Decoherence on Staggered Quantum Walks

Raqueline A. M. Santos[1], Franklin L. Marquezino[2]

[1]Center for Quantum Computer Science, University of Latvia - Riga, Latvia
[2]Federal University of Rio de Janeiro – Rio de Janeiro, Brazil
rsantos@lu.lv franklin@cos.ufrj.br

Universidade Federal do Rio de Janeiro

Latvijas Universitate ANNO 1919 UNIVERSITY OF LATVIA

PostDoc Latvia

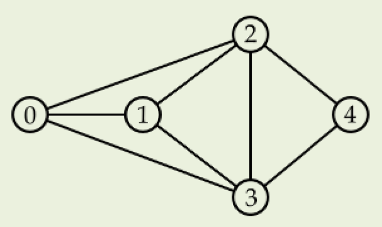
1. Introduction

When implementing quantum systems, decoherence problems are inevitable. These generally undesired effects are present in quantum walk implementations. Hence, it is crucial to understand how decoherence affects them. Decoherence inspired by percolation allows removing of vertices and/or edges in the graph. This type of decoherence was analyzed in many papers [2, 3, 4], using the discrete-time (coined and Szegedy's model) and the continuous-time quantum walk models. Our goal is to analyze decoherence inspired by percolation on Staggered quantum walks [1].

Staggered Quantum Walks

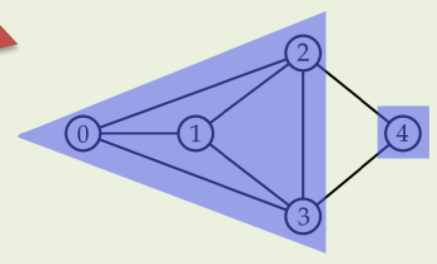
- A quantum walk model have an evolution operator based on local unitary operators.
- Local operators obey the graph structure in the sense that if a particle is on a vertex v, it can move only to its adjacent vertices.
- The Staggered Quantum Walk is obtained by partitioning the vertices into **cliques**.
- An element of the partition is called a **polygon**; The union of polygons is called a **tessellation**; Usually we can define a quantum walk with two tessellations, but depending on the graph more tessellations can be required.
- The Hilbert space is spanned by the vertices of the graph.

The recipe to build the SQW on the graph below is



STEP 1

Make a partition of the vertices so that each element of the partition is a **clique**.



- An element of the partition is called a **polygon** (<u>it is always a clique</u>, but not necessarily a maximal clique);
- <u>The polygons do not overlap and their union contain all vertices</u>;
- The polygon union is called a **tessellation**;

STEP 2

Associate a unit vector to each polygon, for example:

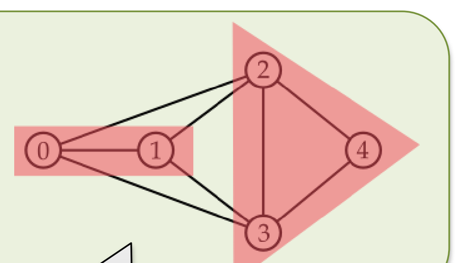
$$|\alpha_0\rangle = \frac{1}{2}(|0\rangle + |1\rangle + |2\rangle + |3\rangle)$$
$$|\alpha_1\rangle = |4\rangle$$

And we have the local unitary operator

$$U_0 = 2|\alpha_0\rangle\langle\alpha_0| + 2|\alpha_1\rangle\langle\alpha_1| - I$$

STEP 3

Make a second vertex partition in order to cover the edges not included in the first tessellation.



We may need to use more than two tessellations.

STEP 4

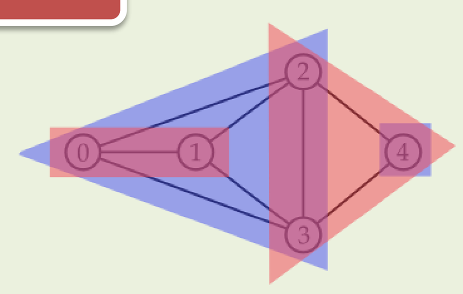
Associate a unit vector in the subspace spanned by the polygon vertices, for example:

$$|\beta_0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$
$$|\beta_1\rangle = \frac{1}{\sqrt{3}}(|2\rangle + |3\rangle + |4\rangle)$$

And we have the local unitary operator

$$U_1 = 2|\beta_0\rangle\langle\beta_0| + 2|\beta_1\rangle\langle\beta_1| - I$$

STEP 5



Since we have covered all edges of the graph, the evolution operator is

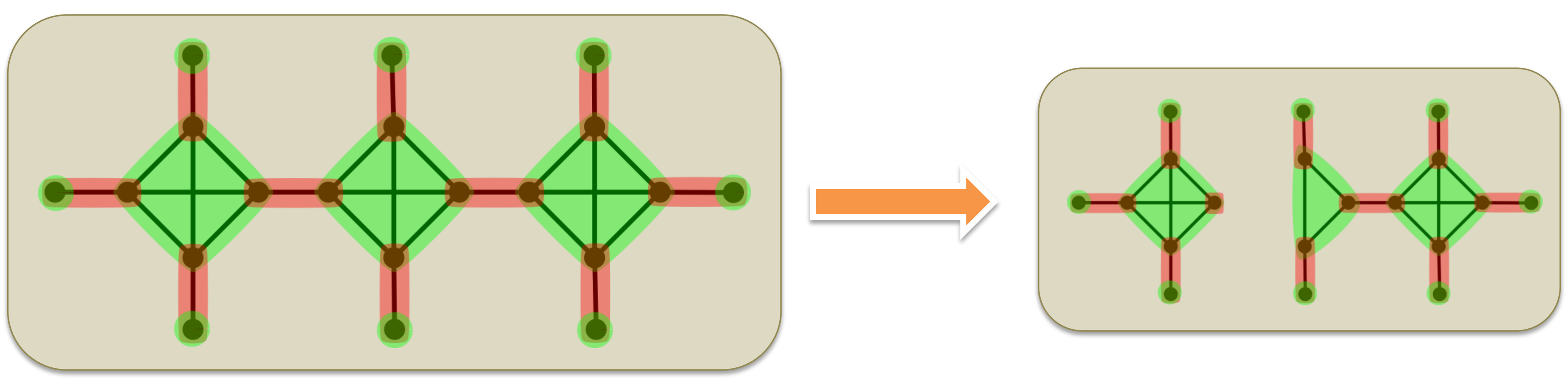
$$U = U_1 \cdot U_0$$

For details, refer to [6].

2. Decoherence Models (inspired on percolation)

Removing Vertices

Removing a vertex ⟺ We should remove the vertex from the polygons which contains it. Small arrangements in the tessellations are needed. Since, each polygon is a clique, by removing one of its vertices it continues to be a clique.



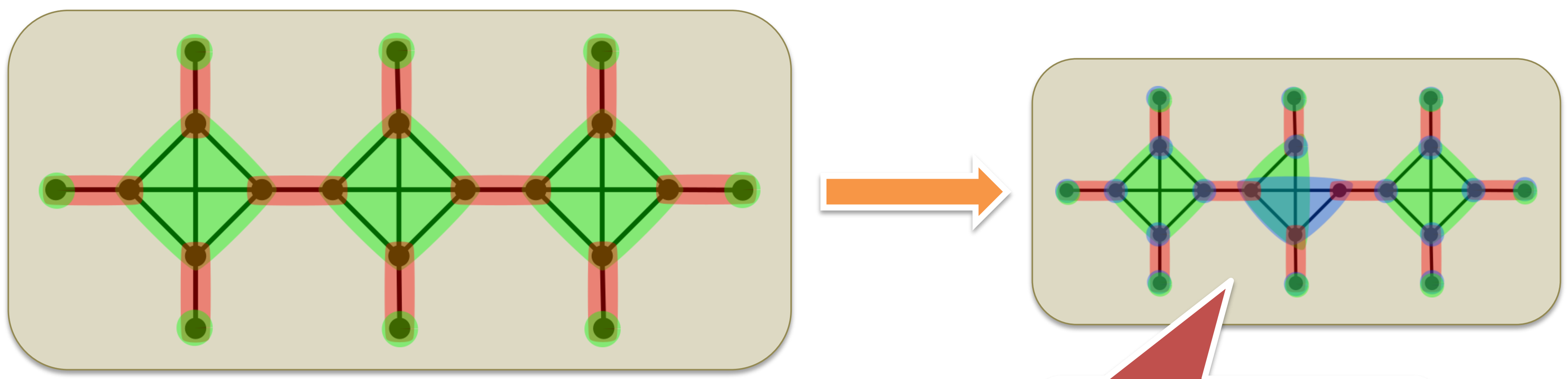
It seems removing a vertex does not increase the tessellation number, and also preserve the "general structure" of the tessellations.

If we consider the staggered model obtained from the coined model. The equivalence of removing a vertex will be "almost" as removing an edge in the coined version. Only if we remove the two adjacent vertices which belongs to the polygon associated to the shift operator, then it will be equivalent to removing an edge in the coined version.

* RAMS was supported by the ERDF project number 1.1.1.2/I/16/002. FLM was supported by FAPERJ E-26/203.235/2016.

Removing Edges

Removing an edge ⟺ Removing edges in the graph may not be simple. Each polygon contains a clique. By removing an edge, the polygon will not contain a clique anymore. In order to fulfill the required properties, new tessellations may be needed. This makes the process non-trivial and it will strictly depend on the structure of the graph.

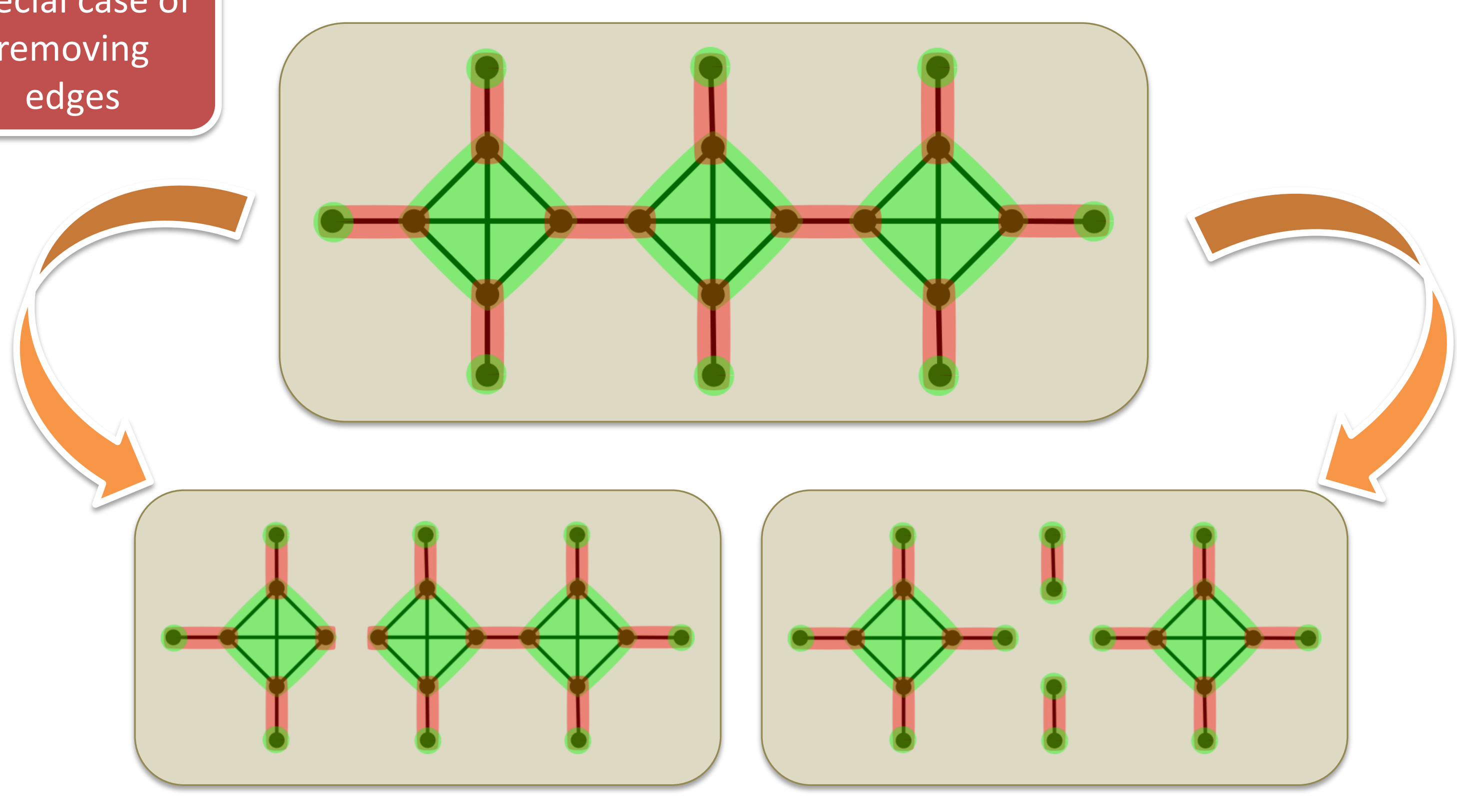


New tessellations may need to be created.

Removing Polygons

Removing a polygon ⟺ Remove the edges inside the clique contained by the polygon. And a new polygon is added for each vertex in the clique.

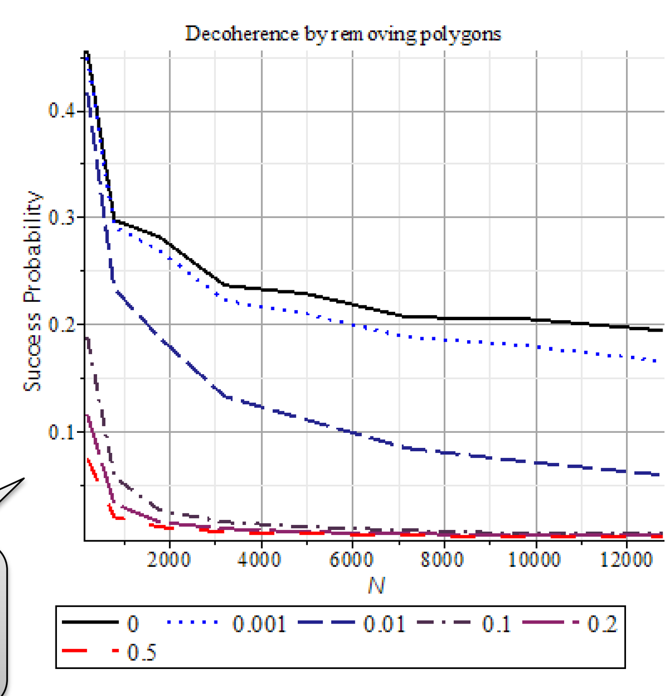
Special case of removing edges



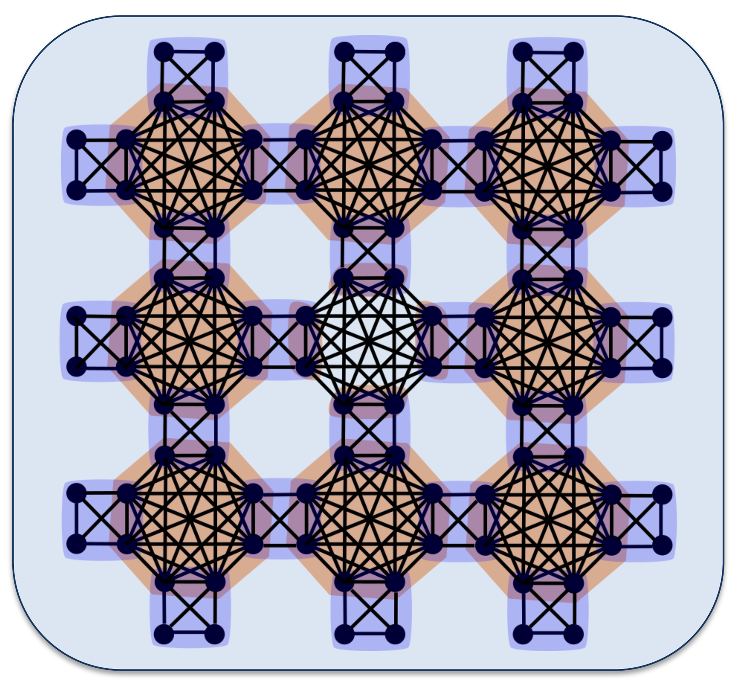
The added polygons are necessary to maintain the property that the tessellation should cover all vertices of the graph. If we don't add them, we will use partial tessellations like in the search algorithm (which is not our goal here).

Numerical experiments

We numerically obtained the success probability of finding a marked 8-clique in the following graph when we randomly remove polygons in the graph with some probability. It would be interesting to obtain the range of probability in which we still have a speedup for the search algorithm.



Average over 100 runs of the algorithm.



3. Final Remarks

This is an ongoing work. So far, we showed how decoherence inspired by percolation can be modeled on staggered quantum walks. Removing vertices seems to not increase the tessellation number, and also preserve the "general structure" of the tessellations. Removing edges can be harder and it would be interesting if we could find a class of graphs where we can handle the removal of an arbitrary edge. Also it would be interesting to obtain analytical results on the range of probability in which we still have a speedup for the search algorithm. Since some instances of the coined model are included in the staggered model (Ref. [5]), we can establish a comparison of how the decoherence affects both models.

References

1. R. Portugal, R. A. M. Santos, T. D. Fernandes, and D. N. Goncalves. *The staggered quantum walk model*. Quantum Information Processing, 15(1):85-101 (2015).
2. A. Romanelli, R. Siri, G. Abal, A. Auyuanet, and R. Donangelo. *Decoherence in the quantum walk on the line*. Physica A, 347(C):137-152 (2005).
3. X. P. Xu and F. Liu. *Continuous-time quantum walks on Erdos-Renyi networks*. Physics Letters A, 372:6727-6732 (2008).
4. R. A. M. Santos, R. Portugal, and M. D. Fragoso. *Decoherence in quantum Markov chains*. Quantum Information Processing, 13(2):559-572 (2014).
5. R. Portugal. *Establishing the equivalence between Szegedy's and coined quantum walks using the staggered model*. Quantum Information Processing, 15(4):1387– 1409 (2016).
6. A. Abreu, L. Cunha, T. Fernandes, C. de Figueiredo, L. Kowada, F. Marquezino, D. Posner, R. Portugal. *The graph tessellation cover number: extremal bounds, efficient algorithms and hardness*, arxiv:1712.07476 (2017) .