A Mini-Introduction To (Quantum) Random Walks

Bhavishya

July 2018

Abstract

This article is a short introduction to (Quantum) Random Walks which is an emerging paradigm for designing new "Quantum" algorithms. We start with Classical Random walks, then Quantum Random Walks for both continuous and discrete cases are described and eventually some applications, in designing algorithms, are discussed.[1]

Table of Contents

Introduction

Randomized algorithm have been very successful in history of theoritical computer science due to their simplicity and speed [2] (Eg: primality-testing, min-cut etc.). Many known intractable problems only have randomized algorithms like Approximate-Counting[2]. The basic complexity class for Randomized algorithms is RP (the class of decision problems with success probability at least 1/2). The class of problems with success probability more than 1/2 is called BPP. Note that this probability can be arbitrarily increased by multiple runs and then taking the median. The quantum equivalent of BPP is the class BQP, [3] showed that BPP $\subseteq$ BQP and it is believed that BPP $\neq$ BQP. Our believe that Quantum

[1] This article follows the style of a recent article [1] by Ed Witten.

[2] Approximate Counting belongs #P the class associated with set of counting problems in NP and thus even harder than NP

algorithms are better than classical ones, justifies the study of Quantum Random Walks. Infact [4] proved an oracle separation between classical and quantum case for a graph traversal problem.

Classical Random Walks

Preliminaries and Definitions

Markov Chain It is a discrete time stochastic process defined by a matrix P of transition probabilities over states S.

A discrete-time Markov chain is a sequence of random variables X_1, X_2,... X_n, such that

$\Pr(X_{n+1} = x \mid X_1 = x_1, ..., X_n = x_n) = \Pr(X_{n+1} = x | X_n = x_n)$ Now $P_{ij} = Pr[X_t = j | X_i]$, Thus a **Markov Chain is memoryless.**

Stationary Distribution A π such that $\pi = \pi P$ for transition matrix P.

Irreducibility of Markov Chain Any state(j) is accesible from any other state(i) i.e. $Pr(X_t = j | X_1 = i) > 0$.

Aperiodicity of Markov Chain For all states i and j, if gcd$\{Pr(X_n = j | X_0 = i)\} = 1$ then it is aperiodic.

Ergodicity A markov chain is ergodic if it's both Irreducible and Aperiodic.

Fundamental theorem of Markov Chains An ergodic markov chain has a stationary distribution i.e. π such that $P\pi = \pi$.

Examples of Classical Random walks:

1. **Walk on a infinite Line**

Suppose the probability of moving right is p and left q. You start from 0, you might want to find the position after k steps, or at least the mean(and variance). Let's use analysis similar to [5], We define Z_n to be the state after n steps, Now number of "right" steps can be written as $(Z_n + n)/2$, but moving "right"(or "left") is a binary variable with probability p(or q). E["right"] = np, V["right"] = npq, putting back in previous eq. you can find E$[Z_n]$ = n(p-q) and V$[Z_n]$ = $4npq$. Also note variance increases **linearly** in n.

2. **Walk on Graphs : S-T connectivity**

We start with defining some terms common in random walks on graphs, For $G = (V, E)$ we can define a markov chain, M_G with

$P_{uv} = 1/\text{d(u)}$ if (u,v) ϵ E and 0 otherwise.

For all $v \epsilon V$, $\pi_v = d(v)/2m$.

Hitting Time Expected number of steps for a random walk to start at u and end upon first reaching v.

Mixing Time It is the smallest time when $P_u(t)$ is within ϵ distance of π

Now we move to the problem: Given G, determine whether s and t are connected or not. The random walk algorithm is particularly effective when we have limited space(logarithmic).

S-T connectivity Algorithm

```
Start at y = s
repeat this T=|V|^3 times,
Choose any neighbour x of y at random
If x is t
    Stop and Return True
Else
    y = x
Return False
```

This algorithm has greater than 1/2 probability of correct answer. And we can decrease the failure probability to ϵ by repeating the algorithm about $\log(1/\epsilon)$ and taking the median.

3. **2-SAT** A very similar algorithm to above, is for solving the 2-SAT problem for n variables.

2-SAT Algorithm

```
Start with some assignment T
repeat 2n^2 times
    If no UNSAT clause
        Stop and Return SAT
    Else
        Flip literals and update T
Stop and Return UNSAT
```

Quantum Random Walks

A basic hypothesis of Quantum Mechanics is Unitarity. But we just saw that the classical random walk has a tendency to lose memory so we do expect that Quantum Walks would be different in some ways.

Quantum Discrete Walk

To define Quantum Walks we need to define "coin" state that determines "shift". The usual choice for "coin" is the Hadamard operator.

So we can define C(coin flip operator)

```
C|n,0> = a1|n,0> + a2|n,1>
C|n,1> = a3|n,0> + a4|n,1>
```

For usual when using Hadamard operator as coin flip we can keep a1, a2, a3 to be equal to $1/\sqrt{2}$ and a4=-$1/\sqrt{2}$.

Also define S(shift operator) as

```
S|n,0> = |n+1,0>
S|n,1> = |n-1,1>
```

Now t^{th} step of random walk is given by applying $(SC)^t$. With Classical walk we obtained a normal distribution, but with quantum walks things are pecularily different. Observe the negative sign in a4,(try applying the above operator 4-5 times) this leads to cancellation in amplitudes and the walks start to shift towards Right. Another aspect is that the expected distance from origin after t steps is $\Omega(t)$(compared to $\sqrt{t}$ in classical case).

Quantum Continuous Walk

For the continuous case we do not require a coin, we can define them in a very natural way as follows, by taking inspiration from Adjacency matrix of a graph G. We can dee fine the **Laplacian** of G defined as

$L_{j,k} = -deg(j)$ when $j = k$ else equal to 1 if $(j,k)\ \epsilon\ E$ and 0 otherwise.

Now we can write the following equation,

$$\frac{d}{dt}p_j(t) = \Sigma_{k\epsilon V} L_{jk} p_k(t) = 0$$

which has the solution,

$$p(t) = e^{Lt}p(0)$$

which is very similar to Schrödenger equation

$$\mathrm{i}\frac{d}{dt}|\Psi> = H|\Psi>$$

but could be made exactly similar by replacing p_j with complex amplitudes q_j.

So the closed form solution is $q(t) = e^{-iLt}q(0)$.

A more general construction can be done for arbitrary graph [6].

Quantum Algorithms based on Random Walks

As per [6] we see two major groups of algorithms,

1. Exponentially Faster Hitting (for full binary trees)

We saw for the $S-T$ connectivity problem, that a classical algorithm takes exponential time. But [7] showed that for special case of binary tree one root can hit other root with constant probability in polynomial time($O(d^2)$ where d is the depth).

The proof uses the idea that the binary tree could be reduced to a small subspace(of $2d+1$ dimensions) and problem can be treated as continous walk on straight line.

2. Quantum Walk Search(Element Distinctness)

Given numbers $a_1, ..., a_n$ such that there exist i, j such that $i \neq j$ but $a_i = a_j$.

Classically we require $\Omega(n)$ queries. [8] solved this problem by providing an $O(n^{2/3})$ algorithm which is also proven to be optimal.

Start from a vertex, check if it's marked, if not query just one neighbour, now we can define a quantum walk on remaining vertices which finds the appropriate vertex in $O(d^{2/3})$.

References

[1] Witten E. 2018. A mini-introduction to information theory. arXiv preprint arXiv:1805.11965.

[2] Motwani R, Raghavan P. 1995. Randomized algorithms. New York, NY, USA: Cambridge University Press.

[3] Bernstein E, Vazirani U. 1997. Quantum complexity theory. SIAM Journal on Computing 26(5):1411–1473.

[4] Childs AM, Cleve R, Deotto E, Farhi E, Gutmann S, Spielman DA. 2003. Exponential algorithmic speedup by a quantum walk. In: Proceedings of the thirty-fifth annual acm symposium on theory of computing. ACM, pp 59–68.

[5] Venegas-Andraca SE. 2008. Quantum walks for computer scientists. Morgan; Claypool Publishers.

[6] Ambainis A. 2003. Quantum walks and their algorithmic applications. International Journal of Quantum Information 1(04):507–518.

[7] Childs AM, Farhi E, Gutmann S. 2002. An example of the difference between quantum and classical random walks. Quantum Information Processing 1(1-2):35–43.

[8] Ambainis A. 2007. Quantum walk algorithm for element distinctness. SIAM Journal on Computing 37(1):210–239.