A Path Model for Quantum Skew-Symmetric Matrices

Karel Casteels, Jessica Liu, Aesha Siddiqui

Summer 2017

Abstract

In 2014, Casteels developed a combinatorial model for the algebra of quantum matrices, $\mathcal{O}_q(\mathcal{M}_{m\times n}(\mathbb{K}))$, and successfully used the model to develop new results in the theory of torus-invariant prime ideals in $\mathcal{O}_q(\mathcal{M}_{m\times n}(\mathbb{K}))$. We extend Casteels' work to $\mathcal{O}_q(Sk_n(\mathbb{K}))$, the quantized coordinate ring of $n \times n$ skew-symmetric matrices over an infinite field $\mathbb{K}$. It has been shown that these algebras have connections with the theory of totally nonnegative matrices and further applications, including quantum group theory, braided tensor categories, and knot theory.

Although $\mathcal{O}_q(\mathcal{M}_{m\times n}(\mathbb{K}))$ and $\mathcal{O}_q(Sk_n(\mathbb{K}))$ have different commutation relations, Casteels' method can be adapted to find an analogous model for the algebra of quantum skew-symmetric matrices. This model associates each generator of $\mathcal{O}_q(Sk_n(\mathbb{K}))$ to a sum of path weights in a directed grid graph, where weights are elements of the quantum skew-symmetric torus. We want to show that this model embeds the generating set of $\mathcal{O}_q(Sk_n(\mathbb{K}))$ in the algebra's corresponding torus by studying how the model might preserve the commutation relations of each space. These methods may lead to new insights on applying these paths models to other quantum algebras.

1 Introduction

The purpose of this paper is to extend Karel Casteels' combinatorial model for the algebra of quantum matrices. [1]. This work embeds the algebra of quantum matrices into the algebra's corresponding torus. Casteels' path model associates each generator of the algebra of quantum matrices as a sum of monomials constructed from generators of the quantum torus. This allows us to study quantum matrices via a simpler algebra with fewer commutation relations, and aided Casteels in proving that all prime ideals of quantum matrices are generated by quantum minors (i.e. minors of the matrix of generators).

Casteels' path model for quantum matrices first assigns each generator of the quantum affine space to a vertex in a $m \times n$ directed grid graph. The generators of a subalgebra of the torus are then associated to the sum over specific path weights, where weights are specific monomials composed of generators of the quantum torus. Casteels showed that this subalgebra is isomorphic to the algebra of quantum matrices by observing the commutation relations of the subalgebra. The two algebras have identical structures, hence the isomorphism.

Using these methods, we extend this work to the algebra of quantum skew-symmetric matrices. Like all quantum algebras, this algebra is non-commutative, but with some additional structure. Unlike their classical analogues, quantum skew-symmetric matrices are not a subset of quantum matrices. The quantization of each algebra is unique, such that the commutation relations on each generating set differs. Thus, extending this model involves taking into account these new commutation relations.

The path model for quantum skew-symmetric matrices is altered in few ways. Each generator of the algebra's corresponding affine space is associated to a vertex above the diagonal of a $n \times n$ grid graph. These vertices are reflected and negated across the diagonal, as in a skew-symmetric matrix. Similar to the original model, we associate each generator of a subalgebra of the quantum skew-symmetric torus to the sum of path weights. We conjecture that this algebra is isomorphic to the algebra of quantum skew-symmetric matrices. Similar to Casteels' approach, our strategy is to show that the subalgebra of the quantum torus generated by the path model has the same commutation relations as quantum skew-symmetric matrices.

In Section 2, we formally define the quantum algebras of interest, and provide further background on these spaces and their applications. In Section 3, we define terminology necessary to introduce our path model. In Section 4, we define an equivalence relation on paths. This relation allows us to write a program that computes generating sets for subalgebras of the quantum skew-symmetric torus. We provide instructions for using this program in Section 5. Continuation of this work and further open questions are discussed in Section 6.

2 Background

The algebra of quantum skew-symmetric matrices, more formally defined below, is a non-commutative algebra with additional structure. This structure is imposed by a complicated set of commutation relations that make the algebra difficult to study.

Definition 1. *Quantum skew-symmetric matrices* are the quantized coordinate ring of $n \times n$ skew-symmetric matrices, $\mathcal{O}_q(Sk_n(\mathbb{K}))$, is the $\mathbb{K}$-algebra generated by the elements

$$\{x_{(i,j)} | i < j\}$$

subject to the following relations,

$$\begin{aligned}
x_{(i,j)}x_{(i,l)} &= qx_{(i,j)}x_{(i,l)}, && \text{for } j < l; \\
x_{(i,j)}x_{(j,l)} &= qx_{(j,l)}x_{(i,l)}, && \text{for } j < l; \\
x_{(i,j)}x_{(k,j)} &= qx_{(k,j)}x_{(i,l)}, && \text{for } i < k; \\
x_{(i,j)}x_{(k,l)} &= qx_{(k,l)}x_{(i,l)}, && \text{for } j < l; \\
x_{(i,j)}x_{(k,l)} &= qx_{(k,l)}x_{(i,j)} + (q - q^{-1})x_{(i,l)}x_{(k,j)}, && \text{for } i < k < l < j; \\
x_{(i,j)}x_{(k,l)} &= qx_{(k,l)}x_{(i,j)} + (q - q^{-1})x_{(i,k)}x_{(j,l)} - q(q - q^{-1})x_{(i,l)}x_{(j,k)}, && \text{for } i < j < k < l.
\end{aligned}$$

We are interested in studying a related, but simpler, algebra, the quantum affine space. This space has a similar, yet smaller, set of commutation relations, making it more amenable to study.

Definition 2. The *quantum skew-symmetric affine space*, denoted as T_n, is the algebra generated by the elements of

$$\{t_{(i,j)} | i < j\}$$

subject to the following relations,

$$\begin{aligned}
t_{(i,j)}t_{(i,l)} &= qt_{(i,l)}t_{(i,j)}, && \text{for } j < l; \\
t_{(i,j)}t_{(j,l)} &= qt_{(j,l)}t_{(i,j)}, && \text{for } j < l; \\
t_{(i,j)}t_{(k,j)} &= qt_{(k,j)}t_{(i,j)}, && \text{for } i < k; \\
t_{(i,j)}t_{(k,l)} &= qt_{(k,l)}t_{(i,j)}, && \text{for } i < k < l < j; \\
t_{(i,j)}t_{(k,l)} &= qt_{(k,l)}t_{(i,j)}, && \text{for } i < k < j < l; \\
t_{(i,j)}t_{(k,l)} &= qt_{(k,l)}t_{(i,j)}, && \text{for } i < j < k < l.
\end{aligned}$$

We then define the quantum skew-symmetric torus to be an algebra whose generating set is comprised of the generating set of the affine space, along with the inverses of those generators.

Definition 3. The *quantum skew-symmetric torus*, denoted $T_n^{\times}$, is the algebra generated by $\{\hat{X}_{(i,j)}\}$, the multiplicative set generated by the standard generators $t_{(i,j)}$ of the quantum skew-symmetric affine space.

Note that the torus and the affine space have the same set of commutation relations.

3 The Path Model

We now define the path model for studying quantum skew-symmetric matrices. We first define a type of directed graph, then define uesful objects on this graph.

Definition 4. The $m \times n$ *skew symmetric grid* is the graph $G^{m \times n} = \{V, E\}$ with the following vertex and directed edge sets:

$$\begin{aligned} V &= \{(a,b) | a \in [m+1] \text{ and } b \in [n+1]\} - \{(m+1, n+1)\} \\ E_D &= \{((a,b),(a+1,b)) | a, a+1 \leq m \text{ and } b \leq n+1\} \\ E_L &= \{((a,b),(a,b-1) | a \leq m+1 \text{ and } b, b-1 \leq n\} \\ E &= E_D \bigcup E_L \end{aligned}$$

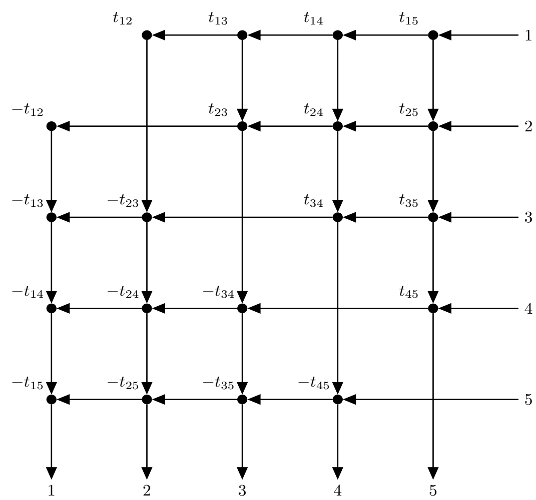

Figure 1: G^5, a 5×5 directed skew-symmetric grid graph

Definition 5. Let G^n be a directed skew-symmetric grid. A *path* p is a sequence of vertices $\{v_0, \dots, v_k\}$ such that for all $i \in [k]$ there exists a directed edge (v_{i-1}, v_i). We say that $p : v_0 \to v_k$ is a path p that begins at v_0 and ends at v_k. We let $P : v_0 \to v_k$ refer to the set of paths that begin at v_0 and end at v_k.

Definition 6. Let $p = (v_0 \dots v_k)$ be a path.

- We say that there exists a Γ-turn at vertex v_i if (v_{i-1}, v_i) is horizontal and (v_i, v_{i+1}) is vertical.
- We say that there exists a $\daleth$-turn at v_i if (v_{i-1}, v_i) is vertical and (v_i, v_{i+1}) is horizontal.

The *weight of a vertex* v of a path p, where $v = t_{i,j}$, is defined by the function

$$\begin{aligned} w(t_{(i,j)}) &= t_{(i,j)}, && \text{if } \exists\ \Gamma\text{-turn at } v; \\ w(t_{(i,j)}) &= t_{(i,j)}{}^{-1}, && \text{if } \exists\ \daleth\text{-turn at } v; \\ w(t_{(i,j)}) &= 1, && \text{otherwise.} \end{aligned}$$

Definition 7. The *weight of a path*, denoted $w(p)$, where $p = \{v_0 \dots v_k\}$, is $q^n w(v_0) \dots w(v_k)$, where

$$\begin{aligned} n &= 0 && \text{if there does not exist a horizontal edge across the diagonal,} \\ n &= 1 && \text{if there exists a horizontal edge across the diagonal.} \end{aligned}$$

Definition 8. The *lexicographic order* on $[m] \times [n]$ is the total order $<$ obtained by setting

$$(i, j) < (k, l) \iff i < k, \text{ or } i = k \text{ and } j < l.$$

Visually, $(i, j) < (k, l)$ if (i, j) is northwest to (k, l).

Example 1.

$$\hat{x}_{(1,1)} = \sum_{P:1\to 1} w(P) = -qt_{(1,3)}t_{(2,3)}^{-1}t_{(1,2)} + t_{(1,2)}t_{(2,3)}^{-1}t_{(1,3)}.$$

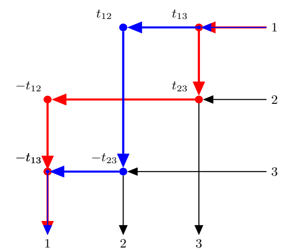

Figure 2: The set $P : 1 \to 1$ on G^3

Definition 9. The *transpose* of a path p, denoted p^T, is its reflection across the diagonal.

Definition 10. A path p is transpose-disjoint if $p \cap p^T = \emptyset$.

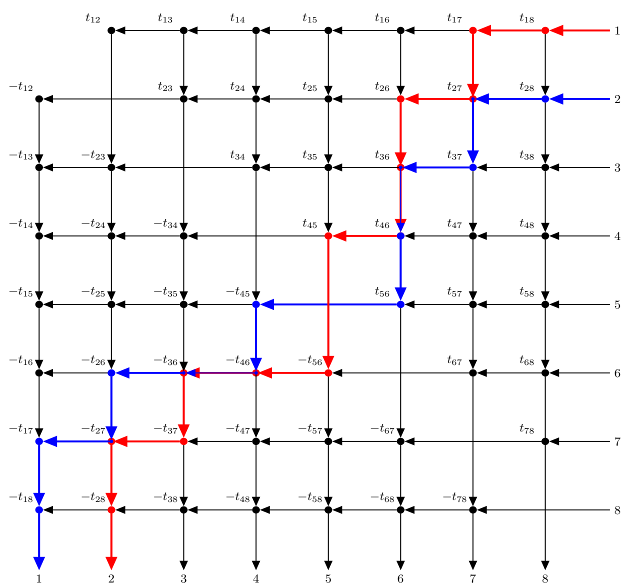

Figure 3: A red path $p : 1 \to 2$, and the blue path is its corresponding transpose, $p^T : 2 \to 1$

3.1 Embedding Quantum Skew-Symmetric Matrices

The following conjecture formally states the embedding of the algebra of quantum skew-symmetric matrices into a subalgebra of the quantum skew-symmetric torus.

Conjecture 1. *Define* $\phi : \mathcal{O}_q(Sk_n(\mathbb{K})) \to T_n^{\times}$ *as*

$$\phi(x_{(i,j)}) = \hat{x}_{(i,j)} = \sum_{p:i \to j} w(p)$$

ϕ is an isomorphism from $\mathcal{O}_q(Sk_n(\mathbb{K}))$ to the subalgebra of $T_n^{\times}$ generated by $\{\hat{X}_{(i,j)}\}$.

We study this map by studying the commutation relations of each algebra. If an isomorphism exists, the two algebras should have the same non-commutative structure. In particular, the commutation relation we focus on throughout this paper is q-commuting. In other words, if generators in a quantum skew-symmetric matrix q-commute, their corresponding generators of the subalgebra should also q-commute. We state this notion more formally below.

Conjecture 2. *Let $x_{(i,j)}, x_{(l,k)} \in \mathcal{O}_q(Sk_n(\mathbb{K}))$. Let $P : i \to j$ be the set of paths from i to j and $Q : l \to k$ be the set of paths from l to k. If $x_{(i,j)}x_{(l,k)} = q^* x_{(l,k)}x_{(i,j)}$, where q^* is the quantum parameter, then*

$$\sum_{p \in P: i \to j} \sum_{q \in Q: l \to k} w(p)w(q) = q^* \sum_{q \in Q: l \to k} \sum_{p \in P: i \to j} w(q)w(p). \tag{1}$$

Studying these summations has lead to the following observation.

Conjecture 3. *Let $p_1, p_2 \in P : i \to j$ and $q_1, q_2 \in Q : l \to k$, such that (1) holds. Then, the following is true.*

$$p_1 q_1 = q^* q_2 p_2 \iff p_2 q_2 = q^* q_1 p_1$$

4 Classifying Paths

4.1 Equivalence Classes on Paths and Path Weights

Theorem 1. *Let p_1, p_2 be two paths from i to j. Then $w(p_1) = q^k w(p_2)$ for some $k \in \mathbb{Z}$ if and only if $E(p_1) \bigcup E(p_1^T) = E(p_2) \bigcup E(p_2^T)$.*

Definition 11. We define an equivalence relation on paths, such that $p_1 \equiv_p p_2$ if and only if

$$E(p_1) \bigcup E(p_1^T) = E(p_2) \bigcup E(p_2^T).$$

We also define an equivalence relation on path weights, such that $w(p_1) \equiv_w w(p_2)$ if and only if

$$w(p_1) = q^k w(p_2)$$

for some $k \in \mathbb{Z}$

We will prove this theorem at the end of the subsection.

Lemma 1. *Let p be any path on G^n. Then $w(p) = -q^k w(p^T)$ for some $k \in \mathbb{Z}$.*

Proof. We know that p^T is the reflection of p across the diagonal. Thus every turn of p is reflected across the diagonal. Note that reflecting a turn across the diagonal does not change the turn type. Thus for every vertex v in p the weight function of v is the same as the weight function of the reflection of v, $-v$ in p^T. Thus $w_p(v) = w_{p^T}(-v)$ for all $v \in p$. Thus $w(p) = (-1)^n q^k w(p^T)$ for some $k, n \in \mathbb{Z}$.

Note that the last turn of any path must be a Γ turn, and the turn types of a path alternate. If the first turn of a path below the diagonal is a Γ turn, the first turn of its transpose below the diagonal is a $\lrcorner$ turn, and vice versa. Thus the number of turns of p always has different parity from the number of turns of p^T, so we can say that n is always 1 in the equation above. $\square$

Lemma 2. *Let $p : i \to j$ be a path. If there are vertices $v_1, v_2 \in p$ such that $w(v_1) = (-1)^n w(v_2)^{-1}$, then v_1 and v_2 both have degree 4 in $E(p) \bigcup E(p^T)$ and $v_1 = -v_2$.*

Proof. Suppose $w(v_1) = (-1)^n w(v_2)^{-1}$. Each generator of the quantum torus is only associated with two vertices of the grid and those two vertices are always symmetric across the diagonal, so we immediately get that $v_1 = -v_2$. If there is a Γ turn at v_1 in p, then there must be a Γ turn at v_2 in p^T and if there is a Γ turn at v_2 in p, then there must be a Γ turn at v_1 in p^T. Since v_1 and v_2 have different turn types in p, at least one of them is a Γ turn in p. Thus there is a Γ turn at both v_1 and v_2 in $E(p) \bigcup E(p^T)$. The same argument shows that there is a $\lrcorner$ turn at both v_1 and v_2 in $E(p) \bigcup E(p^T)$. Therefore both v_1 and v_2 have both a Γ turn and a $\lrcorner$ turn in $E(p) \bigcup E(p^T)$, so they both have degree 4 in $E() \bigcup E(p^T)$. $\square$

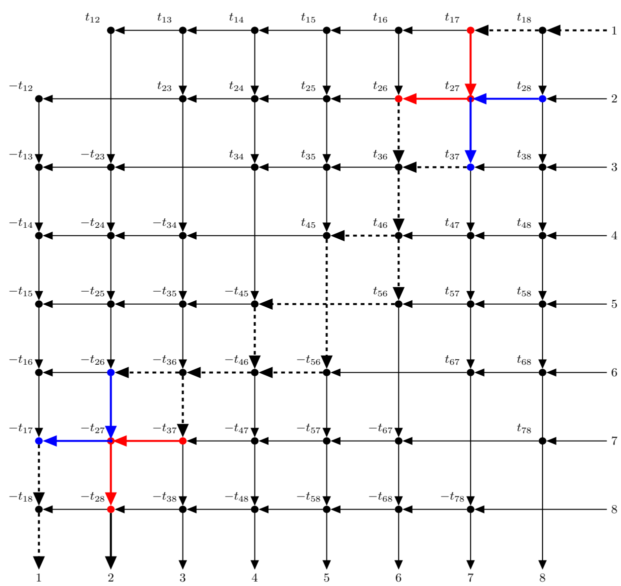

Figure 4: We've highlighted 8 edges from $E(p) \bigcup E(p^T)$ in Figure 3. When putting the weight of p into lexicographic order, t_{27}^{-1} and t_{27} will cancel out.

Definition 12. Let p be a path from $p : i \to j$ such that $i < j$. The *canonical path* associated with p is defined as the path $p_c : i \to j$ such that $p_c \equiv p$ and p_c has no turns at any vertex that has degree 4 in $E(p) \bigcup E(p^T)$.

Lemma 3. *Let p be a path, and let $t_{(i,j)}$, $-t_{(i,j)}$ be two vertices such that both have degree 4 in $E(p) \bigcup E(p^T)$. Then for any $p_i \in \{p_k | p_k \equiv p\}$, p_i either turns at both $t_{(i,j)}$ and $-t_{(i,j)}$ with distinct turn types or p_k doesn't turn at $t_{(i,j)}$ or $-t_{(i,j)}$.*

Proof. Let p_k be a path in $\{p_k | p_k \equiv p\}$. Then $t_{(i,j)}$, $-t_{(i,j)}$ have degree 4 in $E(p_k) \bigcup E(p_k^T)$. Suppose p_k turns at $t_{(i,j)}$. In order for $t_{(i,j)}$ to have degree 4 in $E(p_k) \bigcup E(p_k^T)$, it must be the case that p_k^T also turns at $t_{(i,j)}$ with a different turn type. Then p_k turns at $-t_{(i,j)}$. $\square$

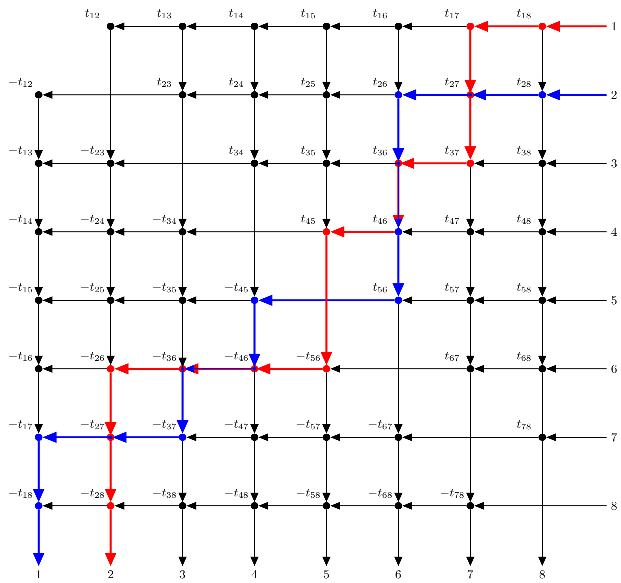

Figure 5: The red path of this figure is p_c, and the blue path is p_c^T

.

Lemma 4. *Let p be a path, and let p^T be the transpose. Let v_1 and v_n be the first and last vertices. Let v_k be the first vertex of p with degree 4 where p turns. We write*

$$p = p_1 \circ p_2 \circ p_3$$

where $p_1 : v_1 \to v_k, p_2 : v_k \to -v_k, p_3 : -v_k \to v_n$. Then let

$$p' = p_3^T \circ p_2 \circ p_1^T.$$

Then $w(p) \equiv w(p')$.

Proof. Note that

$$w(p) = w(p_1)w(v_k)w(p_2)w(-v_k)w(p_3).$$

Obviously $w(p_2) = w(p_2)$. By Lemma 1 $w(p_1) \equiv w(p_1^T)$ and $w(p_3) \equiv w(p_3^T)$. Note that $p' \equiv p$, so v_k and $-v_k$ also have degree 4 in $E(p') \bigcup E(p'^T)$. Note that since p turned at v_k, p' must not turn at v_k. Then by Lemma 3, p' does not turn at v_k or $-v_k$, in which case $w(v_k) = w(-v_k) = 1$. $\square$

Corollary 1. *For any path p, $w(p) \equiv w(p_c)$, where p_c is the canonical path of p.*

Theorem 2. *Let p_1, p_2 be paths. Then, $p_1 \equiv_p p_2$ if and only if $w(p_1) \equiv_w w(p_2)$.*

Proof. Let p_1, p_2 be paths such that $p_1 \equiv p_2$. Then $w(p_1) \equiv w(p_c) \equiv w(p_2)$ by Corollary 1.
Let p_1, p_2 be paths such that $p_1 \not\equiv p_2$. Then let p_c^1 and p_c^2 be their associated canonical paths. Then $p_c^1 \not\equiv p_c^2$. By Lemma 2, any elements of the quantum torus that cancel out in the weight of a path p are the image of a vertex with degree 4 in $E(p) \bigcup E(p^T)$, and by definition, a canonical path has no turns at vertices with degree 4 in $E(p) \bigcup E(p^T)$. Thus can be no cancellations in the weight of the canonical path. Therefore if $p_c^1 \not\equiv p_c^2$, then $w(p_c^1) \not\equiv w(p_c^2)$, so

$$w(p_1) \equiv p_c^1 \not\equiv p_c^2 \equiv p_2.$$

$\square$

Corollary 2. *Let p_1, p_2, q_1, q_2 be some paths, such that $w(p_1)w(q_1) \equiv_w w(p_2)w(q_2)$. This implies,*

$$E(p_1) \cup E(q_1) \cup E(p_1^T) \cup E(q_1^T) = E(p_2) \cup E(q_2) \cup E(p_2^T) \cup E(q_2^T).$$

4.2 Transpose Disjoint Paths

Definition 13. Let p be a path, such that $p \cup p^T \neq \emptyset$. Let v_1 be the first vertex of p and v_n be the last. Let v_k be the last intersection of p and p^T above the diagonal and v_k^T the first intersection of p and p^T after the diagonal. We write $p = p_1 \circ p_2 \circ p_3$, where $p_1 : v_1 \to v_k$, $p_2 : v_k \to v_k^T$, $p_3 : v_k^T \to v_n$. Then, set

$$\tau(p) = p_1 \circ p_2^T \circ p_3.$$

Proposition 1 (K. Casteels, 2016)**.** *If $p : i \to j$ is transpose non-disjoint, then*

$$w(p) + w(\tau(p)) = 0.$$

Thus, we only consider transpose disjoint paths.

5 Guide to the Code

Below we describe the Mathematica code written to aid our research. Helper functions not intended to be used by a human are omitted.

5.1 General Functions

To aid our research, we wrote extensive code in Mathematica to generate examples. General functions for working with quantum matrices and quantum skew-symmetric matrices include the following:

Function 1. xyPath[{x,y,m,n}]

xyPath generates the set of paths $P : i \to j$ for a given (i, j) on an $m \times n$ grid. Paths are represented as a list of coordinates, where each coordinate is a two element list. Note that even when we work with skew-symmetric quantum matrices, the coordinates will be labeled by their location on the grid. (In the code we treat paths for both models as the same. Differences between the models will be addressed when we work with the path weights).

Example 2. xyPath[{1,1,2,2}] would return the two paths drawn in Figure 1 in the following nested list:

$$\{\{\{1, 2\}, \{2, 2\}, \{2, 1\}\}, \{\{1, 2\}, \{1, 1\}, \{2, 1\}\}\}$$

Function 2. translateTurn[{vertex, turnType}]
translateTurn returns the weight of a turn in a path for the quantum matrices path model. The input "vertex" is a coordinate of the form $\{i, j\}$. If there is a down turn at $\{i, j\}$, then $t_{(i,j)}$ is formatted as

Subscript[Superscript[t, 1], $\{i, j\}$]

$t^{-1}_{(i,j)}$ is formatted as

Subscript[Superscript[t, -1], $\{i, j\}$]

Mathematica treats the outputted string $t^1_{\{i,j\}}$ as a nested list, such that the structure is equivalent to

$\{\{$t, 1$\}, \{i, j\}\}$

To retrieve either the exponent or the subscript, you would use the normal list access methods.

Function 3. translateSkewTurn[{vertex, turnType}]
translateSkewTurn returns the weight of a turn in a path for the quantum skew-symmetric matrices path model. Unlike translateTurn, the output is a list instead of a string. The input "vertex" should be the coordinate of the turn, and "turnType" is specified as in translateTurn.
If $i < j$ then the output is

$$\{\text{translateTurn}[\{\{i, j\}, \text{turnType}\}]\}$$

if $i > j$ then the output is

$$\{\text{"-"}, \text{translateTurn}[\{\{j, i\}, \text{turnType}\}]\}$$

Function 4. pathToMonomial[path],skewPathToMonomial[path]
pathToMonomial outputs the path weight of a path according to the quantum matrices path model. skewPathToMonomial outputs the path weight of a path according to the quantum skew-symmetric matrices path model.

Function 5. sortMonomial[monomial]
sortMonomial takes a path weight outputted by pathtoMonomial or skewPathtoMonomial and does the following:

- Cancels out negatives
- Puts the monomial in lexicographic order
- Cancels out terms

The output is a list of terms formatted with Row[]. To retrieve the list use [[1]]. For example, if the output is

$$q^1 t^1_{\{1,3\}} t^{-1}_{\{2,3\}} t^1_{\{2,5\}}$$

then

$$q^1 t^1_{\{1,3\}} t^{-1}_{\{2,3\}} t^1_{\{2,5\}}[[1]] \text{ gives } \{q^1, t^1_{\{1,3\}}, t^{-1}_{\{2,3\}}, t^1_{\{2,5\}}\}$$

The first entry of the list will always be q^k for some $k \in \mathbb{Z}$. If the weight has no q-coefficient in lexicographic order then the first entry will just be q^0. The q-coefficient is formatted as

$$\text{Superscript[q,k]}$$

To access the value k, you would type q^k[[2]].

Function 6. matrixTerm[{i,j,m,n}]
Outputs a sum of path weights associated with an entry $x_{(i,j)}$ of the matrix of generators of quantum matrices. All of the summands will be in lexicographic order as outputted by sortMonomial.

Function 7. skewMatrixTerm[{i,j,m,n}]
Outputs a sum of path weights associated with an entry $x_{(i,j)}$ of the matrix of generators of quantum skew-symmetric matrices. All of the summands will be in lexicographic order as outputted by sortMonomial.

Function 8. matrix[{m,n}]
Outputs the matrix of generators for quantum matrices according to the corresponding path model.

Function 9. skewMatrix[{m,n}]
Outputs the matrix of generators for quantum skew-symmetric matrices according to the corresponding path model.

5.2 Elements of the Double Sum

To work on our specific problem, we wrote functions to work on the product of path weights in the double sum. We exclude paths that are non transpose disjoint and paths that cross the diagonal horizontally and their corresponding path weights from the outputs of our functions, since those path weights end up cancelling out in the double sum.

Function 10. distributeWeightSums[{i,j,k,l,m,n}]
Distribute weight sums returns a Mathematica association. For each $p : i \to j$ and $q : k \to l$, $w(p)w(q)$ put into lexicographic order is a key in the association, such that the value associated with the key is

$$\{\{w(p_1), w(q_1)\}, \{w(p_2), w(q_2)\}, \dots\}$$

such that

$$w(p_i)w(q_i) = w(p)w(q)$$

for all i. The weights in the value are in their original order, so that one may use origTermsToPath to retrieve the paths that they correspond to.

Function 11. origTermsToPath[listOfVertexWeights]
Takes a list of a vertex weights and outputs the corresponding path. Note that the vertex weights must be in the original order. Also, the input does not match the output of pathToMonomial. When entering a monomial that is formatted as a string, such as the output of pathToMonomial, accessing the first element of the monomial as if it were a list gives you the list of terms, which you can use as input for origTermsToPath.

Function 12. findMatchingPaths[{i,j,k,l,m,n}]
Find matching paths returns a Mathematica association. Like distribute weight sums, the keys are $w(p_1)w(q_1)$ in lexicographic order for each $p_1 : i \to j$ and $q_1 : k \to l$. The value associated with the key is $q^{-1}w(p)w(q)$ put into lexicographic order. This function is designed to be used in conjunction with distributeWeightSums to retrieve pairs of paths that "map" to one another.

Function 13. pathPairVariations[{path1, path2}]
Given a path pair, outputs the set of path pairs obtained from switching segments of the pair between their intersections.

Function 14. switchWithTranspose[{path1,path2}]
Given a path pair, outputs the set of path pairs obtained from switching segments of path1 with the transpose of path2 and switching segments of path2 with the transpose of path1.

5.3 Plotting Paths

We also include the following functions to plot relevant paths:

Function 15. generateGrid[n]
Generates the Graphics directives for a skew-symmetric grid.

Function 16. plotPathFactors[{product, assoc,size}]
Given the association outputted by distributeWeightSums and a key (product of weights) from that association, plots the corresponding two path weights.

Function 17. plotPair[{{product1, assoc1},{product2, assoc2}, size}]
Given two associations outputted by distributeWeightSums and keys from that association, plots each pair of path weights side by side.

Function 18. makePicture[{path1, size}], makePicture[{path1,path2, size}]
makePicture plots one or two paths on the same grid, and adds the transpose(s) to the picture.

6 Future Work

Given a commutation relation between any two generators of the algebra of quantum skew-symmetric matrices, we conjecture that the corresponding generators of the subalgebra of the quantum skew-symmetric torus generated by our path model share this same commutation relation. Future work involves identifying this map, as well as identifying path models for other quantized algebras.

Acknowledgments

This research was supported by funds from the National Science Foundation. The authors thank their mentor, Karel Casteels.

References

[1] K. Casteels, *Quantum Matrices by Paths*, Algebra Number Theory **8** (2014), 1857-1912. ↑1

[2] Mitsuhiro Takeuchi and Bernd Strüber, *A Short Course on Quantum Matrices* (2002). ↑

[3] K. A. Brown and K. R. Goodearl, *Lectures on Algebraic Quantum Groups*, 2002. ↑