On the probability of finding marked connected subset using quantum walks

Nikolajs Nahimovs[1], Raqueline A. M. Santos[1], Kamil Khadiev[1,2]

[1]Center for Quantum Computer Science, University of Latvia - Riga, Latvia
[2]Kazan Federal University - Kazan, Russia
nikolajs.nahimovs@lu.v rsantos@lu.lv kamilhadi@gmail.com

1. Introduction

Finding a marked vertex in a graph can be a complicated task when using quantum walks. Recent results show that for two or more adjacent marked vertices search by quantum walk with Grover's coin may have no speed-up over classical exhaustive search.
In this work, we analyze the probability of finding a marked vertex for a set of connected components of marked vertices. We prove two upper bounds on the probability of finding a marked vertex and sketch further research directions.

Coined Quantum Walk

We consider the coined quantum walk with evolution operator:

$$U = S.C$$

where C is the coin operator given by Grover's diffusion operator and S is the flip-flop shift.

For searching, we add a query operator, Q, which flips the sign of the marked vertices, that is,

$$U' = S.C.Q$$

The initial state is the equal superposition over all vertex-direction pairs:

$$|\psi(0)\rangle = \frac{1}{2m}\sum_{v=0}^{n-1}\sum_{c=0}^{d_v-1}|v,c\rangle$$

Where n is the number of vertices, m is the number of edges and d_v is the degree of vertex v.

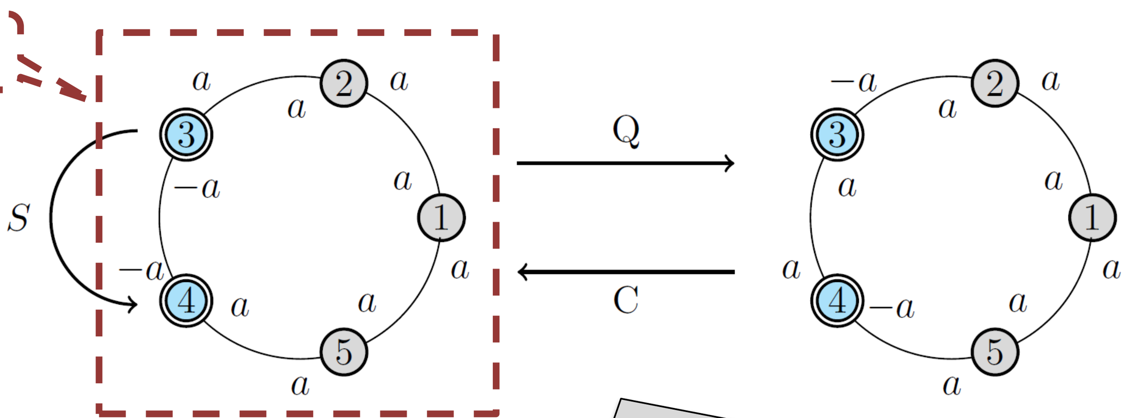
2. Exceptional configurations

General Conditions

A state $|\psi\rangle$ with the following properties is a stationary state of the quantum search algorithm, that is, $U'|\psi\rangle = |\psi\rangle$:

1. All amplitudes of unmarked vertices are equal.
2. The sum of the amplitudes of any marked vertex is 0.
3. The amplitudes of two adjacent vertices pointing to each other are equal.

Stationary state



An example of the application of the evolution operator $U' = S.C.Q$ to a cycle of 5 vertices with two marked vertices ($\mathrm{M} = \{3,4\}$). Labels on edges represent directional amplitudes of a vertex.
The state on the left side is a stationary state. The amplitudes of marked vertices pointing to each other are equal to $-a$, all other amplitudes are equal to a. In this case, the application of the query operator (Q) and the coin operator (C) will flip the sign of amplitudes in the marked vertices.

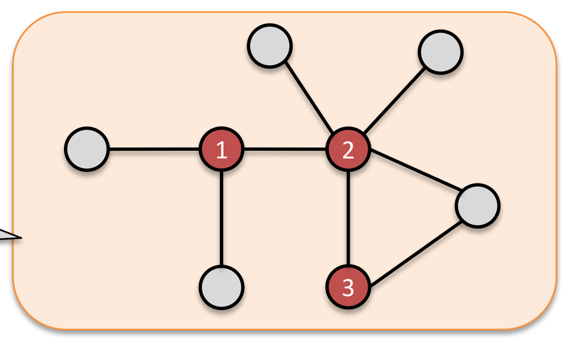
Let $G = (V, E)$ be a graph with a connected set of marked vertices M. According to Ref. [6], the existence of a stationary state depends on whether a marked connected component is bipartite or not, that is,

A **bipartite marked connected component** has a stationary state if and only if the sums of $d_i^{\bar{M}}$ for each bipartite set are equal.

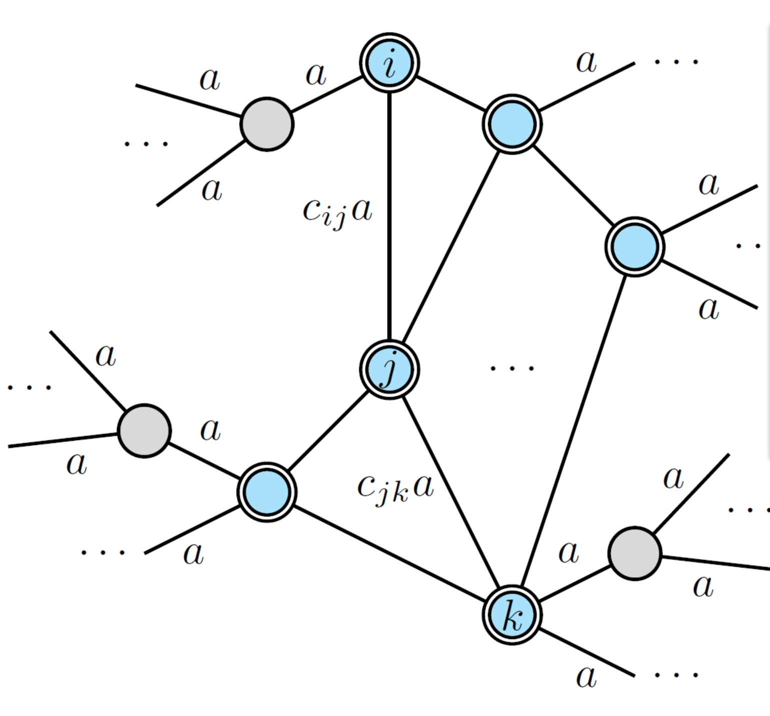
A **non-bipartite marked connected component** always has a stationary state.

$d_i^{\bar{M}}$ is the number of edges from the vertex i to vertices in $V\backslash\mathrm{M}$. $(d_i = d_i^M + d_i^{\bar{M}})$

This bipartite marked component of vertices (in red) has a stationary state. Note that $d_1^{\bar{M}} = 2, d_2^{\bar{M}} = 3$ and $d_3^{\bar{M}} = 1$.



3. Bounds on the probability



Suppose we have a graph with a marked connected component satisfying the aforementioned conditions.
It has a stationary state

$$|\psi_{ST}^a\rangle = \sum_{\substack{i,j\in V\\ i\sim j}} a|i,j\rangle + \sum_{\substack{i,j\in M\\ i\sim j}} (c_{ij}-1)a|i,j\rangle$$

where $j{\sim}i$ means there is an edge connecting vertex j to vertex i.

The sum of the directional amplitudes of a marked vertex must be equal to zero.
Therefore, the amplitudes c_{ij} should satisfy

$$\sum_{\substack{j\in M\\ i\sim j}} c_{ij} = d_i^{\bar{M}} \quad \forall\, i \in M$$

Then our initial state can be written as a sum of the stationary state and another component ($|\psi_{NST}\rangle$) which will change during the evolution,

For $a = 1/\sqrt{2m}$

$$|\psi(0)\rangle = |\psi_{ST}^a\rangle - \sum_{\substack{i,j\in M\\ j\sim i}} (c_{ij}-1)a|i,j\rangle \quad |\psi_{NST}\rangle$$

The probability of finding a marked vertex is maximized when the amplitudes in the changing part will be distributed over the marked vertices only.

Our task is to maximize

$$p_M = \sum_{i\in M}\left(\sum_{\substack{j\in V\backslash M\\ j\sim i}} (a+\alpha)^2 + \sum_{\substack{j\in M\\ j\sim i}} (c_{ij}a + \alpha_{ij})^2\right)$$

$$\sum_{i\in M}\left(\sum_{\substack{j\in V\backslash M\\ j\sim i}} \alpha^2 + \sum_{\substack{j\in M\\ j\sim i}} {\alpha_{ij}}^2\right) = \left||\psi_{NST}\rangle\right|^2$$

Subject to

Then, we obtain that, for any number of steps t,

$$p_M \leq \frac{2}{m}\left(\sum_{\substack{i,j\in M\\ j\sim i}} c_{ij}^2 + 2\,D^{\bar{M}} + 2|E_M|\right)$$

m is the number of edges of the graph.

E_M is the set of edges between marked vertices and $D^{\bar{M}} = \sum_{i\in M} d_i^{\bar{M}}$.

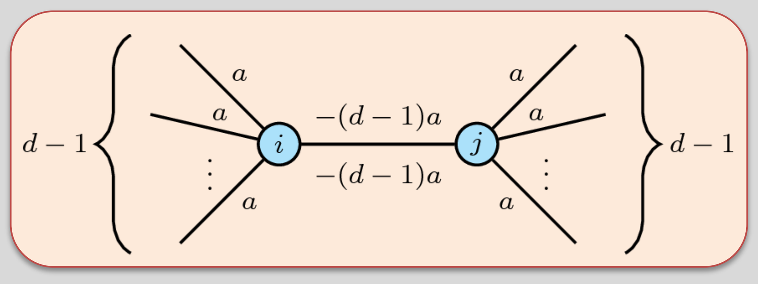
Generalizing for multiple marked components

Consider we have a disjoint set of k marked connected components $\mathrm{M} = \{M_1 \cup M_2 \cup \cdots \cup M_k\}$. Let E_{M_l} be the set of edges with endpoints belonging to the marked component M_l and let $d_i^{\bar{M_l}}$ be the number of edges from the vertex i to vertices in $V\backslash M_l$. Then, it follows that

$$p_M \leq \frac{2}{m}\sum_{l=1}^{k}\left(\sum_{\substack{i,j\in M_l\\ j\sim i}} c_{ij}^2 + 2\,D^{\bar{M_l}} + 2|E_{M_l}|\right)$$

where $D^{\bar{M_l}} = \sum_{i\in M} d_i^{\bar{M_l}}$

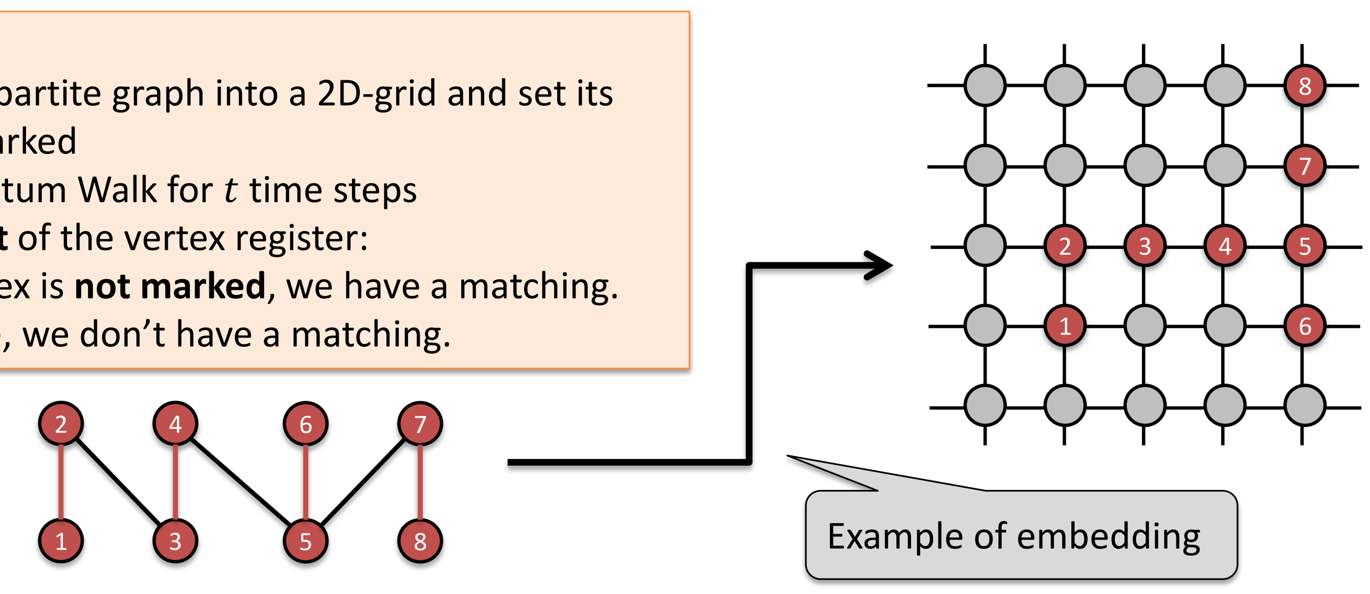
For example, if we consider a d-regular graph with a set of marked vertices which consist of k pairs of adjacent marked vertices (i.e.$|M_1| = |M_2| = \cdots = |M_k| = 2$). Then, the probability of finding a marked vertex, for any number of steps t, is $O\left(\frac{kd^2}{m}\right)$. Note that $D^{\bar{M_l}} = 2(d-1)$ and $|E_{M_l}| = 1$ for all $l = 1, \dots, k$.



4. Applications

We consider the problem of deciding if a **bipartite graph** $G = (V, E)$ has a **perfect matching**. It is known that the bipartite matching problem can be treated as a network flow problem. Using this fact, we claim that if a bipartite graph has a perfect matching then its configuration of marked vertices embedded in the 2D-grid forms a stationary state. A sketch of the algorithm is below.

Algorithm:
1. **Embed** the bipartite graph into a 2D-grid and set its vertices as marked
2. **Run** the Quantum Walk for t time steps
3. **Measurement** of the vertex register:
   If the vertex is **not marked**, we have a matching.
   Otherwise, we don't have a matching.



Example of embedding

5. Final Remarks

We have seen that a placement of marked vertices on a graph can form a stationary state. However, having a stationary state does not automatically mean that the quantum search will not be able to find a marked vertex faster than classically. That is why we need to understand how the probability of finding a marked vertex behaves during the evolution. We proved that the probability is upper bounded by a function on the amplitudes of the stationary state and on the structure of the marked components.
It is still an open problem to find which stationary state gives the minimum probability to find a marked vertex. In this way, we can obtain a tighter bound on the probability. Another interesting question, is whether we can find applications for the exceptional configurations.

References

1. Ambainis, A., Kempe, J., Rivosh, A.: Coins make quantum walks faster. In: Proceedings of the 16th ACM-SIAM Symposium on Discrete Algorithms. (2005) 1099-1108.
2. Shenvi, N., Kempe, J., Whaley, K.B.: A quantum random walk search algorithm. Physical Review A 67(052307) (2003).
3. Nahimovs, N., Rivosh, A.: Quantum Walks on Two-Dimensional Grids with Multiple Marked Locations. In: Proceedings of SOFSEM 2016. (2016) 381-391.
4. N. Nahimovs, R. A. M. Santos. Adjacent vertices can be hard to find by quantum walks. In: Proceedings of SOFSEM (2017).
5. N. Nahimovs, R. A. M. Santos, K. Khadiev. On the probability of finding marked connected components using quantum walks. In Proceedings of CCQ'17 (2017).
6. K. Prusis, J. Vihrovs, and Thomas G. Wong. Stationary states in quantum walk search. Phys. Rev. A, 94:032334.

* This work is supported by the RAQUEL (Grant Agreement No. 323970) project, the Latvian State Research Program NeXIT project No. 1, the ERC Advanced Grant MQC, and the ERDF project number 1.1.1.2/I/16/002.