Quantum Search Algorithm and Discrete Schrödinger Dynamics

Jiaqi Leng

Department of Mathematics, University of Maryland, College Park
Institute for Advanced Computer Studies (UMIACS)
Joint Center for Quantum Information and Computer Science (QuICS)

March 6, 2020

Overview

1. Grover's quantum search algorithm
2. Discrete Schrödinger Dynamics on graphs
3. Grover's algorithm as discrete Schrödinger dynamics
4. Appendix: Calculating generators of W and U_f

Search algorithm

Search Problem. There is a connected finite graph $G = (V, E)$. Among all vertices in V, only one vertex v^* is marked. How to find v^*?
We are allowed to query a function (black box) $f : V \to \{0, 1\}$ such that $f(v^*) = 1$ and $f(v) = 0$ if $v \neq v^*$.

Search algorithm

Search Problem. There is a connected finite graph $G = (V, E)$. Among all vertices in V, only one vertex v^* is marked. How to find v^*?
We are allowed to query a function (black box) $f : V \to \{0, 1\}$ such that $f(v^*) = 1$ and $f(v) = 0$ if $v \neq v^*$.
Query complexity: The strategy is to query f on different vertices until we get 1. Each time we can only check 1 vertex. In the worst case, we have to query the black box $n - 1$ times before get v^*.
$\implies$ query complexity $= O(n)$.

Basics of quantum computing

Dirac Notation. In quantum mechanics, a column vector (quantum state) is written as a 'ket',e.g., $|\psi\rangle$. Its conjugate (a row vector) is a 'bra', e.g., $\langle\psi|$. The inner product of $|\psi\rangle$ and $|\phi\rangle$ is a 'braket': $\langle\psi|\phi\rangle$.

Classical Computer	Quantum Computer
bit	qubit (a **unit** vector in a 2-dim Hilbert space)
gate	quantum gate (a unitary matrix)
tape reader	measurement (an orthonormal basis)

Basics of quantum computing

Dirac Notation. In quantum mechanics, a column vector (quantum state) is written as a 'ket',e.g., $|\psi\rangle$. Its conjugate (a row vector) is a 'bra', e.g., $\langle\psi|$. The inner product of $|\psi\rangle$ and $|\phi\rangle$ is a 'braket': $\langle\psi|\phi\rangle$.

Classical Computer	Quantum Computer
bit	qubit (a **unit** vector in a 2-dim Hilbert space)
gate	quantum gate (a unitary matrix)
tape reader	measurement (an orthonormal basis)

Measurement of a quantum system: we can use an orthonormal basis $\{|0\rangle, ..., |n-1\rangle\}$ to 'measure' a quantum system, and any element in this basis can be an output.
By **Born rule**, the probability of getting result $|j\rangle$ is $|\langle\Psi|j\rangle|^2$. Since $\|\Psi\|_2 = 1$, all probabilities adds up to 1.

More on Dirac notation

Dirac notation can be used to express matrix.

Example

Let $|e_0\rangle = \frac{1}{\sqrt{n}} \begin{pmatrix} 1 \\ 1 \\ .. \\ 1 \end{pmatrix} = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} |j\rangle$. What is $|e_0\rangle \langle e_0|$?

$$|e_0\rangle \langle e_0| = \frac{1}{\sqrt{n}} \begin{pmatrix} 1 \\ 1 \\ .. \\ 1 \end{pmatrix} \frac{1}{\sqrt{n}} (1, 1, ..., 1) = \frac{1}{n} \begin{bmatrix} 1 & 1 & ... & 1 \\ 1 & 1 & ... & 1 \\ ... & ... & ... & ... \\ 1 & 1 & ... & 1 \end{bmatrix} := \frac{1}{n} J_n,$$

where J_n is a n-by-n all-ones matrix.

Back to the search problem

On a quantum computer, we can use orthonormal basis $|j\rangle$ $(j = 0, 1, ..., n-1)$ to enumerate vertices. This basis is called *computational basis*.
The function f can be implemented as a *quantum oracle* U_f (a unitary matrix).

$$U_f |j\rangle = (-1)^{f(j)} |j\rangle = \begin{cases} -|j\rangle & (j \text{ is marked}) \\ |j\rangle & (j \text{ is not marked}) \end{cases}$$

The most crazy thing of quantum computer is that it allows linear combinations of orthonormal basis! This means, we can have a 'superposition' of all vertices at a time!

Grover's algorithm

- Prepare a state $|e_0\rangle = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} |j\rangle$. It is a uniform superposition of all possible 'vertices'.
- Apply the quantum oracle U_f.
- Apply gate (unitary matrix) $W := 2|e_0\rangle\langle e_0| - \mathbb{I}_n$. ($\mathbb{I}_n$ is identity matrix.)
- Repeat step 2 and 3 for K times, where K is the closest integer to $\frac{\pi}{4}\sqrt{n} - \frac{1}{2}$.
- Measure the resulting state in computational basis. With probability at least $1 - \frac{1}{n}$, the measurement outcome is v^*.

Query complexity: We use the quantum oracle U_f $K \sim \sqrt{n}$ times.
$\Longrightarrow$ query complexity $= O(\sqrt{n})$. Quantum Speedup

Geometric visualization: amplitude amplification

Decompose the input state into two parts:

$$|e_0\rangle = \frac{1}{\sqrt{n}}|v^*\rangle + \frac{1}{\sqrt{n}}\sum_{j\neq v^*}|j\rangle = \sin\theta\,|v^*\rangle + \cos\theta\,|\phi\rangle\,,$$

where $\sin\theta = \frac{1}{\sqrt{n}}$, $|\phi\rangle = \frac{1}{\sqrt{n-1}}\sum_{j\neq v^*}|j\rangle$ is a unit vector.
We can prove $WU_f|e_0\rangle = \sin 3\theta\,|v^*\rangle + \cos 3\theta\,|\phi\rangle$, and inductively, $(WU_f)^k|e_0\rangle = \sin(2k+1)\theta\,|v^*\rangle + \cos(2k+1)\theta\,|\phi\rangle$. By Born rule, the probability we get $|v^*\rangle$ is

$$p = |\sin(2k+1)\theta|^2.$$

When $(2k+1)\theta = \frac{\pi}{2}$, the probability of getting v^* is 1. At this time, $k \approx \frac{\pi}{4}\sqrt{n} - \frac{1}{2}$.

Overview

1. Grover's quantum search algorithm
2. Discrete Schrödinger Dynamics on graphs
3. Grover's algorithm as discrete Schrödinger dynamics
4. Appendix: Calculating generators of W and U_f

Schrödinger equation

The dynamics of quantum systems are described by Schrödinger equation:

$$i\hbar \frac{\mathrm{d}}{\mathrm{d}t} |\Psi(t)\rangle = \hat{H} |\Psi(t)\rangle ,$$

where $\hbar$ is the Planck constant, $|\Psi(t)\rangle : \mathbb{R}^3 \times [0, \infty) \to \mathbb{C}$ is the *wave function*, $\hat{H}$ is the *Hamiltonian*.

Schrödinger equation

The dynamics of quantum systems are described by Schrödinger equation:

$$i\hbar \frac{\mathrm{d}}{\mathrm{d}t} |\Psi(t)\rangle = \hat{H} |\Psi(t)\rangle ,$$

where $\hbar$ is the Planck constant, $|\Psi(t)\rangle : \mathbb{R}^3 \times [0, \infty) \to \mathbb{C}$ is the *wave function*, $\hat{H}$ is the *Hamiltonian*.
For a single non-relativistic particle, the Hamiltonian is a linear partial differential operator

$$\hat{H} = -\frac{\hbar^2}{2m}\nabla^2 + V(x, t).$$

Copenhagen interpretation of $|\Psi(t)\rangle$: $|\Psi(t)|^2$ is the probability density of the particle in $\mathbb{R}^3$ at time t.

Wave functions on graph

Let $G = (V, E)$ be a connected finite graph, $V = \{v_1, v_2, ..., v_n\}$.
We can cook up a definition of wave functions on graph G analogous to its counterpart on Euclidean space.

$$|\Psi\rangle : V \to \mathbb{C} \text{ s.t. } |\Psi|^2 = 1.$$

In other words, $|\Psi\rangle$ is a $n-$dimensional complex-valued unit vector in the Hilbert space $H_G = \mathbb{C}^n$.

Wave functions on graph

Let $G = (V, E)$ be a connected finite graph, $V = \{v_1, v_2, ..., v_n\}$.
We can cook up a definition of wave functions on graph G analogous to its counterpart on Euclidean space.

$$|\Psi\rangle : V \to \mathbb{C} \text{ s.t. } |\Psi|^2 = 1.$$

In other words, $|\Psi\rangle$ is a $n-$dimensional complex-valued unit vector in the Hilbert space $H_G = \mathbb{C}^n$.

$$|\Psi\rangle = (\Psi_1, ..., \Psi_n)^T \in H_G.$$

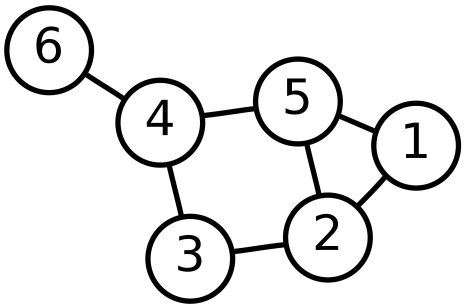
Copenhagen interpretation of $|\Psi\rangle$: $p = (|\Psi_1|^2, |\Psi_2|^2, ..., |\Psi_n|^2)$ is the probability mass function of the 'particle' on the graph G.

Graph Laplacian

The Laplacian operator on the (undirected) graph $G = (V, E)$ is a $n-$by$-n$ symmetric matrix

$$L_{ij} = \begin{cases} -deg(v_i) & (i = j) \\ 1 & ((i,j) \in E) \\ 0 & elsewhere \end{cases}$$

For example, the Laplacian of the graph on the left is



$$L = \begin{bmatrix} -2 & 1 & & & 1 & \\ 1 & -3 & 1 & & 1 & \\ & 1 & -2 & 1 & & \\ & & 1 & -3 & 1 & 1 \\ 1 & 1 & & 1 & -3 & \\ & & & & 1 & -1 \end{bmatrix}$$

Laplacian of N-complete graph

A graph with N vertices is called a *N-complete* graph if any vertex is connected to all the other vertices.

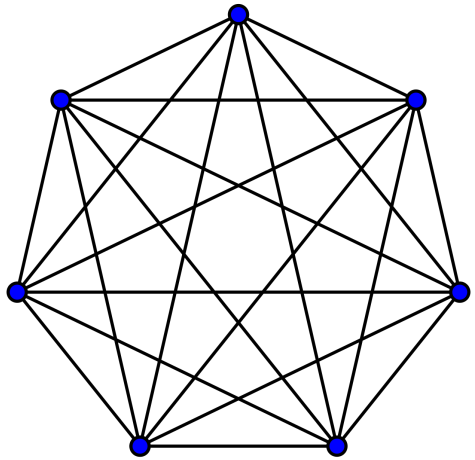


Figure: A 7-complete graph

Laplacian of $N-$complete graph is of a neat form:

$$L = J_N - N\mathbb{I}_N,$$

where J_N is a $N-$by$-N$ all-ones matrix, and $\mathbb{I}_N$ is the $N-$by$-N$ identity matrix.

Schrödinger equation on graph

Suppose the particle is of unit mass $m = 1$, and the Planck constant is normalized as 1, we can formulate the Schrödinger equation on a graph $G = (V, E)$ as

$$i\frac{\mathrm{d}}{\mathrm{d}t}\left|\Psi(t)\right\rangle = \hat{H}\left|\Psi(t)\right\rangle,$$

where $\left|\Psi(t)\right\rangle : [0, T] \to H_G = \mathbb{C}^n$, $\hat{H} = -L + V$.

Schrödinger equation on graph

Suppose the particle is of unit mass $m = 1$, and the Planck constant is normalized as 1, we can formulate the Schrödinger equation on a graph $G = (V, E)$ as

$$i\frac{\mathrm{d}}{\mathrm{d}t}\left|\Psi(t)\right\rangle = \hat{H}\left|\Psi(t)\right\rangle,$$

where $|\Psi(t)\rangle : [0, T] \to H_G = \mathbb{C}^n$, $\hat{H} = -L + V$.
V is the *potential operator*. It is a $n-$by$-n$ diagonal matrix:

$$V = Diag(f(v_1), f(v_2), ..., f(v_n)),$$

where $f(v)$ is the potential function on the graph

$$f : V \to \mathbb{R}.$$

Solutions to discrete Schrödinger equation, time evolution operator

We will call Schrödinger equation on graph as *discrete Schrödinger equation:*

$$i\frac{\mathrm{d}}{\mathrm{d}t}\left|\Psi(t)\right\rangle = \hat{H}\left|\Psi(t)\right\rangle, \ \hat{H} \text{ is a Hermitian matrix.}$$

Unlike the Schrödinger equation in $\mathbb{R}^3$ (a linear PDE), Schrödinger equation on graph is a $n-$dimensional system of ODEs.

Solutions to discrete Schrödinger equation, time evolution operator

We will call Schrödinger equation on graph as *discrete Schrödinger equation:*

$$i\frac{\mathrm{d}}{\mathrm{d}t}\left|\Psi(t)\right\rangle = \hat{H}\left|\Psi(t)\right\rangle, \ \hat{H} \text{ is a Hermitian matrix.}$$

Unlike the Schrödinger equation in $\mathbb{R}^3$ (a linear PDE), Schrödinger equation on graph is a $n-$dimensional system of ODEs.
Given an initial condition $\left|\Psi(0)\right\rangle$, the solution can be written as

$$\left|\Psi(t)\right\rangle = e^{-i\hat{H}t}\left|\Psi(0)\right\rangle.$$

In physics, we call $U(t) := e^{-i\hat{H}t}$ the *time evolution operator* of the system, and $\hat{H}$ is the *generator* of $U(t)$.

Overview

Motivating example: Nesterov's accelerated gradient method

$$\min f(x).$$

Fix x_0, set $y_0 = x_0$. In the $k-$th iteration, compute

$$x_k = y_{k-1} - s\nabla f(y_{k-1}), y_k = x_k + \frac{k-1}{k+2}(x_k - x_{k-1}).$$

This method is known to have quadratic convergence rate, which is optimal among all first-order methods.

Motivating example: Nesterov's accelerated gradient method

$$\min f(x).$$

Fix x_0, set $y_0 = x_0$. In the $k-$th iteration, compute

$$x_k = y_{k-1} - s\nabla f(y_{k-1}), y_k = x_k + \frac{k-1}{k+2}(x_k - x_{k-1}).$$

This method is known to have quadratic convergence rate, which is optimal among all first-order methods.
Su, Boyd and Candes (2014) showed this method is a discretization of a second-order initial value problem:

$$\ddot{X} + \frac{3}{t}\dot{X} = -\nabla f(X), X(0) = x_0.$$

They gained better understanding and analysis of Nesterov's method, and as a biproduct, they are able to design a family of schemes with similar convergence rate.

Trotter product formula

In our formulation, $\hat{H} = -L + V$. It is tempted to ask if

$$U(t) := e^{-i(-L+V)t} = e^{-i(-L)t} e^{-iVt}?$$

In general, it is not true. (We can prove $e^{A+B} = e^A e^B$ if and only if $[A, B] := AB - BA = 0$.) But for small time $t = \epsilon << O(1)$, the product is a good approximation:

$$U(\epsilon) \approx e^{iL\epsilon} e^{-iV\epsilon}.$$

Trotter product formula

In our formulation, $\hat{H} = -L + V$. It is tempted to ask if

$$U(t) := e^{-i(-L+V)t} = e^{-i(-L)t} e^{-iVt}?$$

In general, it is not true. (We can prove $e^{A+B} = e^A e^B$ if and only if $[A, B] := AB - BA = 0$.) But for small time $t = \epsilon << O(1)$, the product is a good approximation:

$$U(\epsilon) \approx e^{iL\epsilon} e^{-iV\epsilon}.$$

Theorem (Trotter Product Formula)

$$e^{(A+B)} = \lim_{N\to\infty} \left(e^{\frac{A}{N}} e^{\frac{B}{N}} \right)^N$$

$$U(t) \approx (e^{i\frac{L}{N}t} e^{-i\frac{V}{N}t})^N$$

Revisit Grover's algorithm

In Grover's algorithm, we want to iterate the unitary transformation WU_f K times, i.e., $(WU_f)^K$.
If we can find some large N and write $W = e^{iH_1/N}$, $U_f = e^{iH_2/N}$, then by Trotter formula:

$$(WU_f)^N = (e^{iH_1/N} e^{iH_2/N})^N \approx e^{i(H_1+H_2)}.$$

Revisit Grover's algorithm

In Grover's algorithm, we want to iterate the unitary transformation WU_f K times, i.e., $(WU_f)^K$.
If we can find some large N and write $W = e^{iH_1/N}$, $U_f = e^{iH_2/N}$, then by Trotter formula:

$$(WU_f)^N = (e^{iH_1/N}e^{iH_2/N})^N \approx e^{i(H_1+H_2)}.$$

It turns out that

$$H_1 = \frac{\pi}{n}L, \ L \text{ is the Laplacian of } n-\text{complete graph},$$

$$H_2 = -\frac{\pi}{n}V,$$

V has $-n$ at the v^*-th diagonal entry, with all other entries being 0. It can be interpreted as a potential operator. The potential at the vertex v^* is $-n$, while being 0 at all other vertices.

Grover's algorithm as Schrödinger dynamics

$$WU_f = e^{iH_1}e^{-iH_2} = e^{-i\pi(-L/n)}e^{-i\pi V/n},$$

by Trotter product formula,

$$(WU_f)^n = \left(e^{-i\pi(-L/n)}e^{-i\pi V/n}\right)^n \approx e^{-i\hat{H}\pi} = U(\pi),$$

with $\hat{H} = -L + V$, L, V are defined in previous slides. Recall that $K \approx \frac{\pi}{4}\sqrt{n}$, so

$$(WU_f)^K = [(WU_f)^n]^{K/n} \approx U(K\frac{\pi}{n}) = U\left(\frac{\pi^2}{4\sqrt{n}}\right).$$

Physical interpretation of Grover's algorithm

Grover's algorithm can be thought as a discretization of a discrete Schrödinger dynamics on graph as follows:

- Prepare a 'photon' in state $|e_0\rangle$.
- Place it on a $n-$complete graph with the potential field V.

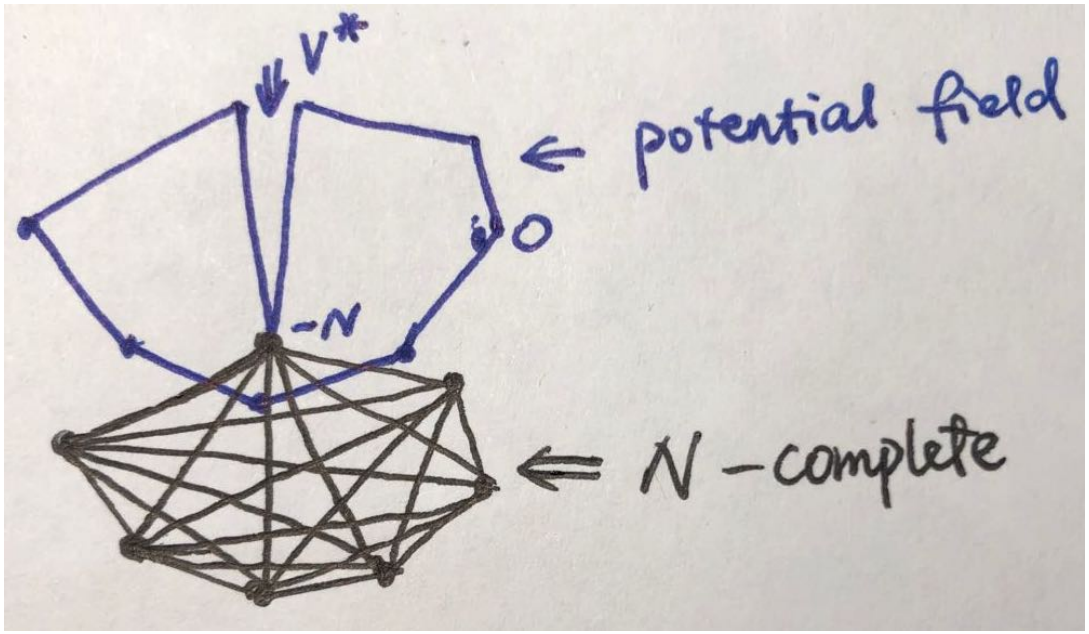


Figure: Physical scenario of Grover's algorithm

Numerical simulation

Set $n = 1000$, and mark vertex # 7. We simulate the Schrödinger equation (on the same L, V as before) on a 1000-complete graph.

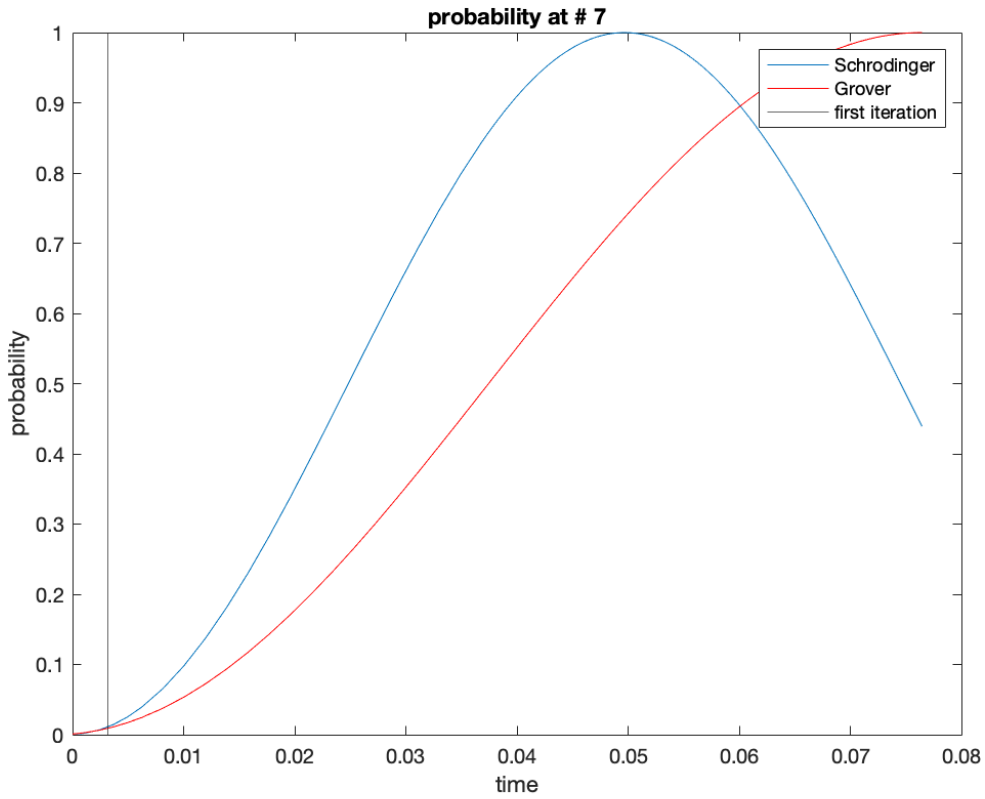


Figure: Comparison between Grover and Schrödinger dynamics

Summary

- We give a 'physical interpretation' of Grover's search algorithm.
- Nielsen and Chuang (they wrote one of the most used references in quantum computation) proposed an alternative approach to Grover's algorithm as a quantum simulation (Section 6.2). Our work is a more detailed exposition of this perspective.
- As what we expect, this 'discrete Schrödinger dynamics' approach **CAN** help us design new quantum algorithms. We are currently working in this direction, and a lot of positive evidence was found.

Overview

W and U_f

Recall that $|e_0\rangle = \frac{1}{\sqrt{n}} \begin{pmatrix} 1 \\ \dots \\ 1 \end{pmatrix}$.

$$W = 2|e_0\rangle\langle e_0| - I = \frac{1}{n} \begin{bmatrix} -(n-2) & 2 & 2 & \dots & 2 \\ 2 & -(n-2) & 2 & \dots & 2 \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & 2 & -(n-2) \end{bmatrix}$$

$$U_f = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & -1 & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ & & & & 1 \end{bmatrix}$$

The generator of W

W is a circulant matrix, and we can readily compute its diagonalization:

$$W = FDiag(-1, -1, ..., -1, 1)F^{-1},$$

where F is the Fourier basis. Suppose $W = e^{iH_1}$, we have

$$H_1 = -i \log W = \frac{\pi}{n} \begin{bmatrix} -(n-1) & 1 & 1 & ... & 1 \\ 1 & -(n-1) & 1 & ... & 1 \\ ... & ... & ... & ... & ... \\ ... & ... & ... & 1 & -(n-1) \end{bmatrix}$$

$$= \frac{\pi}{n} L, \ L \text{ is the Laplacian of } n-\text{complete graph.}$$

The generator of U_f

U_f is a diagonal matrix with entries ± 1, suppose $U_f = e^{iH_2}$, we have

$$H_2 = i \log U_f = iDiag(0, 0, ..., i\pi, 0, .., 0) = \frac{\pi}{n} Diag(0, 0, ..., -n, 0, .., 0).$$

Let $V = Diag(0, 0, ..., -n, 0, .., 0)$, it is a potential operator on the graph G, with the only non-trivial potential being $-n$ at the marked vertex. Now, we have $H_2 = \frac{\pi}{n} V$.