Generators and Relations for the Group $\mathrm{O}_n\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$

Sarah Meng Li, Neil Julien Ross, and Peter Selinger

Department of Mathematics and Statistics
Dalhousie University, Halifax, Canada

ATCAT 2020

Integral Clifford+T circuits and $O_n(\mathbb{Z}[1/2])$

- *Integral Clifford+T circuits* are circuits over

$$\{X, CX, CCX, H \otimes H\}.$$

Integral Clifford+T circuits and $O_n(\mathbb{Z}[1/2])$

- *Integral Clifford+T circuits* are circuits over

$$\{X, CX, CCX, H \otimes H\}.$$

- $\mathbb{Z}\left[\frac{1}{2}\right] = \left\{\frac{u}{2^q} \middle| u \in \mathbb{Z}, q \in \mathbb{N}\right\}$ is the ring of *dyadic fractions.*

Integral Clifford+T circuits and $O_n(\mathbb{Z}[1/2])$

- *Integral Clifford+T circuits* are circuits over

$$\{X, CX, CCX, H \otimes H\}.$$

- $\mathbb{Z}\left[\frac{1}{2}\right] = \left\{\frac{u}{2^q} | u \in \mathbb{Z}, q \in \mathbb{N}\right\}$ is the ring of *dyadic fractions.*
- $\mathrm{O}_n(\mathbb{Z}[\frac{1}{2}])$ is the group of orthogonal matrices over $\mathbb{Z}\left[\frac{1}{2}\right]$, namely, the group of *orthogonal dyadic matrices.*

Integral Clifford+T circuits and $O_n(\mathbb{Z}[1/2])$

- *Integral Clifford+T circuits* are circuits over

$$\{X, CX, CCX, H \otimes H\}.$$

- $\mathbb{Z}\left[\frac{1}{2}\right] = \left\{\frac{u}{2^q} | u \in \mathbb{Z}, q \in \mathbb{N}\right\}$ is the ring of *dyadic fractions.*
- $\mathrm{O}_n(\mathbb{Z}[\frac{1}{2}])$ is the group of orthogonal matrices over $\mathbb{Z}\left[\frac{1}{2}\right]$, namely, the group of *orthogonal dyadic matrices.*
- [Amy et al., 2020]: A $2^n \times 2^n$ unitary matrix V can be exactly represented by an n-qubit circuit over $\{X, CX, CCX, H \otimes H\}$ if and only if $V \in O_{2^n}\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$.

Motivation

- Integral Clifford+T circuits play an important role in many quantum algorithms.
- Given an orthogonal dyadic matrix, how to find a circuit for it?
- How to ensure that we find a short circuit?

Basic Gates

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \qquad (-1) = [-1],$$

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \qquad K = H \otimes H = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}.$$

Two-level Operators

Definition

Let $U = \begin{bmatrix} x_{1,1} & x_{1,2} \\ x_{2,1} & x_{2,2} \end{bmatrix}$. The action of $U_{[\alpha,\beta]}$, $1 \leq \alpha < \beta \leq n$, is defined as

$$U_{[\alpha,\beta]} v = w, \text{ where } \begin{cases} \begin{bmatrix} w_\alpha \\ w_\beta \end{bmatrix} = U \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix}, \\ w_i = v_i, i \notin \{\alpha, \beta\}. \end{cases}$$

Example

Let $X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$. Then $X_{[2,3]} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ and $X_{[2,3]} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_3 \\ v_2 \\ v_4 \end{bmatrix}$.

Four-level Operator $U_{[\alpha,\beta,\gamma,\delta]}$

Similarly, we can create a four-level operator by embedding a 4×4 matrix U into an $n \times n$ identity matrix.

Example

$$K = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}. \text{ Then } K_{[1,2,4,6]} = \begin{bmatrix} 1/2 & 1/2 & 0 & 1/2 & 0 & 1/2 \\ 1/2 & -1/2 & 0 & 1/2 & 0 & -1/2 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & -1/2 & 0 & -1/2 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1/2 & -1/2 & 0 & -1/2 & 0 & 1/2 \end{bmatrix}.$$

$$K_{[1,2,4,6]} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \end{bmatrix} = \begin{bmatrix} (v_1 + v_2 + v_4 + v_6)/2 \\ (v_1 - v_2 + v_4 - v_6)/2 \\ v_3 \\ (v_1 + v_2 - v_4 - v_6)/2 \\ v_5 \\ (v_1 - v_2 - v_4 + v_6)/2 \end{bmatrix}.$$

Generators of $O_n(\mathbb{Z}[1/2])$

- Our generating set:

$$\mathcal{G} = \left\{(-1)_{[\alpha]}, X_{[\alpha,\beta]}, K_{[\alpha,\beta,\gamma,\delta]} : 1 \leq \alpha < \beta < \gamma < \delta \leq n\right\}.$$

Exact Synthesis of Integral *Clifford*+T Circuits

Theorem (Amy et al., 2020)

Let M be a unitary $n \times n$ matrix. Then $M \in \mathrm{O}_n\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$ if and only if M can be written as a product of elements of $\mathcal{G}$.

Proof

$\Leftarrow$) $\mathcal{G} \subset \mathrm{O}_n\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$ and $\mathrm{O}_n\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$ is closed under multiplication.

Synthesis Algorithm in a Nutshell

Proof

⇒) For every $M \in \mathrm{O}_n\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$, construct a sequence of generators representing M.

$$M \xrightarrow{\overrightarrow{G_1}} \left(\begin{array}{ccc|c} & & & 0 \\ & M' & & \vdots \\ & & & 0 \\ \hline 0 & \cdots & 0 & 1 \end{array}\right) \xrightarrow{\overrightarrow{G_2}} \left(\begin{array}{ccc|cc} & & & 0 & 0 \\ & M'' & & \vdots & \vdots \\ & & & 0 & 0 \\ \hline 0 & \cdots & 0 & 1 & 0 \\ 0 & \cdots & 0 & 0 & 1 \end{array}\right) \xrightarrow{\overrightarrow{G_3}} \cdots \xrightarrow{\overrightarrow{G_\ell}} \mathbb{I}$$

$$\overrightarrow{G_\ell} \cdot \ldots \cdot \overrightarrow{G_1} M = \mathbb{I} \Rightarrow M = (\overrightarrow{G_\ell} \cdot \ldots \cdot \overrightarrow{G_1})^{-1}.$$

Characterize Integral Clifford+T Circuits

Corollary (Amy et al., 2020)

$\mathcal{G}$ can be exactly represented by integral Clifford+T circuits using at most one clean ancilla.

Theorem (Amy et al., 2020)

A $2^n \times 2^n$ unitary matrix V can be exactly represented by an n-qubit circuit over $\{X, CX, CCX, H \otimes H\}$ if and only if $V \in O_{2^n}\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$.

Complexity of a Vector

Definition (Least Denominator Exponent)

Let $t \in \mathbb{Z}\left[\frac{1}{2}\right]$. A natural number $k \in \mathbb{N}$ is a *denominator exponent* for t if $2^k t \in \mathbb{Z}$. The least such k is called the *least denominator exponent* of t, written $\operatorname{lde}(t)$.

Lemma

Let $v \in \mathbb{Z}\left[\frac{1}{2}\right]^n$ be a unit vector. Let $k = \operatorname{lde}(v)$. If $k = 0$, then $v = \pm e_j$ for some $j \in \{1, \cdots, n\}$.

Correctness of the Synthesis Algorithm

Lemma (Parity)

Let u_1, u_2, u_3, u_4 be odd integers. Then there exists $\tau_1, \tau_2, \tau_3, \tau_4 \in \mathbb{Z}_2$ such that

$$K_{[1,2,3,4]}(-1)_{[1]}^{\tau_1}(-1)_{[2]}^{\tau_2}(-1)_{[3]}^{\tau_3}(-1)_{[4]}^{\tau_4}\begin{bmatrix}u_1\\u_2\\u_3\\u_4\end{bmatrix}=\begin{bmatrix}u_1'\\u_2'\\u_3'\\u_4'\end{bmatrix},\ u_1', u_2', u_3', u_4' \text{ are even integers.}$$

Correctness of the Synthesis Algorithm

Lemma (Parity)

Let u_1, u_2, u_3, u_4 be odd integers. Then there exists $\tau_1, \tau_2, \tau_3, \tau_4 \in \mathbb{Z}_2$ such that

$$K_{[1,2,3,4]}(-1)_{[1]}^{\tau_1}(-1)_{[2]}^{\tau_2}(-1)_{[3]}^{\tau_3}(-1)_{[4]}^{\tau_4}\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} u_1' \\ u_2' \\ u_3' \\ u_4' \end{bmatrix},\ u_1', u_2', u_3', u_4' \text{ are even integers.}$$

Lemma (Counts)

Let $v \in \mathbb{Z}\left[\frac{1}{2}\right]^n$ be a unit vector, and $\operatorname{lde}(v) = k > 0$. Let $w = 2^k v$. Then the number of odd entries in w is a multiple of 4.

Correctness of the Synthesis Algorithm

Lemma (Parity)

Let u_1, u_2, u_3, u_4 be odd integers. Then there exists $\tau_1, \tau_2, \tau_3, \tau_4 \in \mathbb{Z}_2$ such that

$$K_{[1,2,3,4]}(-1)^{\tau_1}_{[1]}(-1)^{\tau_2}_{[2]}(-1)^{\tau_3}_{[3]}(-1)^{\tau_4}_{[4]} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} u_1' \\ u_2' \\ u_3' \\ u_4' \end{bmatrix}, \ u_1', u_2', u_3', u_4' \text{ are even integers.}$$

Lemma (Counts)

Let $v \in \mathbb{Z}\left[\frac{1}{2}\right]^n$ be a unit vector, and $\operatorname{lde}(v) = k > 0$. Let $w = 2^k v$. Then the number of odd entries in w is a multiple of 4.

Proof.

Let $w = 2^k v \in \mathbb{Z}^n$. Since $v^\intercal v = 1$, we have $w^\intercal w = 4^k$ and therefore $\sum w_j^2 = 4^k$. Note that $w_j^2 \equiv 1(4)$ if and only if w_j is odd and $w_j^2 \equiv 0(4)$ if and only if w_j is even. Hence the number of w_j such that $w_j^2 \equiv 1(4)$ is a multiple of 4. □

Example (Input: $v \in \mathbb{Z}\left[\frac{1}{2}\right]^8$ Output: G_1, G_2, G_3 Result: $G_3 \cdot G_2 \cdot G_1 \cdot v = e_1$)

$$v : \frac{1}{4}\begin{pmatrix} -1 \\ 1 \\ -1 \\ -1 \\ 3 \\ 1 \\ 1 \\ 1 \end{pmatrix} \xrightarrow{G_1 = K_{[1,2,3,4]}(-1)_{[4]}(-1)_{[3]}(-1)_{[1]}} v' : \frac{1}{4}\begin{pmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 3 \\ 1 \\ 1 \\ 1 \end{pmatrix} \xrightarrow{G_2 = K_{[5,6,7,8]}(-1)_{[5]}}$$

$\mathrm{lde}(v) = 2$ $\mathrm{lde}(v') = 2$

$$v'' : \frac{1}{4}\begin{pmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ -2 \\ -2 \\ -2 \end{pmatrix} = \frac{1}{2}\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ -1 \\ -1 \end{pmatrix} \xrightarrow{G_1 = K_{[1,6,7,8]}(-1)_{[8]}(-1)_{[7]}(-1)_{[6]}} v''' : \frac{1}{2}\begin{pmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = e_1.$$

$\mathrm{lde}(v'') = 1$ $\mathrm{lde}(v''') = 0$

Correctness of the Synthesis Algorithm

Lemma (Reducibility)

Let $v \in \mathbb{Z}\left[\frac{1}{2}\right]^n$ be a unit vector. Let $k = \operatorname{lde}(v)$. If $k > 0$, then there exists a sequence of generators $G_1, \ldots, G_\ell$ such that $\operatorname{lde}(G_\ell \cdot \ldots \cdot G_1 v) < k$.

Correctness of the Synthesis Algorithm

Lemma (Reducibility)

Let $v \in \mathbb{Z}\left[\frac{1}{2}\right]^n$ *be a unit vector. Let* $k = \operatorname{lde}(v)$*. If* $k > 0$*, then there exists a sequence of generators* $G_1, \ldots, G_\ell$ *such that* $\operatorname{lde}(G_\ell \cdot \ldots \cdot G_1 v) < k$*.*

Lemma (Column Reduction)

Let $v \in \mathbb{Z}\left[\frac{1}{2}\right]^n$ *be a unit vector. Then there exists a sequence of generators* $G_1, \ldots, G_\ell$ *such that* $G_\ell \cdot \ldots \cdot G_1 v = e_j$*.*

Correctness of the Synthesis Algorithm

Lemma (Reducibility)

Let $v \in \mathbb{Z}\left[\frac{1}{2}\right]^n$ be a unit vector. Let $k = \mathrm{lde}(v)$. If $k > 0$, then there exists a sequence of generators $G_1, \ldots, G_\ell$ such that $\mathrm{lde}(G_\ell \cdot \ldots \cdot G_1 v) < k$.

Lemma (Column Reduction)

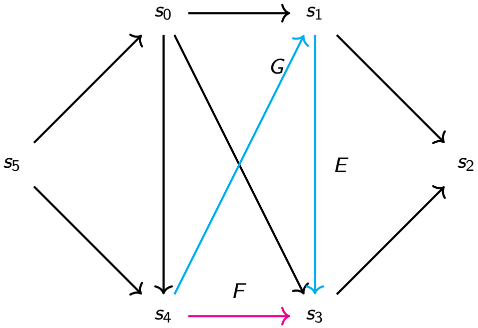
Let $v \in \mathbb{Z}\left[\frac{1}{2}\right]^n$ be a unit vector. Then there exists a sequence of generators $G_1, \ldots, G_\ell$ such that $G_\ell \cdot \ldots \cdot G_1 v = e_j$.

Lemma

If $M \in \mathrm{O}_n\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$, then M can be written as a product of generators from $\mathcal{G}$.

Graph Representation of $\mathrm{O}_n\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$

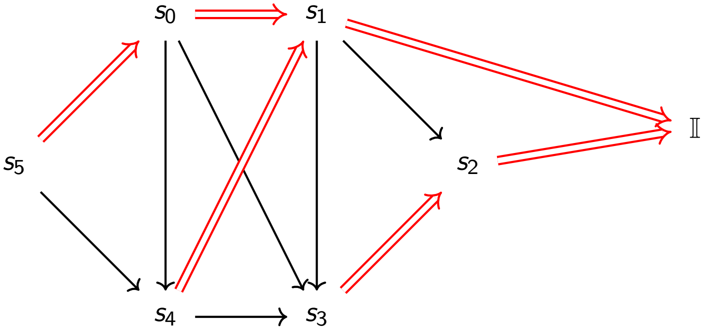
1. Build a graph for $\mathrm{O}_n\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$.



- Vertex = group element (aka, operators, matrices, states).
- Edge = a sequence of generators (e.g., $Fs_4 = s_3$).
- Cycle = relation (e.g., $EG = F$).

Proof of Completeness

2. The exact synthesis algorithm gives a canonical path from each group element to $\mathbb{I}$.



Semantic Equivalence

- A *word* is a sequence of generators. We write $\overrightarrow{G}$ for $G_q \ldots G_1$.
- Each operator has a unique *normal form*, which is the word output by the exact synthesis algorithm.
- The *interpretation* of $\overrightarrow{G}$ is $[\![\overrightarrow{G}]\!] = G_q \cdot \ldots \cdot G_1$.

Definition

Two words $\overrightarrow{G}$ and $\overrightarrow{F}$ are *semantically equivalent*, written $\overrightarrow{G} \sim \overrightarrow{F}$, if $[\![\overrightarrow{G}]\!] = [\![\overrightarrow{F}]\!]$.

Motivation

Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be two circuits where

$$\mathcal{C}_1 = X_{[1,2]}X_{[3,4]}X_{[1,2]}, \qquad \mathcal{C}_2 = X_{[3,4]}.$$

To see if $\mathcal{C}_1 \sim \mathcal{C}_2$, we can check

- by direct computation;
- or by simplifying $\mathcal{C}_1$:

$$\mathcal{C}_1 = X_{[1,2]}X_{[3,4]}X_{[1,2]} \sim X_{[1,2]}X_{[1,2]}X_{[3,4]} \sim \mathbb{I}X_{[3,4]} \sim X_{[3,4]} = \mathcal{C}_2.$$

Syntactic Equivalence

Definition

Two words $\overrightarrow{G}$ and $\overrightarrow{F}$ are *syntactically equivalent*, written $\overrightarrow{G} \approx \overrightarrow{F}$, where $\approx$ is the smallest congruence relation on words containing $R_1, \ldots, R_k$ and such that

$$\overrightarrow{G} \approx \overrightarrow{G'}, \overrightarrow{F} \approx \overrightarrow{F'} \Rightarrow \overrightarrow{G}\,\overrightarrow{F} \approx \overrightarrow{G'}\,\overrightarrow{F'}.$$

Question: Can we use syntactic and semantic relations interchangeably?

Goal

Theorem (Analogous to Greylyn's Theorem, 2014)

Let $\overrightarrow{G}$ *and* $\overrightarrow{F}$ *be words over* $\mathcal{G}$ *of* $\mathrm{O}_n\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$, *then*

$$\overrightarrow{G} \approx \overrightarrow{F} \iff \overrightarrow{G} \sim \overrightarrow{F}$$

Theorem (Soundness)

$\overrightarrow{G} \approx \overrightarrow{F} \Rightarrow \overrightarrow{G} \sim \overrightarrow{F}$

Proof

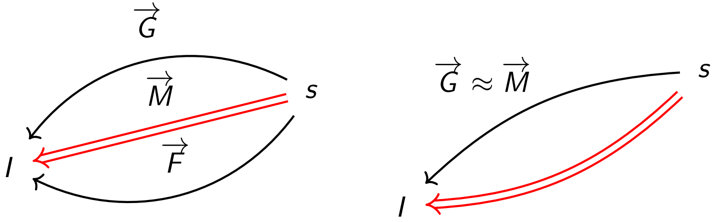
By matrix multiplication.

Theorem (Completeness)

$\overrightarrow{G} \sim \overrightarrow{F} \Rightarrow \overrightarrow{G} \approx \overrightarrow{F}$

Proof Idea

If two words are semantically equivalent, they corresponds to the same normal form. If we can reduce an arbitrary path to its normal form using **syntactic relations**, this implies completeness.



A Complete Set of Syntactic Relations

$$
\begin{aligned}
X_{[a,b]}^2 &\approx \epsilon \\
(-1)_{[a]}^2 &\approx \epsilon \\
K_{[a,b,c,d]}^2 &\approx \epsilon \\
X_{[a,b]}X_{[c,d]} &\approx X_{[c,d]}X_{[a,b]} \\
X_{[a,b]}(-1)_{[c]} &\approx (-1)_{[c]}X_{[a,b]} \\
X_{[a,b]}K_{[c,d,e,f]} &\approx K_{[c,d,e,f]}X_{[a,b]} \\
(-1)_{[a]}(-1)_{[b]} &\approx (-1)_{[b]}(-1)_{[a]} \\
(-1)_{[a]}K_{[b,c,d,e]} &\approx K_{[b,c,d,e]}(-1)_{[a]} \\
K_{[a,b,c,d]}K_{[e,f,g,h]} &\approx K_{[e,f,g,h]}K_{[a,b,c,d]} \\
K_{[a,b,c,d]}K_{[b,d,e,f]} &\approx K_{[c,d,e,f]}K_{[a,b,c,e]}
\end{aligned}
$$

$$
\begin{aligned}
X_{[a,a']}X_{[a,b]} &\approx X_{[a',b]}X_{[a,a']} \\
X_{[b,b']}X_{[a,b]} &\approx X_{[a,b']}X_{[b,b']} \\
X_{[a,b]}(-1)_{[b]} &\approx (-1)_{[a]}X_{[a,b]} \\
X_{[a,a']}K_{[a,b,c,d]} &\approx K_{[a',b,c,d]}X_{[a,a']} \\
X_{[b,b']}K_{[a,b,c,d]} &\approx K_{[a,b',c,b]}X_{[b,b']} \\
X_{[c,c']}K_{[a,b,c,d]} &\approx K_{[a,b,c',d]}X_{[c,c']} \\
X_{[d,d']}K_{[a,b,c,d]} &\approx K_{[a,b,c,d']}X_{[d,d']} \\
X_{[a,b]}K_{[a,b,c,d]} &\approx K_{[a,b,c,d]}X_{[b,d]}(-1)_{[b]}(-1)_{[d]} \\
X_{[c,d]}K_{[a,b,c,d]} &\approx K_{[a,b,c,d]}X_{[b,d]} \\
X_{[b,c]}K_{[a,b,c,d]} &\approx (-1)_{[a]}K_{[a,b,c,d]}(-1)_{[a]}K_{[a,b,c,d]}(-1)_{[a]}
\end{aligned}
$$

$$
K_{[e,f,g,h]}K_{[a,b,c,d]}X_{[d,e]}K_{[a,b,c,d]}K_{[e,f,g,h]}
$$

$$
\approx
$$

$$
(-1)_{[a]}(-1)_{[h]}X_{[a,h]}K_{[e,f,g,h]}K_{[a,b,c,d]}X_{[d,e]}K_{[a,b,c,d]}K_{[e,f,g,h]}X_{[a,h]}(-1)_{[a]}(-1)_{[h]}
$$

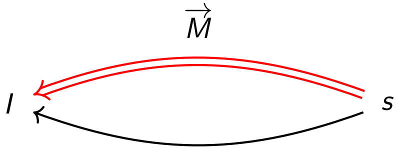
Proof of Completeness

Use induction to leverage **finitely** many syntactic relations such that an arbitrary path can be rewritten into its equivalent canonical path.

Lemma 1

Let $s \xrightarrow{\vec{G}} I$ be any sequence of simple edges with final state I, and let $s \overset{\vec{M}}{\Longrightarrow} I$ be the unique sequence of normal edges from s to I. Then $\vec{G} \approx \vec{M}$.

To prove Lemma 1, we proceed by induction on the length of $\vec{G}$.

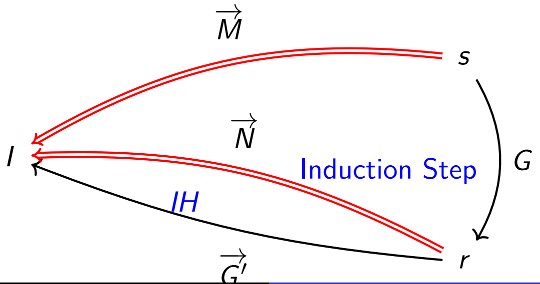


Proof of Completeness

Lemma 1

Let $s \xrightarrow{\overrightarrow{G}} I$ be any sequence of simple edges with final state I, and let $s \overset{\overrightarrow{M}}{\Longrightarrow} I$ be the unique sequence of normal edges from s to I. Then $\overrightarrow{G} \approx \overrightarrow{M}$.

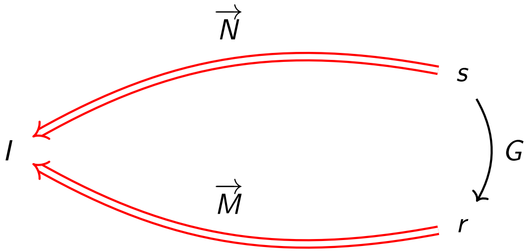
To prove Lemma 1, we proceed by induction on the length of $\overrightarrow{G}$.



Lemma 2

Let $s \xrightarrow{G} r$ be a simple edge. Let $s \overset{\vec{N}}{\Longrightarrow} I$ be the unique sequence of normal edges from s to I, $r \overset{\vec{M}}{\Longrightarrow} I$ be the unique sequence of normal edges from r to I. Then $\vec{M}G \approx \vec{N}$.

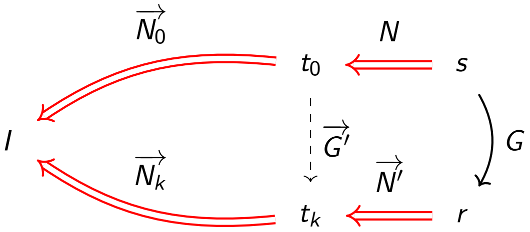
To prove Lemma 2, we proceed by induction on the level of s.



Lemma 2

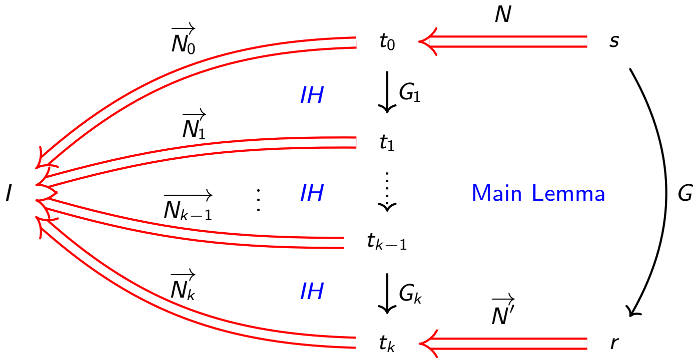
Let $s \xrightarrow{G} r$ be a simple edge. Let $s \overset{\overrightarrow{N}}{\Longrightarrow} I$ be the unique sequence of normal edges from s to I, $r \overset{\overrightarrow{M}}{\Longrightarrow} I$ be the unique sequence of normal edges from r to I. Then $\overrightarrow{M}G \approx \overrightarrow{N}$.

To prove Lemma 2, we proceed by induction on the level of s.



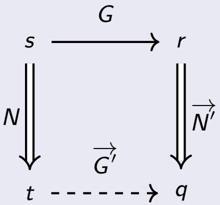
Lemma 2

Let $s \xrightarrow{G} r$ *be a simple edge. Let* $s \overset{\overrightarrow{N}}{\Longrightarrow} I$ *be the unique sequence of normal edges from* s *to* I*,* $r \overset{\overrightarrow{M}}{\Longrightarrow} I$ *be the unique sequence of normal edges from* r *to* I*. Then* $\overrightarrow{M}G \approx \overrightarrow{N}$*.*



Main Lemma

Let s, t, and r be states, $N : s \Rightarrow t$ be a normal edge, and $G : s \rightarrow r$ be a simple edge. Then there exists a state q, a sequence of normal edges $\overrightarrow{N'} : r \Rightarrow q$ and a sequence of simple edges $\overrightarrow{G'} : t \rightarrow q$ such that the diagram



commutes syntactically and level$(\overrightarrow{G'} : t \rightarrow q) <$ level(s).

Proof

Since t and N are uniquely determined by s, and r is uniquely determined by G, it suffices to distinguish cases based on the pair (s, G).

Basic Edges

Definition

Consider

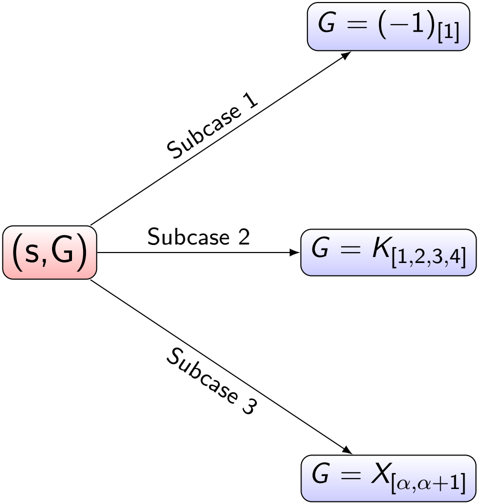
$$\mathcal{G}' = \{X_{[\alpha,\alpha+1]}, K_{[1,2,3,4]}, (-1)_{[1]} \mid 1 \leq \alpha \leq n-1\}$$

and $\mathcal{G}' \subset \mathcal{G}$. We call an element from $\mathcal{G}$ a simple generator, an element from $\mathcal{G}'$ a basic generator. Furthermore, an edge $s \xrightarrow{G} t$ is simple if G is a simple generator. An edge $s \xrightarrow{G} t$ is basic if G is a basic generator.

Lemma

Basic edges and simple edges can be used interchangeably while the levels of edges are respected.

Proof by Cases



A Complete Set of Syntactic Relations

$$
\begin{aligned}
X_{[a,b]}^2 &\approx \epsilon \\
(-1)_{[a]}^2 &\approx \epsilon \\
K_{[a,b,c,d]}^2 &\approx \epsilon \\
X_{[a,b]}X_{[c,d]} &\approx X_{[c,d]}X_{[a,b]} \\
X_{[a,b]}(-1)_{[c]} &\approx (-1)_{[c]}X_{[a,b]} \\
X_{[a,b]}K_{[c,d,e,f]} &\approx K_{[c,d,e,f]}X_{[a,b]} \\
(-1)_{[a]}(-1)_{[b]} &\approx (-1)_{[b]}(-1)_{[a]} \\
(-1)_{[a]}K_{[b,c,d,e]} &\approx K_{[b,c,d,e]}(-1)_{[a]} \\
K_{[a,b,c,d]}K_{[e,f,g,h]} &\approx K_{[e,f,g,h]}K_{[a,b,c,d]} \\
K_{[a,b,c,d]}K_{[b,d,e,f]} &\approx K_{[c,d,e,f]}K_{[a,b,c,e]}
\end{aligned}
$$

$$
\begin{aligned}
X_{[a,a']}X_{[a,b]} &\approx X_{[a',b]}X_{[a,a']} \\
X_{[b,b']}X_{[a,b]} &\approx X_{[a,b']}X_{[b,b']} \\
X_{[a,b]}(-1)_{[b]} &\approx (-1)_{[a]}X_{[a,b]} \\
X_{[a,a']}K_{[a,b,c,d]} &\approx K_{[a',b,c,d]}X_{[a,a']} \\
X_{[b,b']}K_{[a,b,c,d]} &\approx K_{[a,b',c,b]}X_{[b,b']} \\
X_{[c,c']}K_{[a,b,c,d]} &\approx K_{[a,b,c',d]}X_{[c,c']} \\
X_{[d,d']}K_{[a,b,c,d]} &\approx K_{[a,b,c,d']}X_{[d,d']} \\
X_{[a,b]}K_{[a,b,c,d]} &\approx K_{[a,b,c,d]}X_{[b,d]}(-1)_{[b]}(-1)_{[d]} \\
X_{[c,d]}K_{[a,b,c,d]} &\approx K_{[a,b,c,d]}X_{[b,d]} \\
X_{[b,c]}K_{[a,b,c,d]} &\approx (-1)_{[a]}K_{[a,b,c,d]}(-1)_{[a]}K_{[a,b,c,d]}(-1)_{[a]}
\end{aligned}
$$

$$
K_{[e,f,g,h]}K_{[a,b,c,d]}X_{[d,e]}K_{[a,b,c,d]}K_{[e,f,g,h]}
$$

$$
\approx
$$

$$
(-1)_{[a]}(-1)_{[h]}X_{[a,h]}K_{[e,f,g,h]}K_{[a,b,c,d]}X_{[d,e]}K_{[a,b,c,d]}K_{[e,f,g,h]}X_{[a,h]}(-1)_{[a]}(-1)_{[h]}
$$

Future Work

- Improve the complexity of the synthesis algorithm:

$$O(2^{2^n} nk) \xrightarrow[\text{Decomposition}]{\text{Householder}} O(4^n nk) \xrightarrow[\text{Synthesis}]{\text{Global}} O(?)$$

Future Work

- Improve the complexity of the synthesis algorithm:

$$O(2^{2^n} nk) \xrightarrow[\text{Decomposition}]{\text{Householder}} O(4^n nk) \xrightarrow[\text{Synthesis}]{\text{Global}} O(?)$$

- Interpret syntactic relations in terms of quantum circuit relations.

Future Work

- Improve the complexity of the synthesis algorithm:

$$O(2^{2^n} nk) \xrightarrow[\text{Decomposition}]{\text{Householder}} O(4^n nk) \xrightarrow[\text{Synthesis}]{\text{Global}} O(?)$$

- Interpret syntactic relations in terms of quantum circuit relations.
- Find a minimal set of syntactic relations for $\mathrm{O}_n\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$.

Future Work

- Improve the complexity of the synthesis algorithm:

$$O(2^{2^n} nk) \xrightarrow[\text{Decomposition}]{\text{Householder}} O(4^n nk) \xrightarrow[\text{Synthesis}]{\text{Global}} O(?)$$

- Interpret syntactic relations in terms of quantum circuit relations.
- Find a minimal set of syntactic relations for $\mathrm{O}_n\left(\mathbb{Z}\left[\frac{1}{2}\right]\right)$.
- Find syntactic relations for other restricted Clifford+T matrix groups (e.g., imaginary Clifford+T circuits).

Thank You!