving to lattice positions at $\mathbb{O}(t)$ along both the axes, while maintaining a great deal of symmetry throughout the evolution.

- In fact, the Grover walk has a rotation symmetry about 90^o on the lattice(see figure 7), which is maintained throughout the walk. The final distribution is almost circular in profile, having a near zero probability close to the origin.

- As an extension, we also investigated the time-evolution of the Grover walk under the Hadamard-symmetric initial state given by equation 31. As per the discussion in [5], this leads to a distribution sharply peaked about the origin, with probability $P(x=0, y=0) > 0.3$ even at $t = 40$, the final time-step. The obtained results for this simulation are plotted in figure 8.

- Thus, the Grover walk is seen to be very versatile. By appropriate choices of initial states, the Grover walk can generate both the fastest and slowest spreading times on the lattice, as shown by [5]. More details about controlling the spreading behavior of the Grover 2D quantum walk in the context of Grover's search algorithm can be found in [6].

6 Summary and Conclusions

- The quantum analog of the classical random walk spreads much faster along the lattice than the $\mathbb{O}(\sqrt{t})$ classical limit, and can be systematically studied for the 1D case using an extension of the Discrete-time Fourier Transform(DTFT).

- The chief observable in case of the quantum walk is no longer the position of the walker, but the probability of finding her on a given lattice site.

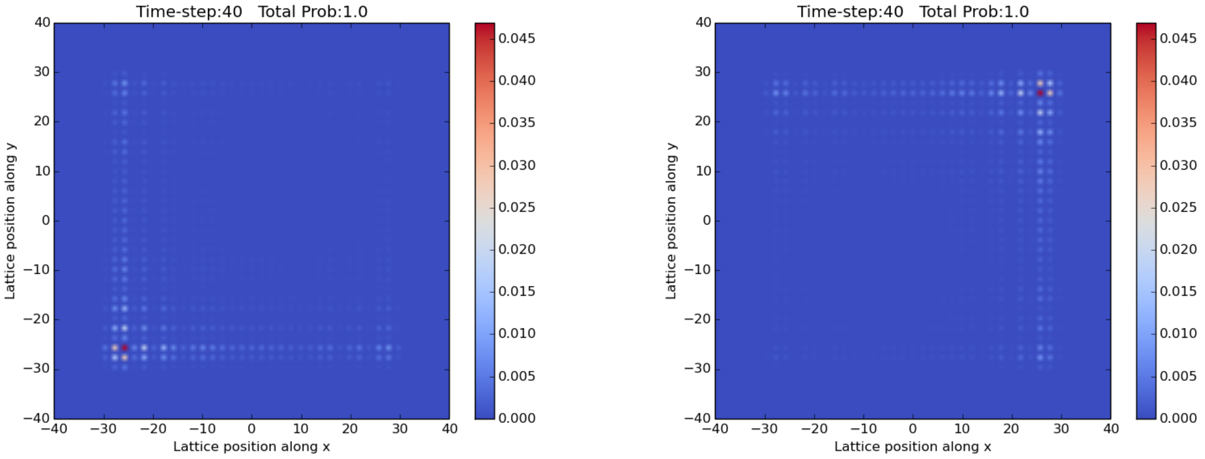

Figure 4: Probability distribution of a walker performing a 2D quantum Hadamard walk on a lattice of $n = 40$ at $t = 40$, for the asymmetric initial conditions $|0,0\rangle_c|0,0\rangle_r(left)$ and $|1,1\rangle_c|0,0\rangle_r(right)$. The first case has the coins initially in the left/down state, the second case has the coin in the right/up state. The distribution is just a superposition of two uncorrelated 1D asymmetric walks along the two coordinates, due to the nature of the coin used.

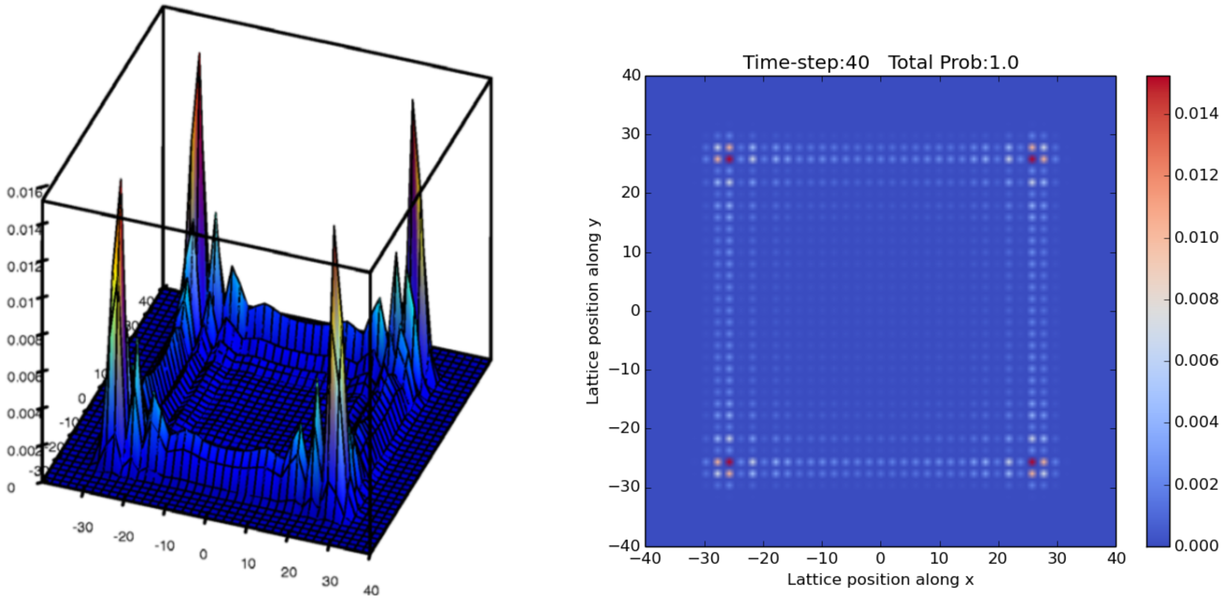

Figure 5: Probability distribution of a walker performing a 2D quantum Hadamard walk on a lattice of $n = 40$ at $t = 40$, for symmetric initial conditions given in equation 31. The first figure(left) is reproduced from Tregenna et al.[5], and the second figure is generated by the Python code.

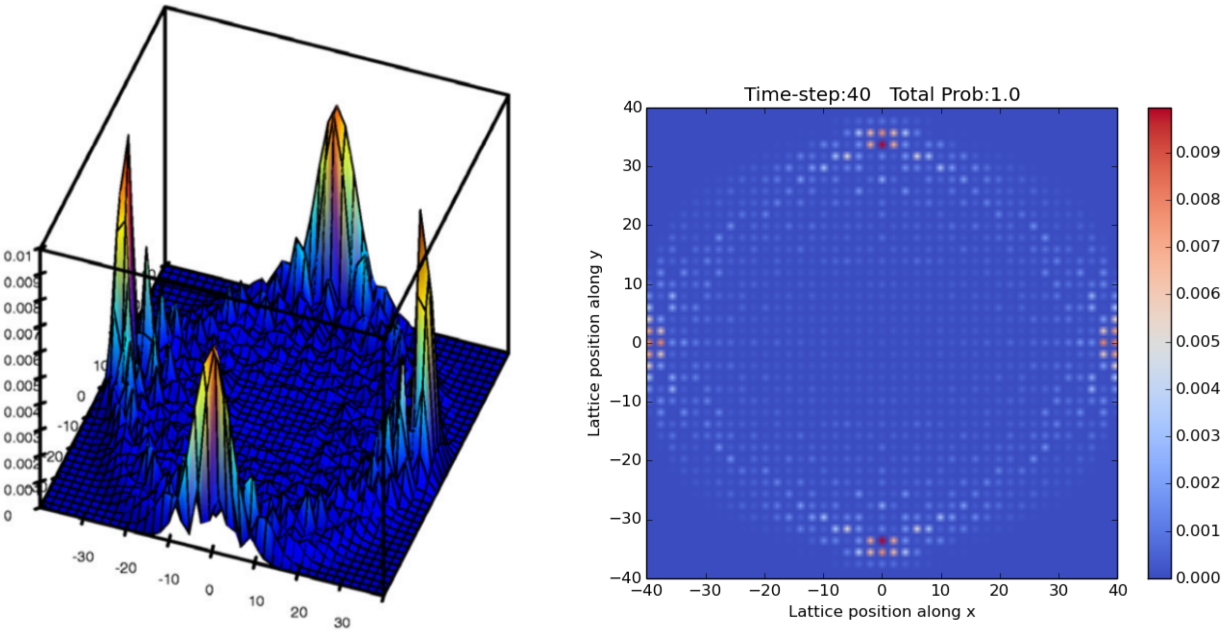

Figure 6: Probability distribution of a walker performing a 2D quantum Fourier walk on a lattice of $n = 40$ at $t = 40$, for symmetric initial conditions given in equation 33. The first figure(left) is reproduced from Tregenna et al.[5], and the second figure is generated by the Python code. The walker seems to 'hit' the lattice boundary along the x-axis, and the distribution is symmetric with respect to a rotation of 180^o about the center.

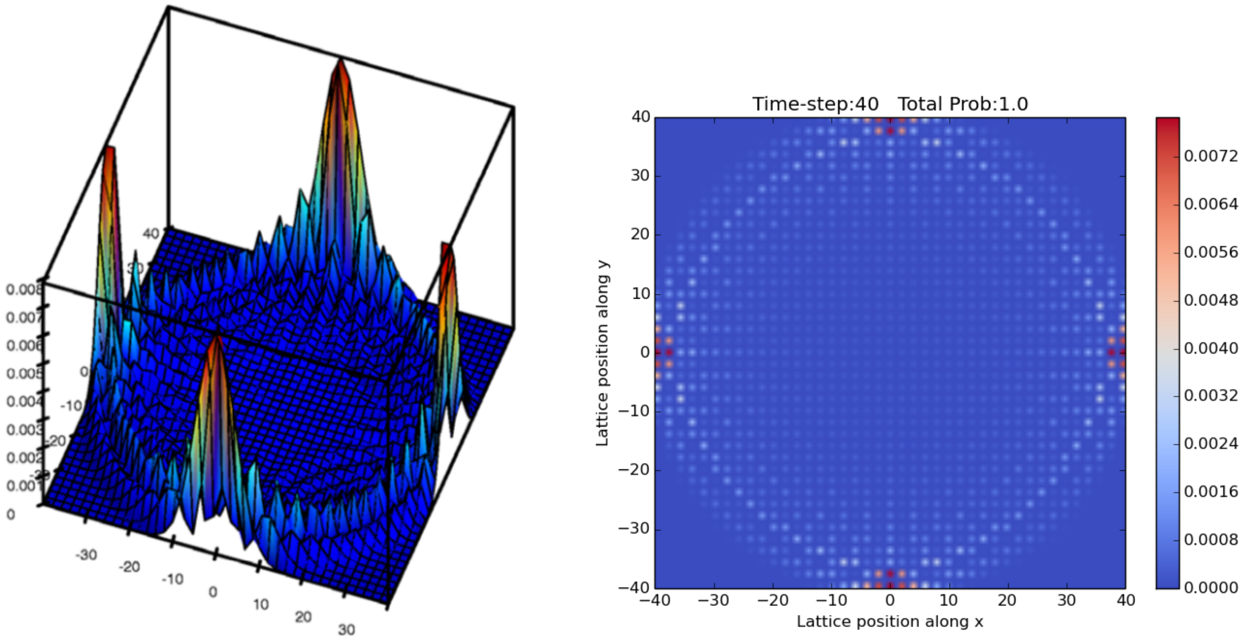

Figure 7: Probability distribution of a walker performing a 2D quantum Grover walk on a lattice of $n = 40$ at $t = 40$, for symmetric initial conditions given in equation 35. The first figure(left) is reproduced from Tregenna et al.[5], and the second figure is generated by the Python code. The walker seems to 'hit' the lattice boundary along both the axes, and the distribution is symmetric with respect to a rotation of 90^o about the center.

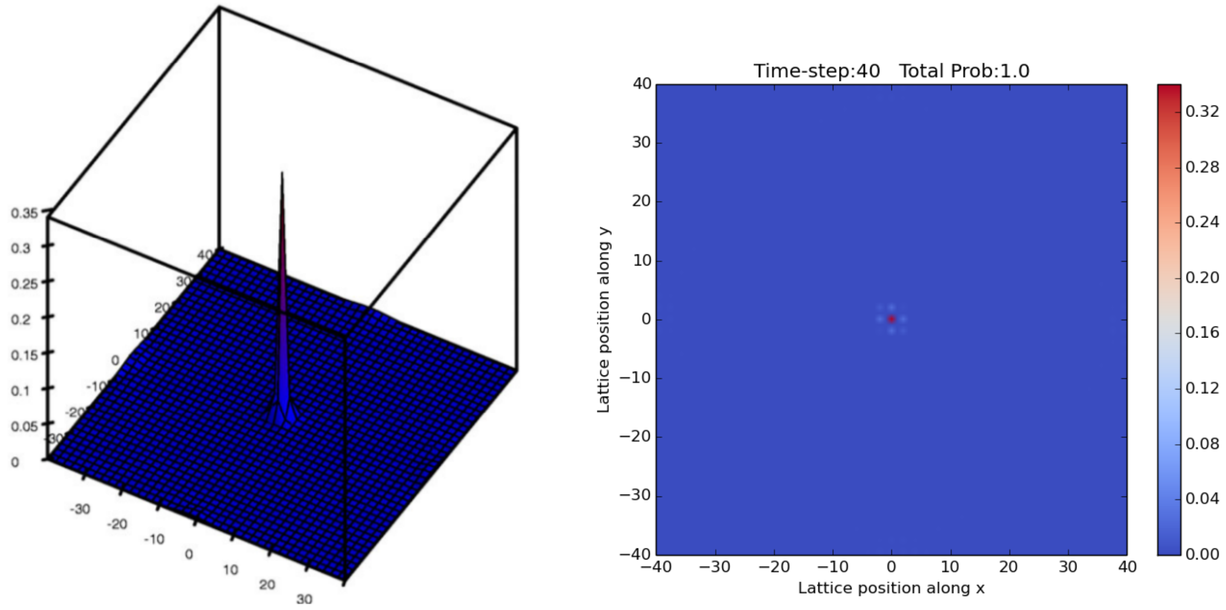

Figure 8: Probability distribution of a walker performing a 2D quantum Grover walk on a lattice of $n = 40$ at $t = 40$, for Hadamard-symmetric initial conditions given in equation 31. The first figure(left) is reproduced from Tregenna et al.[5], and the second figure is generated by the Python code. There is a finite probability that the walker moves to the edge of the lattice, but there is a high probability > 0.3 she stays at the center $(x = 0, y = 0)$.

- A single coin operator(the 1D Hadamard coin) can generate the full range of possible evolutions in the 1D walk. We considered the extreme cases of fully asymmetric and fully symmetric initial conditions for the coin, which lead to probability distributions peaked along one direction, or symmetrically peaked about the origin, respectively. Completeness of the basis-states used is used as a check on numerical stability of the simulations.

- The use of DTFT leads to a closed analytical expression for the occupation probabilities along the 1D-lattice for a quantum walker, which was tested and verified to describe the simulation outcomes reliably for a particular asymmetric initial state.

- In case of the quantum walk on a 2D lattice, we considered three different coins which demonstrate the rich structure of possible unbiased walks possible in the lattice. Of a total of 640 possible unbiased coin operators that can be reduced to 10 categories, we explored the time evolution for three types - the 2D Hadamard coin, the 2D Fourier coin, and the 2D Grover coin.

- The 2D Hadamard coin can be factorized into two 1D Hadamard coins, and hence leads to a probability distribution which resembles a superposition of two separate 1D quantum walks along the two coordinates. This walk was studied for fully asymmetric and fully symmetric initial conditions.

- Even when using the most symmetric initial state, the 2D Hadamard coin does not quite reach the periphery of the lattice - despite its speedup compared to the classical random walk. However, using a Fourier or a Grover coin along with appropriately chosen initial states is seen to speed up the quantum walk even further.

- The most symmetric Fourier walk is seen to move along one of the two coordinates at $\mathbb{O}(t)$

and a tad slower along the other coordinate, while the symmetric Grover walk is seen to move at $\mathbb{O}(t)$ along both the coordinates. This speedup can have potential applications in quantum search algorithms, which perform much faster than their classical counterparts which use Brownian motion.

- In addition, the Grover walk is seen to be very versatile. By an appropriate choice of initial states, the same Grover coin can cause both maximal and minimal spreading on a 2D lattice, which can have potential applications in quantum information processing.

References

[1] 'Quantum random walks - an introductory overview', Julia Kempe, Contemporary Physics, Vol. 44(4), pp.307-327(2003).

[2] 'Quantum walks: a comprehensive review', Salvador E. Venegas-Andraca, Quantum Information Processing vol. 11(5), pp.1015-1106(2012).

[3] 'Quantum Walk on the Line', A.Nayak and A.Vishwanath, https://arxiv.org/abs/quant-ph/0010117

[4] 'One-dimensional Quantum Walks', A.Ambainis et al, Proceedings of the Thirty-Third Annual ACM Symposium on the Theory of Computing, pages 37–49, 2001.

[5] 'Controlling discrete quantum walks: coins and initial states', B.Tregenna et al., New Journal of Physics 5 (2003) 83.1–83.19

[6] 'Localization of two-dimensional quantum walks', N. Inui et al., Phys. Rev. A 69, 052323(2004).