Cambridge Quantum

REPRESENTING MATRICES USING ALGEBRAIC *Zx-calculus*

by Quanlong Wang

Cambridge Quantum
QUANLONG WANG
harny.wang@cambridgequantum.com

Cambridge Quantum
Terrington House
13-15 Hills Road
Cambridge CB2 1NL
United Kingdom

Published by Cambridge Quantum

09 September 2021

Representing Matrices Using Algebraic ZX-calculus

Quanlong Wang
Cambridge Quantum Computing Ltd.
harny.wang@cambridgequantum.com

Abstract

Elementary matrices play an important role in linear algebra applications. In this paper, we represent all the elementary matrices of size $2^m \times 2^m$ using algebraic ZX-calculus. Then we show their properties on inverses and transpose using rewriting rules of ZX-calculus. As a consequence, we are able to depict any matrices of size $2^m \times 2^n$ by string diagrams without resort to a diagrammatic normal form for matrices as shown in [15]. By doing so we pave the way towards visualising by string diagrams important matrix technologies deployed in AI especially machine learning.

1 Introduction

Matrices are used everywhere in modern science, like machine learning [11] or quantum computing [12], to name a few. Meanwhile, there is a graphical language called ZX-calculus that could also deal with matrix calculations such as matrix multiplication and tensor product [2, 3]. Then there naturally arises a question: why are people bothering with using diagrams for matrix calculations given that matrix technology has been applied with great successes? There are a few reasons for doing so. First, there is a lot of redundancy in matrix calculations which could be avoided in graphical calculus. For example, to prove the cyclic property of matrices $tr(AB) = tr(BA)$, all the elements of the two matrices will be involved, while in graphical language like ZX-calculus, the proof of the cyclic property is almost a tautology [4]. Second, matrix calculations always have all the elements of matrices involved, thus a "global" operation, while in ZX-calculus, the operations are just diagram rewriting where only a part of a diagram is replaced by another sub-diagram according to certain rewriting rule, thus essentially a "local" operation which makes things much easier. Finally, graphical calculus is much more intuitive than matrix calculation, therefore a pattern/structure is more probably to be recognised in a graphical formalism. In fact, as a graphical calculus for matrix calculation, ZX-calculus has achieved plenty of successes in the field of quantum computing and information [1, 5, 7, 13]

For research realm beyond quantum, traditional ZX-calculus [2] is inconvenient as "it lacks a way to directly encode the complex numbers" [18]. To remedy this inconvenience while still preserving its powerfulness, we introduced an "extended" version called algebraic ZX-calculus with new generators added in (not a real extension in the sense that the new generators can be expressed by the original generators) [14].

In this paper, we first represent elementary matrices [8] of size $2^m \times 2^m$ in ZX diagrams with some examples, then we show by diagrams some properties of elementary matrices regarding their inverses and transpose. Finally we demonstrate how to represent an arbitrary matrix of size $2^m \times 2^n$ in ZX diagrams. The results of this paper can be generalised from the filed of complex numbers to arbitrary commutative semirings due to the completeness of algebraic ZX-calculus over semirings [16].

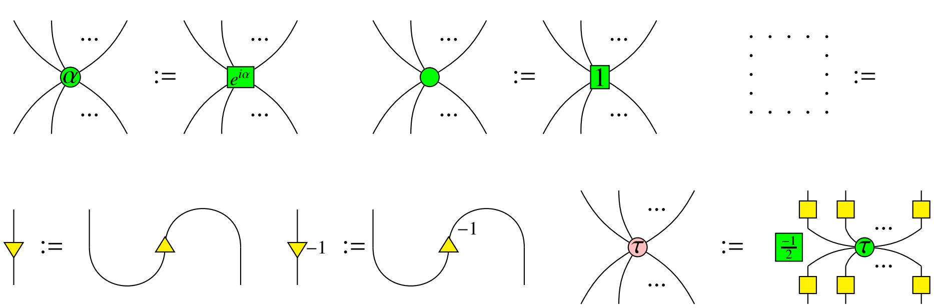
2 Algebraic ZX-calculus

In this section, we give a brief introduction to algebraic ZX-calculus by showing its generators, standard interpretation and rewriting rules. More details about algebraic ZX-calculus can be found in [14] and [15]. First we give the generators of algebraic ZX-calculus as follows:

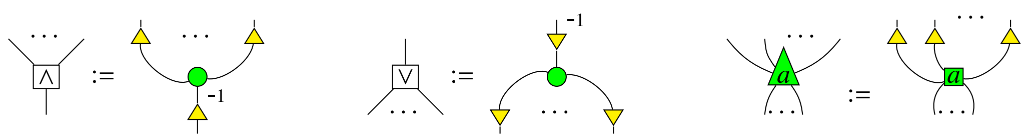
$R_{Z,a}^{(n,m)} : n \to m$	$\mathbb{I} : 1 \to 1$
$H : 1 \to 1$	$\sigma : 2 \to 2$
$C_a : 0 \to 2$	$C_u : 2 \to 0$
$T : 1 \to 1$	$T^{-1} : 1 \to 1$

Table 1: Generators of algebraic ZX-calculus, where $m, n \in \mathbb{N}$, $a \in \mathbb{C}$.

For simplicity, we make the following conventions:

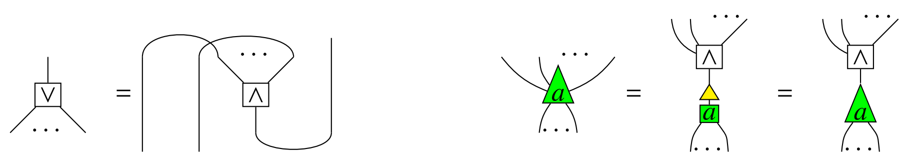


(H)



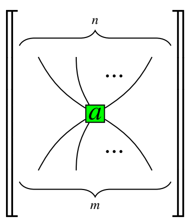
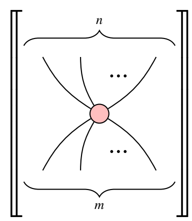
(1)

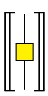
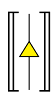
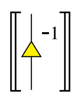
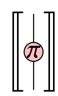
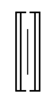
where $\alpha \in \mathbb{R}, \tau \in \{0, \pi\}$. As a consequence,

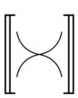
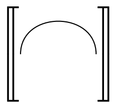
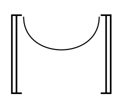
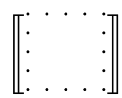


(2)

There is a standard interpretation $[\![\cdot]\!]$ for the ZX diagrams:

 $= |0\rangle^{\otimes m}\langle 0|^{\otimes n} + a|1\rangle^{\otimes m}\langle 1|^{\otimes n}$,  $= \sum_{\substack{0\leq i_1,\cdots,i_m,j_1,\cdots,j_n\leq 1\\ i_1+\cdots+i_m\equiv j_1+\cdots+j_n (mod\ 2)}} |i_1,\cdots,i_m\rangle\langle j_1,\cdots,j_n|$,

 $= \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$,  $= \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$,  $= \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}$,  $= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$,  $= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$,

 $= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$,  $= \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$,  $= \begin{pmatrix} 1 & 0 & 0 & 1 \end{pmatrix}$,  $= 1$,

$$[\![D_1 \otimes D_2]\!] = [\![D_1]\!] \otimes [\![D_2]\!], \quad [\![D_1 \circ D_2]\!] = [\![D_1]\!] \circ [\![D_2]\!],$$

where

$$a \in \mathbb{C}, \quad |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \langle 0| = \begin{pmatrix} 1 & 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad \langle 1| = \begin{pmatrix} 0 & 1 \end{pmatrix}.$$

Remark 2.1 *For convenience, we often ignore the symbol of interpretation* $[\![]\!]$ *and equalise matrices and diagrams directly.*

Next we present a set of rewriting rules for algebraic ZX-calculus [15].

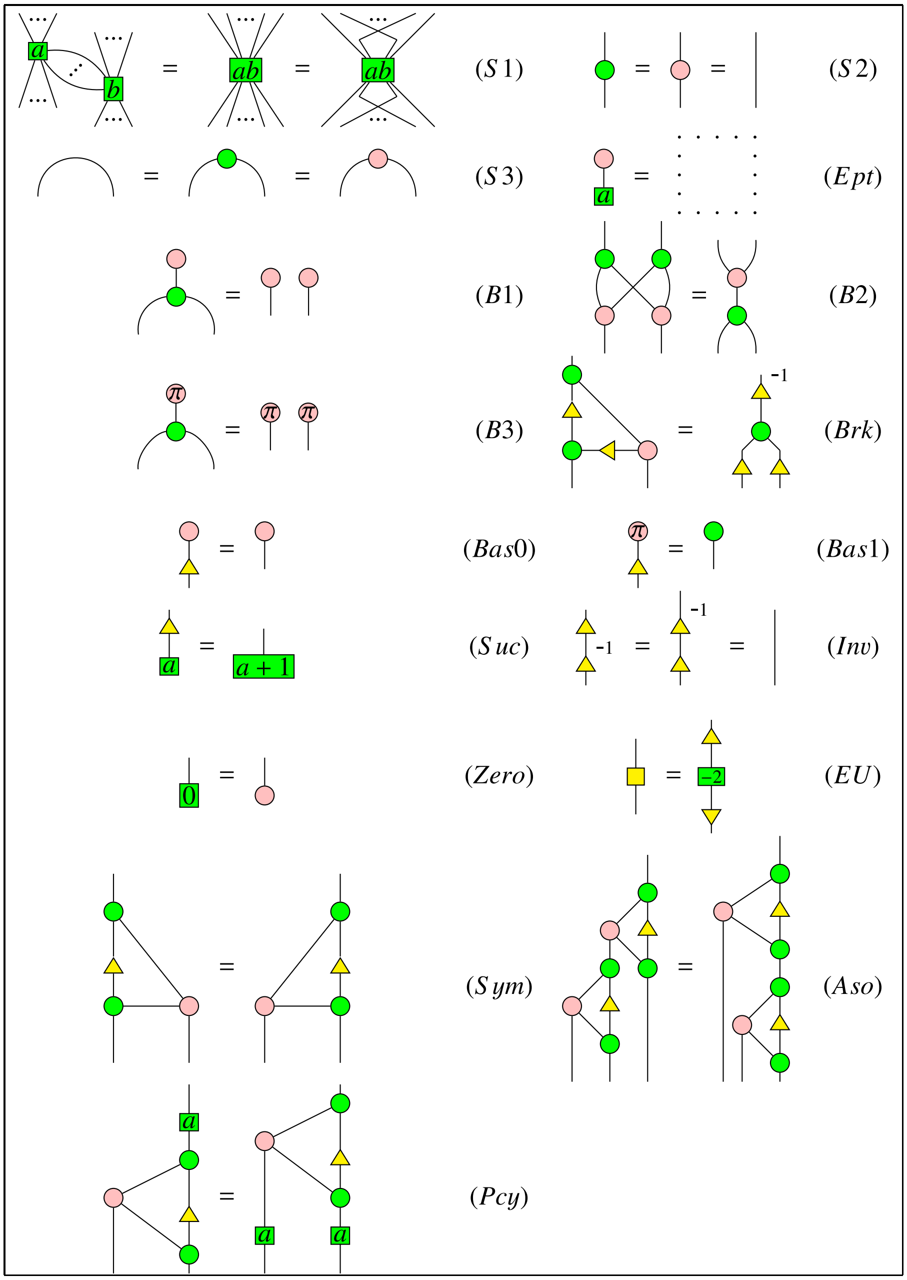


Figure 1: Algebraic rules, $a, b \in \mathbb{C}$. The upside-down flipped versions of the rules are assumed to hold as well.

3 Elementary matrices in ZX diagrams

In this section, we show how to represent elementary matrices of size $2^m \times 2^m$ in algebraic ZX-calculus. Elementary matrices correspond to elementary operations on matrices: left multiplication by an elementary matrix stands for elementary row operations, while right multiplication stands for elementary column operations.

There are three types of elementary matrices, the first type performs the row (column) multiplication:

$$R_{i\times(a)} = C_{i\times(a)} = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & & & \vdots \\ 0 & \cdots & a & \cdots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 1 \end{pmatrix} \begin{matrix} r_0 \\ \\ r_i \\ \\ r_{2^m-1} \end{matrix}$$

The second type performs the row (column) addition:

$$R_{i\times(a)+j} = C_{j\times(a)+i} = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & & & & & \vdots \\ 0 & \cdots & 1 & \cdots & 0 & \cdots & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & \cdots & a & \cdots & 1 & \cdots & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix} \begin{matrix} r_0 \\ \\ r_i \\ \\ r_j \\ \\ r_{2^m-1} \end{matrix}$$

The third type performs the row (column) switching:

$$R_{i\leftrightarrow j} = C_{i\leftrightarrow j} = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & & & & & \vdots \\ 0 & \cdots & 0 & \cdots & 1 & \cdots & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & \cdots & 1 & \cdots & 0 & \cdots & 0 \\ \vdots & & & \ddots & & & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix} \begin{matrix} r_0 \\ \\ r_i \\ \\ r_j \\ \\ r_{2^m-1} \end{matrix}$$

Note that we count the rows from 0 to $2^m - 1$.

Below we give the representation of the three kind of elementary matrices in Theorems 3.1, 3.4, and 3.7. Suppose $\{e_k \mid 0 \leq k \leq 2^m - 1\}$ are the 2^m-dimensional standard unit column vectors (with entries all 0s except for one 1):

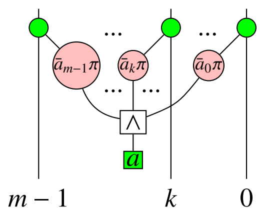
$$e_k = \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} r_0 \\ \\ r_k \\ \\ r_{2^m-1} \end{matrix}$$

where r_i denotes the i-th row, $0 \le i \le 2^m - 1, m \ge 1$. Then

$$|a_{m-1} \cdots a_i \cdots a_0\rangle = e_{\sum_{i=0}^{m-1} a_i 2^i}, \tag{3}$$

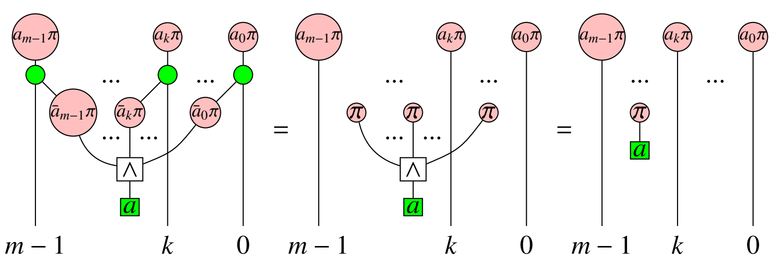
where $a_i \in \{0, 1\}, 0 \le i \le m - 1$ [9].

Theorem 3.1 *(Row multiplication) For any $2^m \times 2^m$ matrix, the i-th $(0 \le i \le 2^{m-1})$ row (column) multiplied by any number a (including 0) can be represented as*

$R_{i\times(a)} =$ 

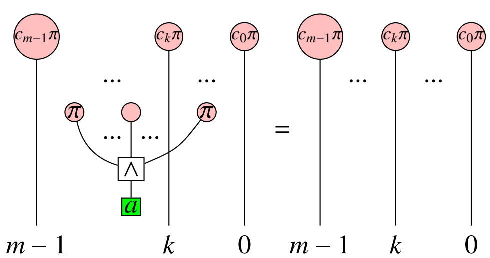
where $i = \sum_{k=0}^{m-1} a_k 2^k, a_k \in \{0, 1\}, \bar{a}_k = a_k \oplus 1, \oplus$ is the modulo 2 addition.

Proof: Note that $a_k \oplus \bar{a}_k = 1$. Then we have



$= ae_{\sum_{k=0}^{m-1} a_k 2^k}$

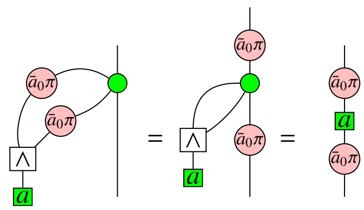
For any other inputting state $c_{m-1}\pi, \cdots c_k\pi, \cdots, c_0\pi$, there must be a j such that $a_j \neq c_j$, i.e., $c_j = \bar{a}_j$, therefore one get



$= e_{\sum_{k=0}^{m-1} c_k 2^k}$

□

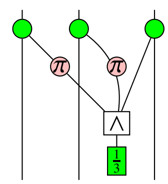
Remark 3.2 *In particular, if $m = 1$, then the diagram becomes*



Example 3.3 *Let*

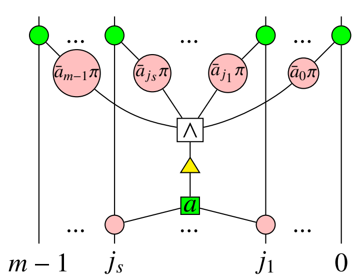
$$R_{1\times(\frac{1}{3})} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Here $m = 3, i = 1 = 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$, so $\bar{a}_2 = 1, \bar{a}_1 = 1, \bar{a}_0 = 0$. Therefore, $R_{1\times(\frac{1}{3})} =$

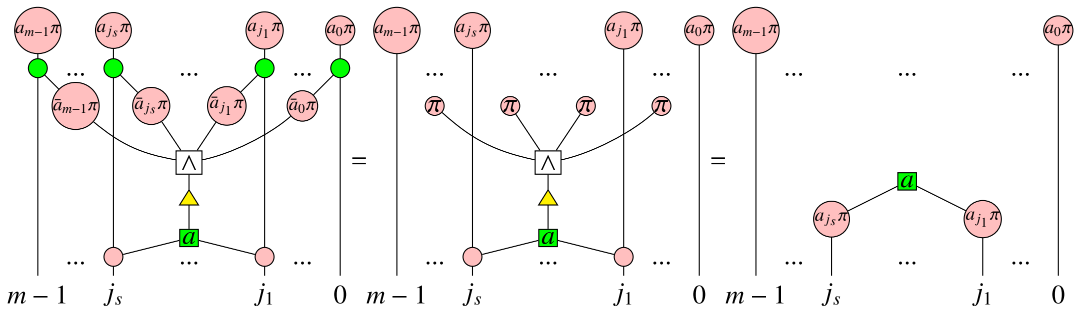


Theorem 3.4 *(Row addition) Suppose $i = a_{m-1}2^{m-1} + \cdots + a_{j_s}2^{j_s} + \cdots + a_{j_1}2^{j_1} + \cdots + a_0 2^0, a_k \in \{0, 1\}, \bar{a}_k = a_k \oplus 1, \oplus$ is the modulo 2 addition, $j = a_{m-1}2^{m-1} + \cdots + \bar{a}_{j_s}2^{j_s} + \cdots + \bar{a}_{j_1}2^{j_1} + \cdots + a_0 2^0$, $0 \leq j_1 < \cdots < j_s \leq m-1, 1 \leq s \leq m$. In other words, $j_1, \cdots, j_s$ are exactly the bit positions where the numbers i and j differ. Then*

$R_{i\times(a)+j} =$

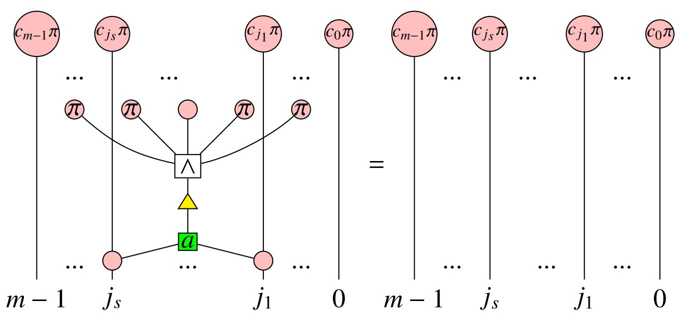


Proof: The i-th column of $R_{i\times(a)+j}$ should be $e_i + ae_j$, and the other columns are just of the form e_k. First for the i-th column we have



$$= |\underset{m-1}{a_{m-1}} \cdots \underset{j_s}{a_{j_s}} \cdots \underset{j_1}{a_{j_1}} \cdots \underset{0}{a_0}\rangle + a|\underset{m-1}{a_{m-1}} \cdots \underset{j_s}{\bar{a}_{j_s}} \cdots \underset{j_1}{\bar{a}_{j_1}} \cdots \underset{0}{a_0}\rangle = e_i + ae_j$$

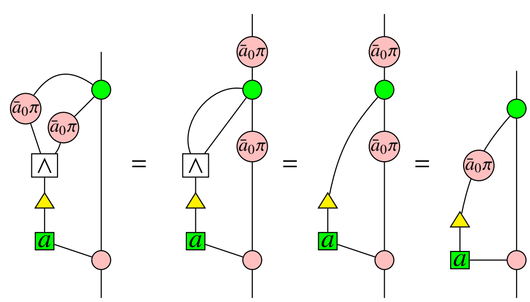
For any other inputting state $c_{m-1}\pi, \cdots c_k\pi, \cdots, c_0\pi$, there must be a j such that $a_j \neq c_j$, i.e., $c_j = \bar{a}_j$, therefore one get



$$= e_{\sum_{k=0}^{m-1} c_k 2^k}$$

□

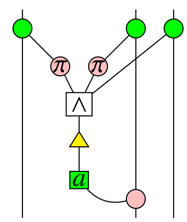
Remark 3.5 *In particular, if $m = 1$, then the diagram becomes*



Example 3.6 *Let*

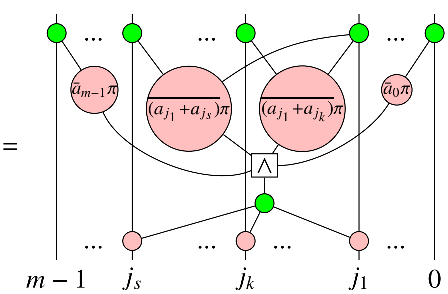
$$R_{1\times(2)+3} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Here $m = 3, i = 1 = 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0, j = 3 = 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$, *so* $\bar{a}_2 = 1, \bar{a}_1 = 1, \bar{a}_0 = 0, j_1 = 1, s = 1$. *Therefore,* $R_{1\times(2)+3} =$



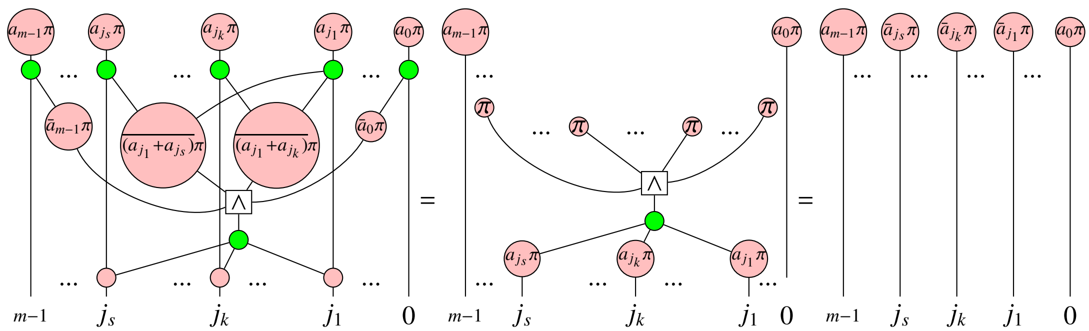
Theorem 3.7 *(Row switching) Suppose* $m \geq 2, i = a_{m-1}2^{m-1} + \cdots + a_{j_s}2^{j_s} + \cdots + a_{j_k}2^{j_k} + \cdots + a_{j_1}2^{j_1} + \cdots + a_0 2^0, a_k \in \{0, 1\}, \bar{a}_k = a_k \oplus 1, \oplus$ *is the modulo 2 addition,* $j = a_{m-1}2^{m-1} + \cdots + \bar{a}_{j_s}2^{j_s} + \cdots + \bar{a}_{j_k}2^{j_k} + \cdots + \bar{a}_{j_1}2^{j_1} + \cdots + a_0 2^0$, $0 \leq j_1 < \cdots < j_k < \cdots < j_s \leq m-1, 1 \leq s \leq m$. *In other words,* $j_1, \cdots, j_s$ *are exactly the bit positions where the numbers* i *and* j *differ. Then*

$R_{i\leftrightarrow j} =$

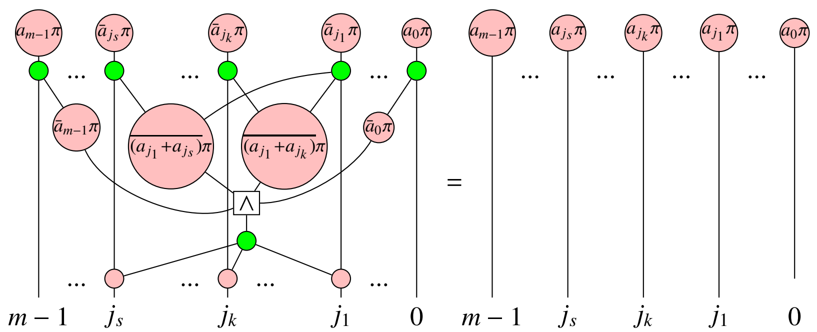


Note: *There are wires between the red spiders labelled* $\overline{(a_{j_1} + a_{j_x})}\pi$ *and the green spider at wire* j_1.

Proof: The state $|a_{m-1} \cdots a_{j_s} \cdots a_{j_k} \cdots a_{j_1} \cdots a_0\rangle$ will be sent to $|a_{m-1} \cdots \bar{a}_{j_s} \cdots \bar{a}_{j_k} \cdots \bar{a}_{j_1} \cdots a_0\rangle$ by $R_{i\leftrightarrow j}$ and vice versa, while the other states are remain unchanged under the action of $R_{i\leftrightarrow j}$. First for the i-th column we have

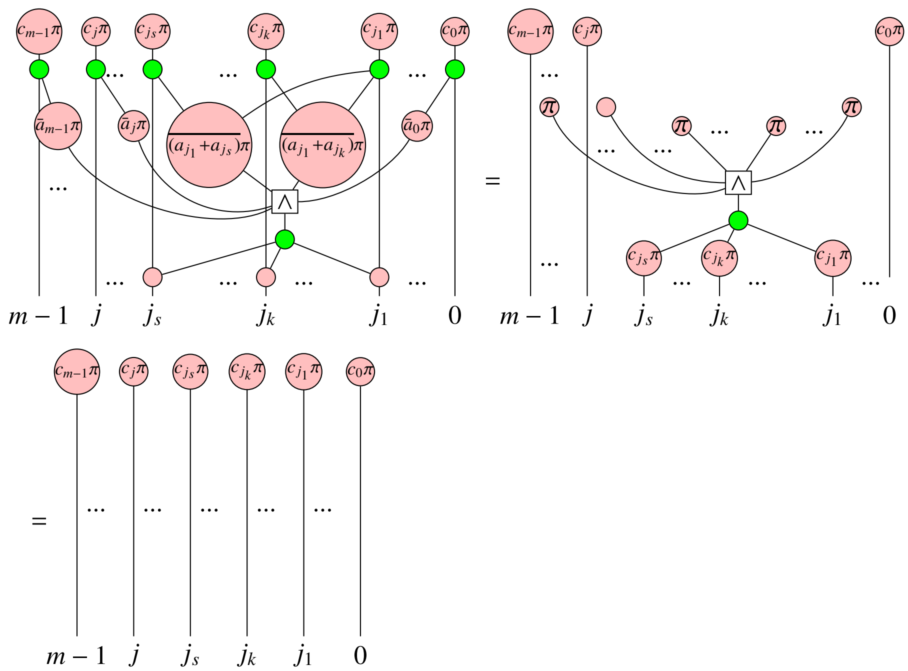


Similarly, we have

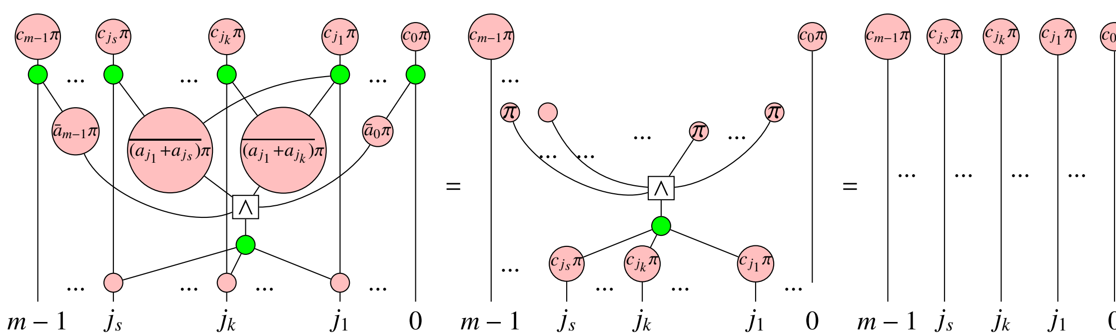


For any other input state $c_{m-1}\pi, \cdots c_k\pi, \cdots, c_0\pi$ (corresponding to u), if u is different from i and j in some place that beyond the set $\{j_1, \cdots, j_s\}$, then there must be a j

such that $a_j \neq c_j$, i.e., $c_j = \bar{a}_j$, therefore one get

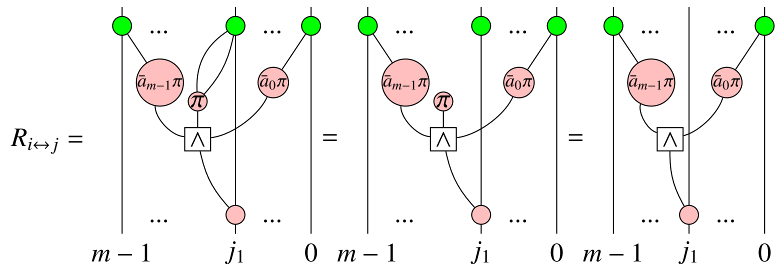


If u is different from i and j only in some places that belong to the set $\{j_1, \cdots, j_s\}$, say j_s and j_k (since i and j are different in these places, the number of places where u is different from i and j must be at least 2). If $c_{j_k} = a_{j_k}$, then $c_{j_s} \neq a_{j_s}$ (otherwise u won't be different from i at places j_s and j_k), i.e., $c_{j_s} = \bar{a}_{j_s}$. Therefore, $c_{j_k} + c_{j_s} = a_{j_k} + \bar{a}_{j_s} \neq a_{j_k} + a_{j_s}$. Similarly, If $c_{j_k} = \bar{a}_{j_k}$, then we still have $c_{j_k} + c_{j_s} \neq a_{j_k} + a_{j_s}$. Now we claim that it is impossible that both $c_{j_k} + c_{j_1} = a_{j_k} + a_{j_1}$ and $c_{j_s} + c_{j_1} = a_{j_s} + a_{j_1}$ hold. Otherwise, we could sum up (modulo 2) these two equalities and then get $c_{j_k} + c_{j_s} = a_{j_k} + a_{j_s}$ which is a contradiction. Hence either $c_{j_k} + c_{j_1} = \overline{a_{j_k} + a_{j_1}}$ or $c_{j_s} + c_{j_1} = \overline{a_{j_s} + a_{j_1}}$. Assume $c_{j_s} + c_{j_1} = \overline{a_{j_s} + a_{j_1}}$, then we have

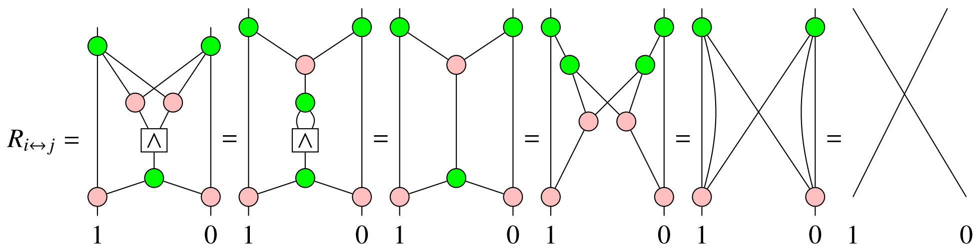


□

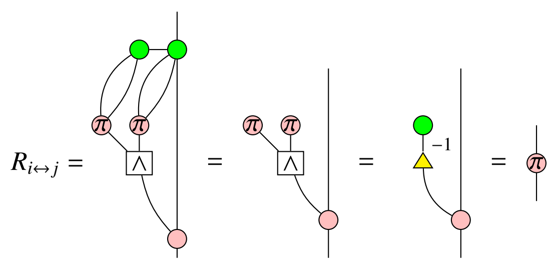
Remark 3.8 *Note that for the special case when $s = 1$, we have $a_{j_1} = a_{j_s}$, therefore $\overline{(a_{j_1} + a_{j_s})}\pi = \pi$. Then*



Also for the special case when $m = s = 2$, we have $a_{j_k} = a_{j_s}, a_0 = \bar{a}_1$, therefore $\overline{(a_1 + a_0)}\pi = 0$. Then



If $m = 1$, then $s = 1$ since $i \neq j$, therefore

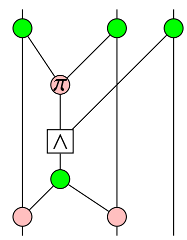


Example 3.9 *Let*

$$R_{1\leftrightarrow 7} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Here $m = 3, i = 1 = 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0, j = 7 = 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$, so

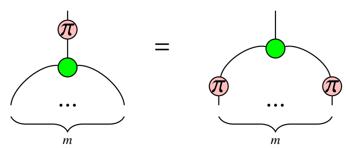
$\bar{a}_2 = 1, \bar{a}_1 = 1, \bar{a}_0 = 0$, $j_1 = 1$, $j_2 = 2$, $s = 2$. *Therefore,* $R_{1\leftrightarrow 7} =$



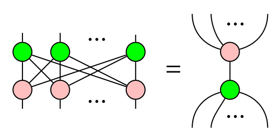
4 Properties of diagrammatic elementary matrices

In this section, we show by diagrams some properties of elementary matrices on their inverses and transpose.

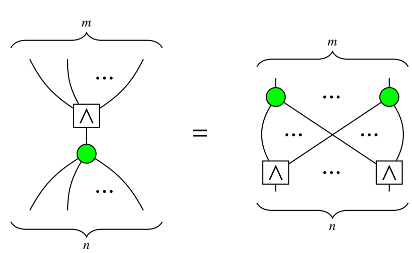
Lemma 4.1 *[15]*



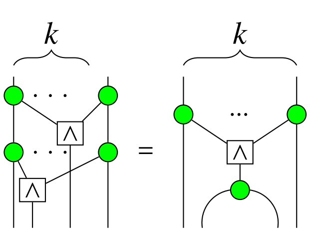
Lemma 4.2 *[6]*



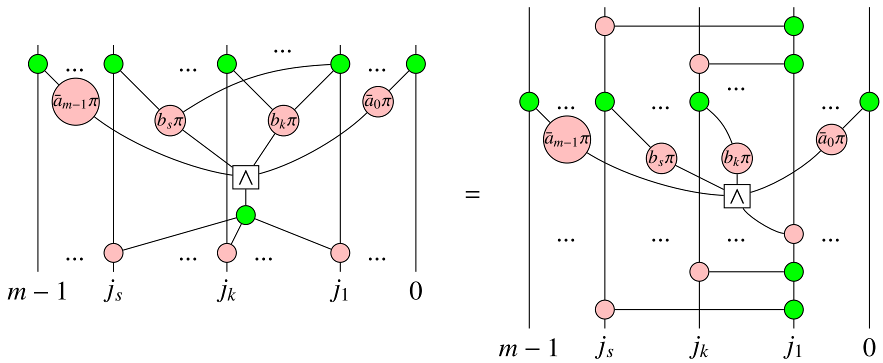
Lemma 4.3 *[15]*



Lemma 4.4 *[10] For any* $k \geq 0$, *we have*

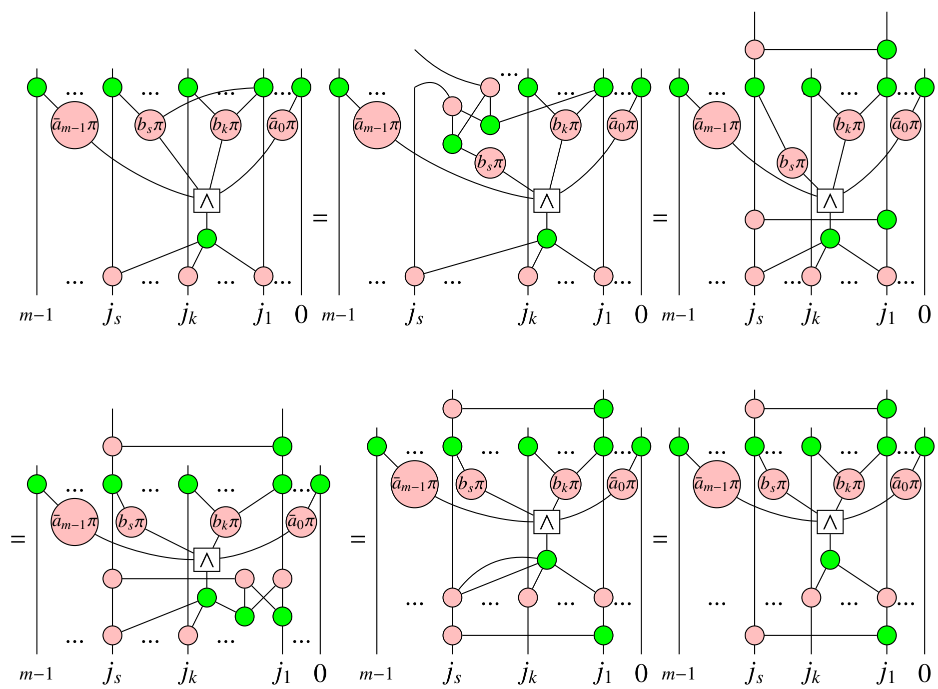


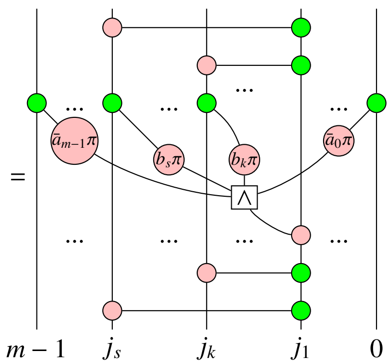
Lemma 4.5

$R_{i\leftrightarrow j} =$ 

where $b_s = \overline{(a_{j_1} + a_{j_s})}, \cdots, b_k = \overline{(a_{j_1} + a_{j_k})}$.

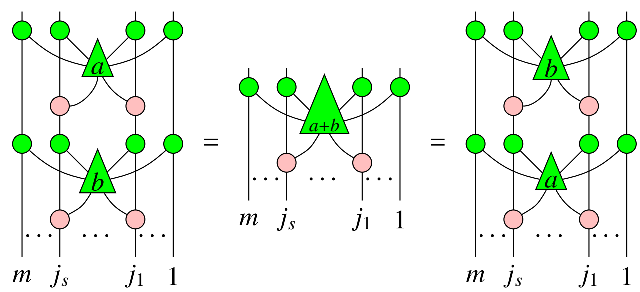
Proof:



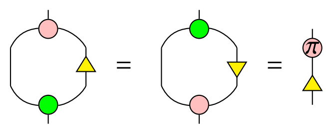


where the last equality is obtained using the same method as previous steps. □

Lemma 4.6 *[15]*

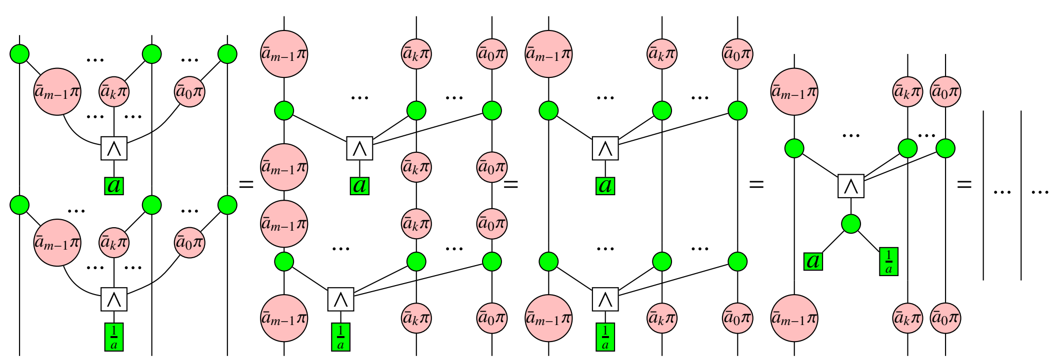


Lemma 4.7 *[15]*



Proposition 4.8 *Suppose $a \neq 0, a \in \mathbb{C}$. Then $R^{-1}_{i\times(a)} = R_{i\times(\frac{1}{a})}$, i.e., $R_{i\times(a)}R_{i\times(\frac{1}{a})} = R_{i\times(\frac{1}{a})}R_{i\times(a)} = I$.*

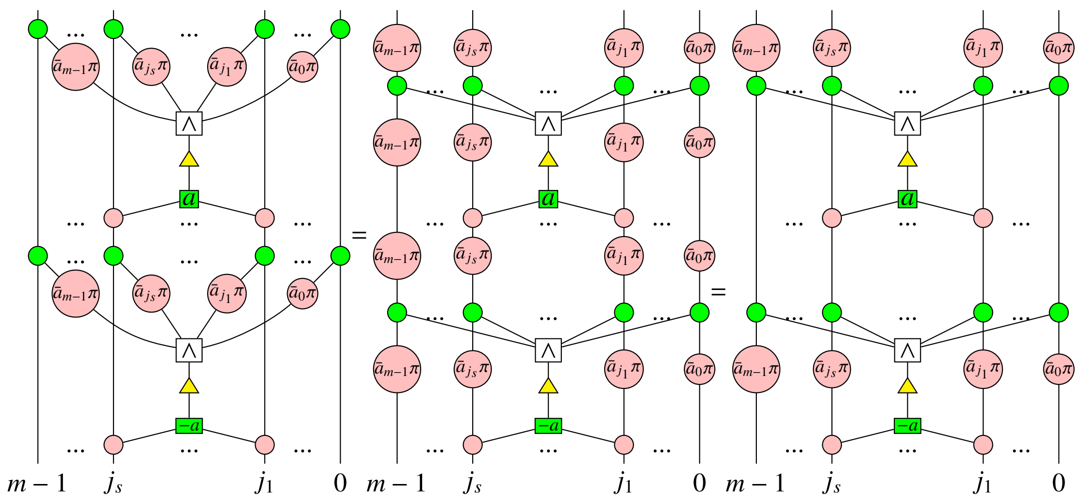
Proof: $R_{i\times(\frac{1}{a})}R_{i\times(a)} =$

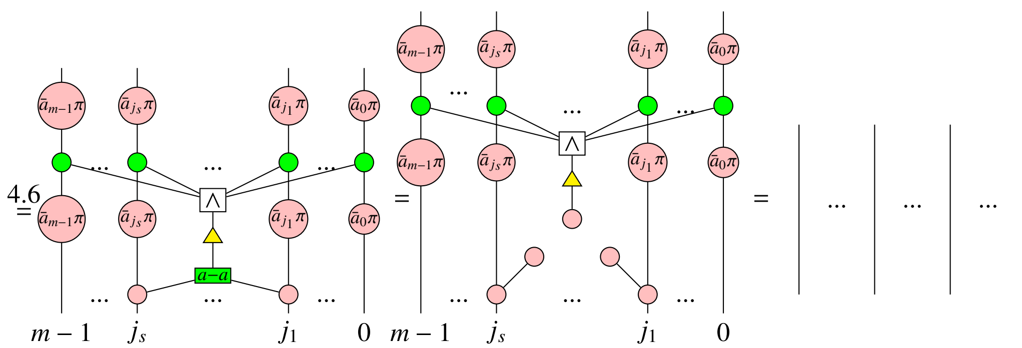


$R_{i\times(a)}R_{i\times(\frac{1}{a})} = I$ can be proved similarly. □

Proposition 4.9 $R^{-1}_{i\times(a)+j} = R_{i\times(-a)+j}$, *i.e.*, $R_{i\times(-a)+j}R_{i\times(a)+j} = R_{i\times(a)+j}R_{i\times(-a)+j} = I$.

Proof: $R_{i\times(-a)+j}R_{i\times(a)+j} =$

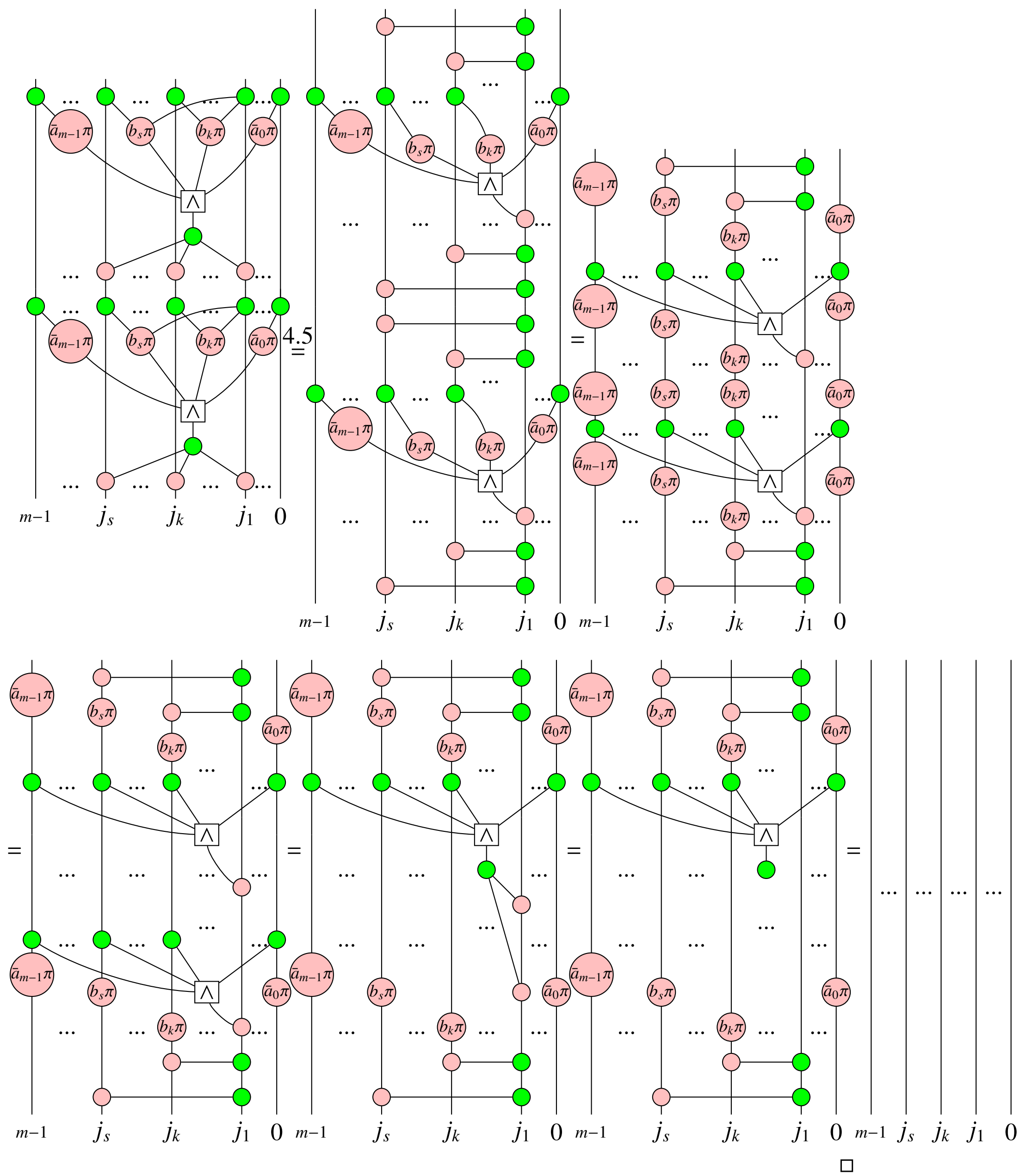




$R_{i\times(a)+j}R_{i\times(-a)+j} = I$ can be proved similarly. □

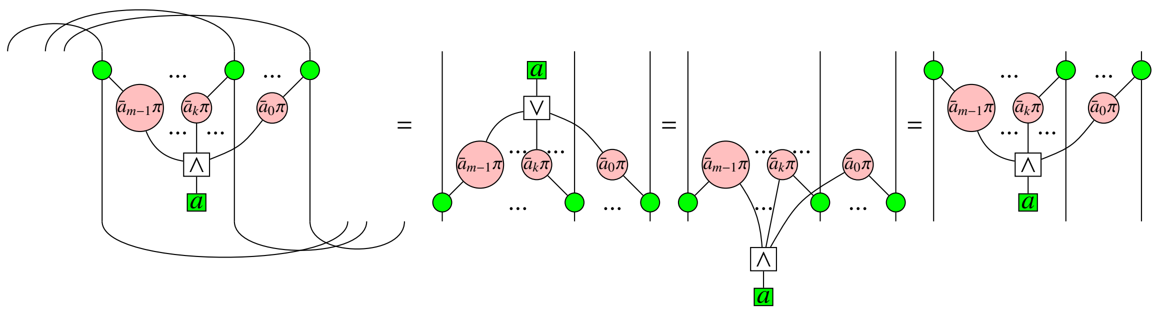
Proposition 4.10 $R^{-1}_{i\leftrightarrow j} = R_{i\leftrightarrow j}$, *i.e.*, $R_{i\leftrightarrow j}R_{i\leftrightarrow j} = I$.

Proof: $R_{i\leftrightarrow j}R_{i\leftrightarrow j} =$



Proposition 4.11 $R^T_{i\times(a)} = R_{i\times(a)}$.

Proof: $R^T_{i\times(a)} =$

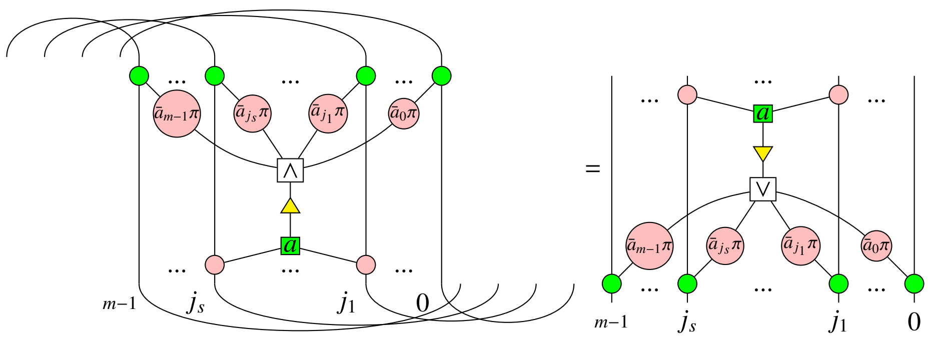


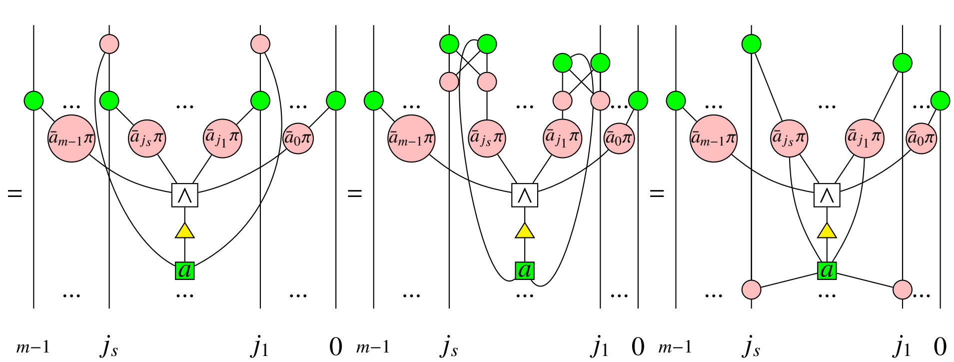
$= R_{i\times(a)}$

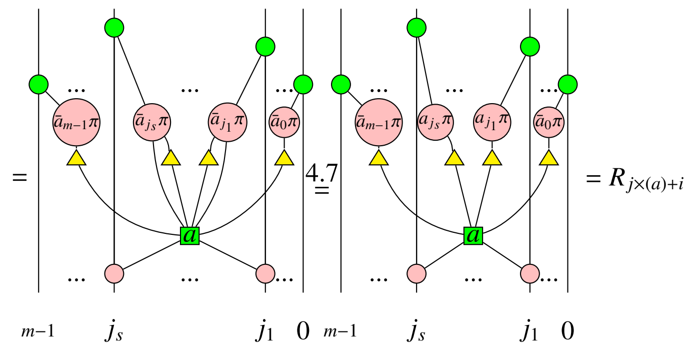
□

Proposition 4.12 $R^T_{i\times(a)+j} = R_{j\times(a)+i}$.

Proof: Note that $i = a_{m-1}2^{m-1} + \cdots + a_{j_s}2^{j_s} + \cdots + a_{j_1}2^{j_1} + \cdots + a_0 2^0, a_k \in \{0,1\}, \bar{a}_k = a_k \oplus 1, \oplus$ is the modulo 2 addition, $j = a_{m-1}2^{m-1} + \cdots + \bar{a}_{j_s}2^{j_s} + \cdots + \bar{a}_{j_1}2^{j_1} + \cdots + a_0 2^0$, $0 \le j_1 < \cdots < j_s \le m-1, 1 \le s \le m$, so j is different from i exactly in the $j_1, \cdots, j_s$ places in their binary expansions. Then $R^T_{i\times(a)+j} =$



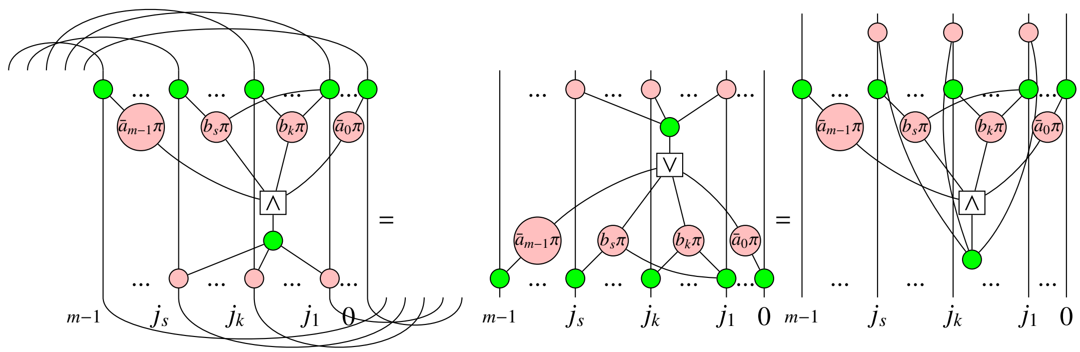


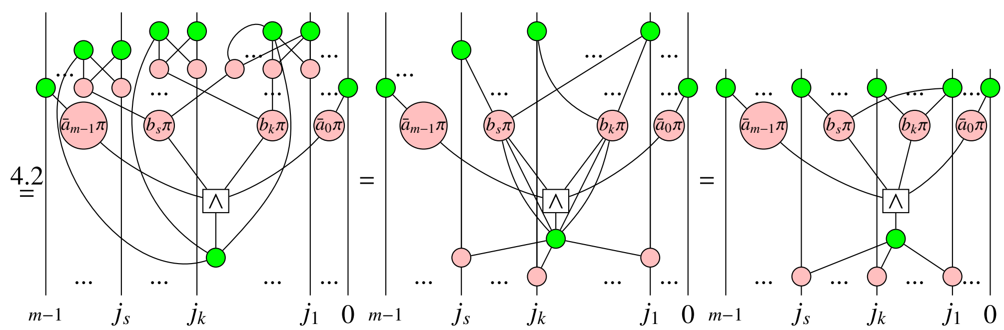


∎

Proposition 4.13 $R^T_{i\leftrightarrow j} = R_{i\leftrightarrow j}$.

Proof: Note that $i = a_{m-1}2^{m-1} + \cdots + a_{j_s}2^{j_s} + \cdots + a_{j_1}2^{j_1} + \cdots + a_0 2^0, a_k \in \{0, 1\}, \bar{a}_k = a_k \oplus 1, \oplus$ is the modulo 2 addition, $j = a_{m-1}2^{m-1} + \cdots + \bar{a}_{j_s}2^{j_s} + \cdots + \bar{a}_{j_1}2^{j_1} + \cdots + a_0 2^0$, $0 \le j_1 < \cdots < j_s \le m-1, 1 \le s \le m$, so j is different from i exactly in the $j_1, \cdots, j_s$ places in their binary expansions. Then $R^T_{i\leftrightarrow j} =$





$= R_{i\leftrightarrow j}$

where $b_s = \overline{(a_{j_1} + a_{j_s})}, \cdots, b_k = \overline{(a_{j_1} + a_{j_k})}$. □

Proposition 4.14 $R_{i\times(a)}R_{i\times(b)} = R_{i\times(ab)}$.

This follows directly from Lemma 4.1, Lemma 4.4 and the rule (S1).

Proposition 4.15 $R_{i\times(a)+j}R_{i\times(b)+j} = R_{i\times(a+b)+j}$.

This follows directly from Lemma 4.1 and Lemma 4.6.

5 Represent matrices by string diagrams

In this section we show how to represent an arbitrary matrix A of size $2^m \times 2^n$ as ZX diagrams. The idea is to use elementary transformations to turn A into a simple matrix which can be easily represented in ZX, then we apply the inverse operations diagrammatically to get the diagram for A.

As is well known in linear algebra, any matrix A can be turned into a reduced row echelon form by means of a finite sequence of elementary row operations. If we further allow elementary column operations to be used, then A can be transformed to a standard form:

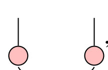
$$C = \begin{pmatrix} E_r & O \\ O & O \end{pmatrix}_{2^m\times 2^n}$$

where r is the rank of A, and E_r is an identity matrix of order r. Below we show that C can be represented as a ZX diagram.

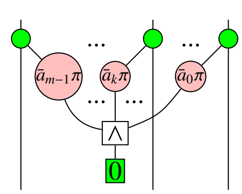
Proposition 5.1 *Any matrix A of size $2^m \times 2^n$ can be represented by a ZX diagram.*

Proof: If $m \le n$, then $C = \begin{pmatrix} K & O \end{pmatrix}_{2^m\times 2^n} = (\underbrace{10\cdots0}_{2^{n-m}}) \otimes K$, where the O in C is a zero matrix of size $2^m \times (2^n - 2^m)$,

$$K = \begin{pmatrix} E_r & \cdots & O \\ \vdots & \ddots & \vdots \\ O & \cdots & O \end{pmatrix}_{2^m\times 2^m}$$

Since $(\underbrace{10\cdots0}_{2^{n-m}}) =$ 

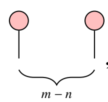
, and K can be represented by sequential composition of $2^m - r$ row multiplication elementary matrices (multiplying 0) of size $2^m \times 2^m$ whose diagrammatic representation is shown in Theorem 3.1 as



Therefore, C can now be represented by string diagrams.

If $m > n$, then $C = \begin{pmatrix} K' \\ O \end{pmatrix}_{2^m \times 2^n} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}_{2^{m-n} \times 1} \otimes K'$, where the O in C is a zero matrix of size $(2^m - 2^n) \times 2^n$,

$$K' = \begin{pmatrix} E_r & \cdots & O \\ \vdots & \ddots & \vdots \\ O & \cdots & O \end{pmatrix}_{2^n \times 2^n}$$

Since $\begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}_{2^{m-n} \times 1} =$ 

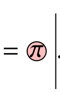
, and K' can be represented by sequential composition of $2^n - r$ row multiplication (by 0) elementary matrices of size $2^n \times 2^n$ whose diagrammatic representation is shown in Theorem 3.1, C can now be represented by diagrams.

To summarise, C can always be represented by a ZX diagram, also each elementary matrix can be represented by a ZX diagram, therefore, if we reverse the procedures from A to C, we then get the diagrammatic representation of A. □

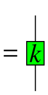
Remark 5.2 *In practice, it is not necessary to go from the beginning to the last step (standard form). You can stop at anywhere you know how to represent the corresponding matrix in ZX diagrams.*

Example 5.3 *Given an arbitrary 2×2 matrix X, let*

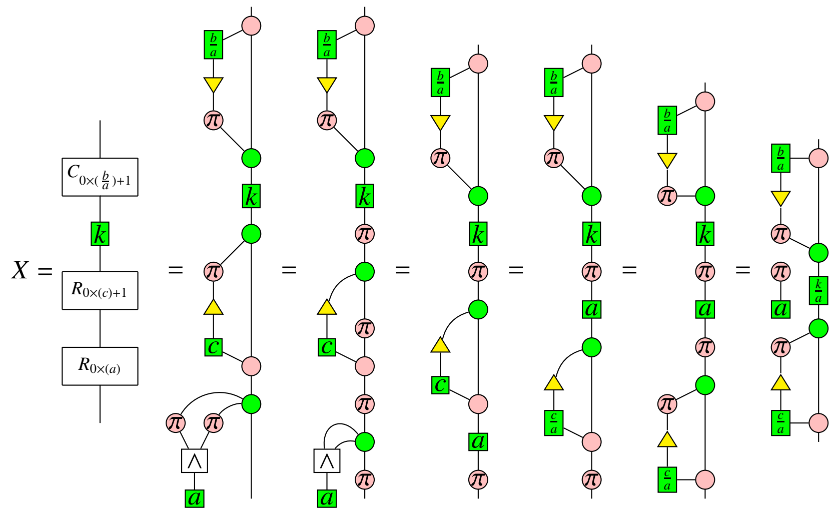
$$X = \begin{pmatrix} a & b \\ c & d \end{pmatrix}.$$

If X is a zero matrix, then $X =$ 

. Otherwise, we assume $a \neq 0$. Then

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \xrightarrow{R_{0\times(\frac{1}{a})}} \begin{pmatrix} 1 & \frac{b}{a} \\ c & d \end{pmatrix} \xrightarrow{R_{0\times(-c)+1}} \begin{pmatrix} 1 & \frac{b}{a} \\ 0 & d - \frac{bc}{a} \end{pmatrix} \xrightarrow{C_{0\times(-\frac{b}{a})+1}} \begin{pmatrix} 1 & 0 \\ 0 & d - \frac{bc}{a} \end{pmatrix} =$$ 

where $k = d - \frac{bc}{a}$. Reverse the procedure, considering $i = 0 \times 2^0, a_0 = 0, j = 1 \times 2^0$, then we obtain the diagram for the matrix X:



6 Further work

In this paper, we first show how to represent elementary matrices of size $2^m \times 2^m$ in ZX diagrams. Based on that, we depict arbitrary matrix of size $2^m \times 2^n$ via algebraic ZX-calculus.

Next, we would like to represent some useful matrix technologies like singular value decomposition (SVD), QR decomposition and lower–upper (LU) decomposition using algebraic ZX-calculus, which we hope could pave the way towards visualising important matrix technologies deployed in machine learning.

Although there is a normal form for arbitrary finite dimensional vectors [17], which means any finite matrix can be represented in higher dimensional ZX diagrams via map-state duality [4], we still look for a simple representation for all kinds of elementary matrices of arbitrary size (some kinds of elementary matrices of size $d^m \times d^m$ for any positive integer d have been given in [17]), so that any finite matrix can be represented in ZX diagrams in a similar way as shown in this paper.

Acknowledgements

The author greatly thanks Richie Yeung for helpful discussions and careful reading of the first version of this paper with valuable comments. He also thanks Konstantinos Mei and Bob Coecke for insightful comments. The author would acknowledge the grant FQXi-RFP-CPW-2018 as well.

References

[1] Niel de Beaudrap, Xiaoning Bian & Quanlong Wang (2020): *Fast and Effective Techniques for T-Count Reduction via Spider Nest Identities*. In Steven T. Flammia, editor: *15th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2020)*, *Leibniz International Proceedings in Informatics (LIPIcs)* 158, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany, pp. 11:1–11:23, doi:10.4230/LIPIcs.TQC.2020.11.

[2] Bob Coecke & Ross Duncan (2011): *Interacting quantum observables: categorical algebra and diagrammatics*. *New Journal of Physics* 13(4), p. 043016. Available at `http://stacks.iop.org/1367-2630/13/i=4/a=043016`. doi:10.1088/1367-2630/13/4/043016.

[3] Bob Coecke, Dominic Horsman, Aleks Kissinger & Quanlong Wang (2021): *Kindergarden quantum mechanics graduates ...or how I learned to stop gluing LEGO together and love the ZX-calculus*. *Theoretical Computer Science*, doi:https://doi.org/10.1016/j.tcs.2021.07.024. Available at `https://www.sciencedirect.com/science/article/pii/S0304397521004308`.

[4] Bob Coecke & Aleks Kissinger (2017): *Picturing quantum processes*. Cambridge University Press.

[5] Alexander Cowtan, Silas Dilkes, Ross Duncan, Will Simmons & Seyon Sivarajah (2020): *Phase Gadget Synthesis for Shallow Circuits*. *Electronic Proceedings in Theoretical Computer Science* 318, pp. 213–228, doi:10.4204/eptcs.318.13.

[6] Ross Duncan & Simon Perdrix (2009): *Graph States and the Necessity of Euler Decomposition*. *Mathematical Theory and Computational Practice* 5635, pp. 167–177. doi: 10.1007/978-3-642-03073-4_18.

[7] Aleks Kissinger & Arianne Meijer-van de Griend (2020): *CNOT circuit extraction for topologically-constrained quantum memories*. *Quantum Inf. Comput.* 20(7&8), pp. 581–596.

[8] David C Lay, Steven R Lay & Judith McDonald (2016): *Linear algebra and its applications*. Pearson.

[9] N. David Mermin (2007): *Quantum Computer Science: An Introduction*. Cambridge University Press.

[10] Anthony Munson, Bob Coecke & Quanlong Wang (2020): *AND-gates in ZX-calculus: spider nest identities and QBC-completeness*. *Proceedings of the 17th International Conference on Quantum Physics and Logic (QPL) 2020*. arXiv:1910.06818.

[11] Kevin P. Murphy (2012): *Machine Learning: A Probabilistic Perspective*. The MIT Press.

[12] Michael A. Nielsen & Isaac L. Chuang (2010): *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, doi:10.1017/CBO9780511976667. doi: 10.1017/CBO9780511976667.

[13] Seyon Sivarajah, Silas Dilkes, Alexander Cowtan, Will Simmons, Alec Edgington & Ross Duncan (2020): *t|ket⟩: a retargetable compiler for NISQ devices*. Quantum Science and Technology 6(1), p. 014003, doi:10.1088/2058-9565/ab8e92. Available at `https://doi.org/10.1088/2058-9565/ab8e92`.

[14] Quanlong Wang (2020): *An algebraic axiomatisation of ZX-calculus*. Proceedings of the 17th International Conference on Quantum Physics and Logic (QPL) 2020. arXiv:1911.06752.

[15] Quanlong Wang (2020): *Algebraic complete axiomatisation of ZX-calculus with a normal form via elementary matrix operations*. arXiv:2007.13739.

[16] Quanlong Wang (2020): *Completeness of algebraic ZX-calculus over arbitrary commutative rings and semirings*. arXiv:1912.01003v3.

[17] Quanlong Wang (2021): *Qufinite ZX-calculus: a unified framework of qudit ZX-calculi*. arXiv:2104.06429.

[18] John van de Wetering (2020): *ZX-calculus for the working quantum computer scientist*. arXiv:2012.13966.