Reconfiguring staggered quantum walks with ZX

Bruno Jardim Jaime Santos Luís Soares Barbosa

November 5, 2024

HASLab - INESCTEC

Introduction

Introduction

Optimization of quantum circuits can be seen as a *reconfiguration* process. Indeed the interpretation of such circuits as ZX-diagrams provides a flexible description of quantum computations graphically.

Introduction to the Staggered Quantum Walk

Staggered Quantum Walk

Thought of as the quantum counterpart to classical random walks, quantum walks provide an interesting technique in algorithmic design, with applications in unstructured search, graph algorithmics and communication protocols.

Staggered Quantum Walk

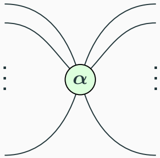
The Staggered Quantum Walks is a recent and very general variant of discrete quantum walks which, avoiding the use of a coin to direct the walker evolution, explores the underlying graph structure to build an evolution operator based on local unitaries induced by adjacent vertices.

Introduction to the ZX-Calculus

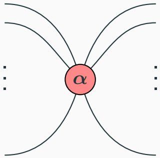
ZX-Calculus

The ZX-calculus is diagrammatic language for reasoning about linear maps between qubits and, as such, about quantum computation in general.

ZX-Calculus - Generators



$$:= \quad |0\cdots0\rangle\langle0\cdots0| + e^{i\alpha}|1\cdots1\rangle\langle1\cdots1|$$



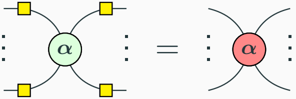
$$:= \quad |+\cdots+\rangle\langle+\cdots+| + e^{i\alpha}|-\cdots-\rangle\langle-\cdots-|$$

ZX-Calculus - Rewrite Rules

- Spider Fusion

- Identity Removal

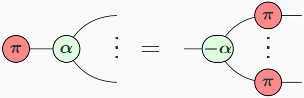
- Color Change



- Hadamard Identity

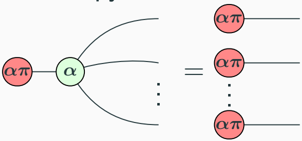
- Bialgebra

- π-commutation



- Hopf

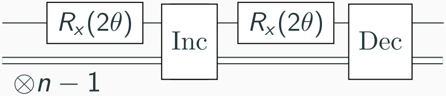
- State copy



Bringing ZX into the picture

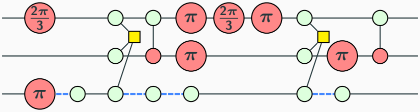
Staggerd Quantum Walk - Circuit

A general implementation of a Staggered Quantum Walk for a line graph.



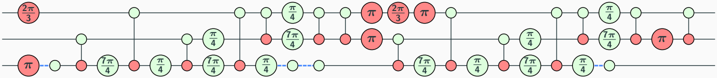
Staggerd Quantum Walk - ZX-diagram

A concrete implementation of a Staggered Quantum Walk for a line graph with 3 qubits and the state $|4\rangle$ as the initial state.

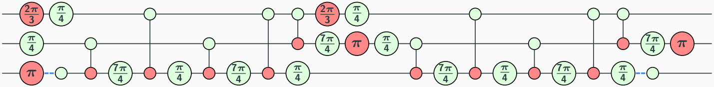


Staggerd Quantum Walk - ZX-diagram

The previous diagram utilizes a notation from the ZH-calculus for the Tofolli gates that greatly simplifies the diagram. Expanding the Tofolli gates it yields



Staggerd Quantum Walk - ZX-diagram



Staggerd Quantum Walk - ZX-diagram

Unfortunately this is the limit one can reasonably optimize the circuit by hand.

This is where PyZX comes in.

PyZX

Automated Diagrammatic Rewriting - PyZX

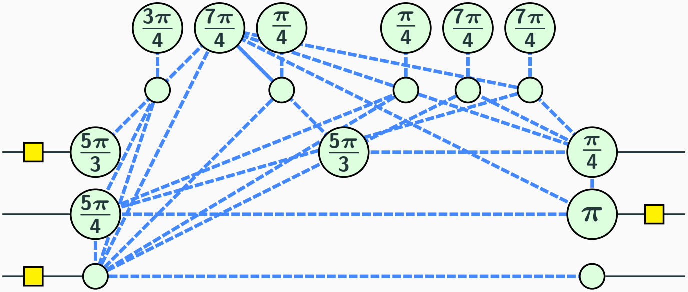
PyZX is a Python tool implementing the theory of ZX-calculus for the creation, visualization, and automated rewriting of large-scale quantum circuits.

Automated Diagrammatic Rewriting - PyZX

Advanced techniques directly implemented in PyZX as the `full_reduce` method, may reduce the circuit T-count in about 50%.

However, the resulting diagram no longer resembles a circuit, making direct comparisons with the original one difficult.

Staggered Quantum Walk - Graph-like



Optimization Comparison - Total Number of Gates

	Number of steps in the staggered quantum walk:			
Optimizations used:	1	2	4	8
None	39	77	153	305
Simple	31	59	115	227
Full-reduce + fusion/id/to_rg	37	47	72	118

Optimization Comparison - T-count

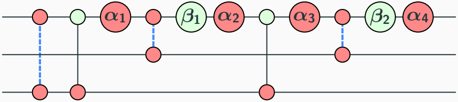
	Number of steps in the staggered quantum walk:			
Optimizations used:	1	2	4	8
None	16	32	64	128
Simple	16	32	64	128
Full-reduce + fusion/id/to_rg	10	16	28	52

An alternative evolution operator

An alternative evolution operator

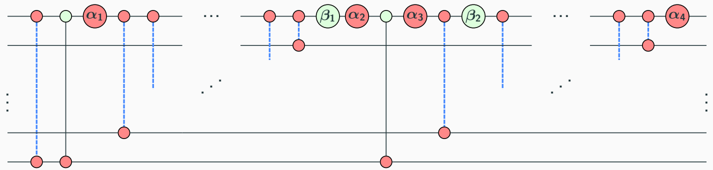
When analyzing the ZX diagram for a long staggered quantum walk (i.e. with more than 5 steps) a pattern starts to emerge, repeating itself as many times as the number of steps considered.

An alternative evolution operator



where $\alpha_n = \pm\frac{\pi}{4}$ and $\beta_n = \frac{2\pi}{3} + m\pi$, with $m = 0$ or $m = 1$.

An alternative evolution operator - Generalization



An alternative evolution operator - Rationale

The rationale behind this operator is easy to explain: it creates a uniform distribution over a certain number of states, applies a rotation that makes some states more likely than others and then spreads these probabilities over the remaining states using CNOT gates. This also explains why the pattern only shows up in staggered quantum walks over a certain length.

An alternative evolution operator - Advantages

- Reduces the total amount of gates needed to represent the evolution of the quantum walk
- Only uses gates controlled by at most 1 qubit
- To go from an n-qubit to an $n+1$ qubit quantum walk, all that needs to be done is to add two more XCX-gates

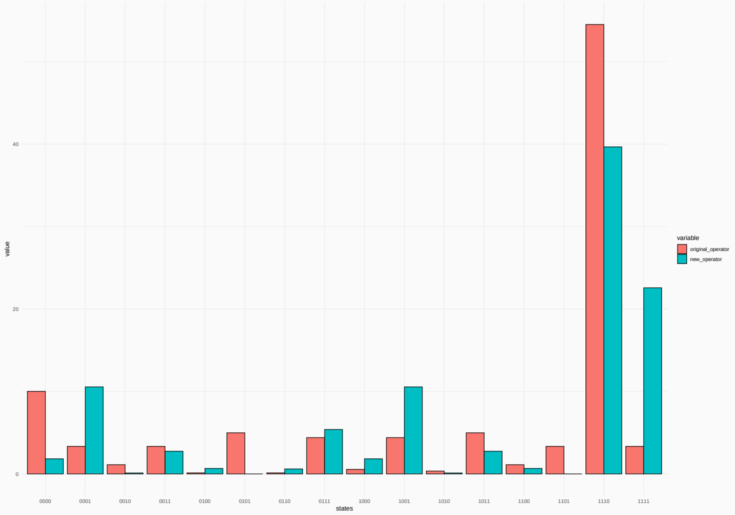
An alternative evolution operator - Advantages

In general, this makes the alternative operator much more efficient with respect to the total number of gates used, leading to lower depth and, therefore, potentially less error-prone circuits.

An alternative evolution operator - Disadvantages

However, a number of challenges remain, requiring further investigation. These concern the most suitable choice of parameters for α_n and β_n, as well as whether and how they depend on the number of qubits used in a particular staggered walk.

An alternative evolution operator - Example



Conclusions and future work

Conclusions

The original, 'intuitive' implementation of a staggered quantum walk can be heavily optimized with respect to both the total number of gates and the T-count value.

It also lead to the identification of an alternative formulation of the evolution operator with a significant reduction in the number of gates involved and thus suitable for running on more limited quantum processing units.

Conclusions - (cont.)

This exercise regards algorithmic optimisation in quantum programming as a *graph reconfiguration* process. This has a huge potential in the development of hybrid quantum-classical algorithms, which are the ones that can actually run in current quantum devices.

They are essentially dynamic in the sense that, depending on a measurement carried over the quantum state, the quantum code running in the quantum device acting as a co-processor is transformed on-the-fly.