GENERALIZED COUNTING CONSTRAINT SATISFACTION PROBLEMS WITH DETERMINANTAL CIRCUITS

JASON MORTON\textsuperscript{1} AND JACOB TURNER\textsuperscript{1}

Abstract. Generalized counting constraint satisfaction problems include Holant problems with planarity restrictions; polynomial-time algorithms for such problems include matchgates and matchcircuits, which are based on Pfaffians. In particular, they use gates which are expressible in terms of a vector of sub-Pfaffians of a skew-symmetric matrix. We introduce a new type of circuit based instead on determinants, with seemingly different expressive power. In these determinantal circuits, a gate is represented by the vector of all minors of an arbitrary matrix. Determinantal circuits permit a different class of gates. Applications of these circuits include a new proof of the Chung-Langlands formula for the number of rooted spanning forests of a graph and a strategy for simulating quantum circuits with closed timelike curves. Monoidal category theory provides a useful language for discussing such counting problems, turning combinatorial restrictions into categorical properties. We introduce the counting problem in monoidal categories and count-preserving functors as a way to study \texttt{FP} subclasses of problems in settings which are generally \#\texttt{P}-hard. Using this machinery we show that, surprisingly, determinantal circuits can be simulated by Pfaffian circuits at quadratic cost.

Keywords: counting complexity, tensor network, monoidal categories

MSC2010: 15A15, 15A69, 15A24, 18D10, 03D15

1. Introduction

Let \text{Vect}_C be the category of finite-dimensional vector spaces and linear transformations over the base field \text{C}. A string diagram \texttt{[9]} in \text{Vect}_C is a tensor (contraction) network. Fixing such a diagram, the problem of computing the morphism represented is the tensor contraction problem, which is in general \#\text{P}-hard (examples include weighted counting constraint satisfaction problems \texttt{[5]}).

We study complex-valued tensor contraction problems in subcategories of \text{Vect}_C by considering them as diagrams in a monoidal category. For a survey of the rich diagrammatic languages that can be specified similarly see \texttt{[19]} and the references therein. By a circuit we mean a combinatorial counting problem expressed as a string diagram in a monoidal subcategory of \text{Vect}_C (that is, a tensor contraction network). Such diagrams generalize weighted constraint satisfaction problems and Boolean circuits; for example fanout (which copies a bit) and swap (which exchanges two bits) are not necessarily available. As in the Boolean case, such circuits often have more familiar description languages. Subcategories of \text{Vect}_C faithfully represent Boolean \texttt{[12]} and quantum circuits\textsuperscript{2}, counting constraint satisfaction problems, and many other problems\textsuperscript{7}.

\textsuperscript{1}arXiv:1302.1932v2 [math.CT] 10 Oct 2013
Suppose we have a problem $L$, for example a counting constraint satisfaction problem given by an embedded planar graph with variables or weighted constraints at each vertex. We can consider such a problem to be described by the data of a monoidal world (see e.g. [11], Chapter 12). We consider an interpretation map $i : L \to \text{Vect}_C$ that gives an interpretation of the problem of interest as a tensor contraction problem in $\text{Vect}_C$. The class $\text{FP}$ comprises the functions $\{0, 1\}^n \to \mathbb{N}$ computable by a deterministic polynomial-time Turing machine (see e.g. [1, p. 344]). A second, count-preserving functor $h : C \to S$ from a category $C$ in which the contraction problem (Problem 2.1) is in $\text{FP}$ and a subcategory $S$ of $\text{Vect}_C$ serves to characterize the problems which can be solved in polynomial time according to a particular contraction scheme. Many such schemes (such as holographic algorithms [20]) work by exploiting some combinatorial identity or kernel relating an exponential sum (corresponding to performing the tensor contraction by a na"ive algorithm) and a polynomial time operation (often the determinant or Pfaffian of a matrix) that yields the same result. They can be viewed as a complementary alternative method to geometric complexity theory [18] in the study of which counting problems (such as computing a permanent) may be embedded in a determinant computation at polynomial cost.

We formulate a class of circuits based on determinants and show that it is always solvable in polynomial time. This existence of such a class was conjectured in [13]. We give some applications of these circuits including a new proof of the Chung-Langlands formula [6] for the number of rooted spanning forests of a graph. We explain the relationship between Pfaffian circuits [16] (and so matchgates) and determinantal circuits and prove a functorial relationship between them. We show that, surprisingly, every determinantal circuit can be expressed as a Pfaffian circuit at quadratic cost.

This paper is organized as follows. In Section 2, we describe the counting problem in monoidal categories and the setting for our results. In Section 3 we define determinantal circuits and give applications to the Chung-Langlands rooted spanning forest theorem (Section 4) and quantum circuits with postselection-based closed timelike curves (P-CTC, Section 4.3). In Section 5, we relate determinantal and Pfaffian circuits. A remark on notation: in most cases we use $M, N$ for matrices, $I, J$ for sets (especially of indices), $f, g$ for morphisms, $F, G$ for functors, and $C, M$ for categories. For a matrix $X$, we let $X_{IJ}$ be the submatrix with rows in $I$ and columns in $J$.

2. Toward a categorical formulation of counting complexity

Let $\mathcal{M}$ be a (strict) monoidal category [15] with monoidal identity $1_{\mathcal{M}}$ and such that $S_{\mathcal{M}} = \text{Hom}_{\mathcal{M}}(1_{\mathcal{M}}, 1_{\mathcal{M}})$ is a semiring; call this a semiringed category. A monoidal word is a collection of morphisms composed and tensored together to form a new morphism [11].

**Problem 2.1.** The counting problem in a semiringed category $\mathcal{M}$ is to determine which morphism in $\text{Hom}_{\mathcal{M}}(1_{\mathcal{M}}, 1_{\mathcal{M}})$ is represented by an arbitrary monoidal word in $\mathcal{M}$ with $1_{\mathcal{M}}$ as its domain and codomain.

Over $\text{Vect}_C$, this is sometimes called the tensor contraction problem. In general Problem 2.1 is $\#P$-hard as the following example illustrates.
Example 2.2. Consider the monoidal category in which the objects are cartesian powers of a two-element set $B = \{T, F\}$. For example
\[\{T, F\}^2 = \{(T, T), (T, F), (F, T), (F, F)\}\]
which we denote by $\{T \otimes T, T \otimes F, F \otimes T, F \otimes F\}$ since the tensor product of objects in this category is the cartesian product of the corresponding sets. Morphisms are nonnegative integer valued binary relations such as $\{2 \cdot (T \otimes F; T \otimes T), 3 \cdot (T \otimes F; T \otimes F)\}$. The composition of morphisms $f$ and $g$ is $f \circ g = \{(\beta \gamma) \cdot (a, c) : \beta \cdot (a, b) \in f, \gamma \cdot (b, c) \in g\}$ where $\alpha, \beta, \gamma \in \mathbb{Z}_{\geq 0}$.

This is a semiringed category with $S_M = \mathbb{N}$ in which the counting problem is #P-hard, and the familiar category of counting constraint satisfaction problems can be expressed as a symmetric monoidal subcategory. Every constraint satisfaction problem can be visualized by a bipartite graph. One independent set is the variables, each of which can be assigned the value true or false. Thus we allow the morphisms in $\text{Hom}_{\mathcal{B}^n}(1, 1)$ whose coefficients are only zero or one. The string diagrams in this category will then be precisely the graphs visualizing boolean constraint satisfaction problems. In the sequel we generally use the set $\{0, 1\}$ instead of $\{F, T\}$ to conform with common usage.

A strict monoidal functor $F : \mathcal{M} \to \mathcal{M}'$ between semiringed categories is count preserving if the induced map $F : S_M \to S_{M'}$ is an injective morphism of semirings. Schemes that generalize holographic algorithms [20] seek a count-preserving functor from a category in which the counting problem (Problem 2.1) is in $\text{FP}$ to a category in which the counting problem is in general #P-hard.

In each type of circuit, we consider two semiringed categories $\mathcal{C}$ and $\mathcal{S}$. Let $\mathcal{L}$ be a problem of interest. We call $\mathcal{C}$ the counting category and $\mathcal{S}$ is a subcategory of $\text{Vect}_\mathcal{C}$. Then let $f : \mathcal{L} \to \mathcal{C}$ be a map that gives an interpretation or encoding of the problem as a string diagram in $\mathcal{C}$. By this we mean that for every instance of a problem $l \in \mathcal{L}$, $f(l)$ is a string diagram that solves this instance of the problem.

The category $\mathcal{C}$ may have a non-intuitive encoding of the problem but has the advantage that there exists a polynomial-time algorithm to determine which morphism of $S_M = \text{Hom}(1, 1)$ is represented by an arbitrary monoidal word. We also have an interpretation $g : \mathcal{L} \to \mathcal{S}$. Then we want a monoidal functor $h$ such that the diagram

\[
\begin{array}{ccc}
\mathcal{L} & \xrightarrow{f} & \mathcal{C} \\
\downarrow{g} & & \downarrow{h} \\
\mathcal{S} & \xrightarrow{} & \mathcal{S}
\end{array}
\]
commutes and such that the count is preserved by $h$. $\mathcal{S}$ is the subcategory generated by the morphisms in the image of either $h \circ f$ or $g$. The induced maps on $S_C$ and $S_S$ make $S_C$ a sub-semiring of $S_S$. The functor $h$ is called sDet, and sPf for determinantal and Pfaffian circuits respectively in the sequel.

Of course, it is important that the construction represented by the functors is implementable in polynomial time. Often this is not a concern, because diagrams in $\mathcal{C}$ and $\mathcal{L}$ are effectively identified, and the problem is expressed in the language that will be used to perform the contraction.
3. Determinantal circuits

Suppose $X$ is an $n \times m$ matrix of elements of a field $k$ with rows and columns labeled by finite disjoint subsets $N$ and $M$ of $\mathbb{N} = \mathbb{Z}_{\geq 0}$. For $i \in \mathbb{N}$, let $V_i \cong k^2$ be spanned by an orthonormal basis (with inner product) $v_{i,0}, v_{i,1}$ and for finite $N \subset \mathbb{N}$ write $V_N := \bigotimes_{i \in N} V_i$. Define the function $s\text{Det}$ by

$$s\text{Det} : \text{Mat}(n, m) \to V_N^* \otimes V_M \cong (\mathbb{C}^2)^{\bigotimes n} \otimes (\mathbb{C}^2)^{\bigotimes m}$$

$$s\text{Det}(X) = \sum_{I \subseteq \{n]\}, J \subseteq \{m\}} \det(X_{I,J}) |I\rangle \langle J|$$

where $|I\rangle = \bigotimes_{i \in N} v_{i,\chi(i,I)}$, $\langle J| = \bigotimes_{i \in M} v_{i,\chi(i,J)}$ and the indicator function $\chi(i,I) = 0$ if $i \notin I$ and 1 if $i \in I$.

This subdeterminant function $s\text{Det}$ induces a strong monoidal functor $s\text{Det} : \mathcal{C} \to \text{Vect}_\mathbb{C}$ from a matrix category to a subcategory $\mathcal{D}$ of $\text{Vect}_\mathbb{C}$. Let $\mathcal{C}$ be the free monoidal category described as follows. The objects of $\mathcal{C}$ are finite ordered subsets of $\mathbb{N}$ (which may have repeated elements), with monoidal product union. The morphisms are $\mathcal{C}$-valued matrices with rows and columns labeled by subsets of $\mathbb{N}$. If $M,N$ are two matrices with the set of row labels of $M$ equal to the set of column labels of $N$, order them and let $N \circ M = NM$ be the ordinary matrix product, with the resulting matrix inheriting the row labels of $N$ and the column labels of $M$. The monoidal product $\otimes_\mathcal{C}$ is the direct sum of labeled matrices.

Let $\mathcal{D}$ be the image of $\mathcal{C}$ in $\text{Vect}_\mathbb{C}$. It will be the free dagger symmetric traced monoidal subcategory of finite-dimensional $\mathbb{C}$-vector spaces generated by the object $\mathbb{C}^2$, endowed with an orthonormal basis, and morphisms $s\text{Det}(M)$ for $M$ a labeled matrix. Tensor product and composition/contraction are the usual operations.

Definitions of free dagger symmetric traced monoidal subcategories and related concepts are given in [10] [19], and in the proofs below we check the necessary axioms.

For $f : A \to A \in \mathcal{C}$, define $\text{tr}(f) = \det(I + f)$ and define trace in $\mathcal{D}$ in the usual way. For a matrix $M$ representing $f$, $\text{tr}(M) = \sum_{I \in |M|} \det M_i$.

**Proposition 3.1.** $\mathcal{C}$ is a strict dagger symmetric monoidal category.

**Proof.** In this proof we denote the monoidal product $\otimes_\mathcal{C}$ for $\mathcal{C}$ defined above simply as $\otimes$. We need to show that it is a bifunctor. For $A \subset \mathbb{N}$, $\text{id}_A$ is the identity matrix with row and column labels $A$. It is easy to see that for any $A,B \subset \mathbb{N}$, $\text{id}_A \otimes \text{id}_B = \text{id}_{A \otimes B}$. Now for morphisms $W,X,Y,Z \in \text{Mor}(\mathcal{C})$, $W \otimes X \circ Y \otimes Z = (W \oplus X)(Y \oplus Z) = WY + XZ = (W \circ Y) \otimes (X \circ Z)$, so $\otimes$ is indeed a bifunctor.

![Figure 1. An example of a determinantal circuit (wires oriented clockwise). The four tensors in $\text{Vect}_\mathbb{C}$, from left to right, are obtained by applying $s\text{Det}$ to each matrix. Letting $V = \mathbb{C}^2$, they lie in $(\mathbb{C}^2)^{\bigotimes 2} \otimes \mathbb{C}^2$, $(\mathbb{C}^2)^{\bigotimes 2} \otimes \mathbb{C}^2$, $\mathbb{C}^2 \otimes \mathbb{C}^2$, and $(\mathbb{C}^2)^{\bigotimes 2} \otimes \mathbb{C}^2$ respectively.](image)
For $A, B, C \in \text{Ob}(C)$, the associator $\alpha_{ABC} : (A \otimes B) \otimes C \to A \otimes (B \otimes C)$ is just equality by the associativity of matrix direct product. The unit for $C$, denoted $\mathbf{1}$, is the empty set. Then $\lambda_A : \mathbf{1} \otimes A \to A$ and $\rho_A : A \otimes \mathbf{1} \to A$ are also equality since it is union with $\emptyset$. It is clear that $\alpha, \lambda,$ and $\rho$ are natural isomorphisms.

We need to check that the diagrams from MacLane’s Coherence Theorem commute. First let us check, for $A, B \in \text{Ob}(C)$:

$$
\begin{align*}
(A \otimes \mathbf{1}) \otimes B & \xrightarrow{\alpha} A \otimes (\mathbf{1} \otimes B) \\
\rho_A \otimes \text{id}_B & \xrightarrow[]{} \text{id}_A \otimes \lambda_B \\
\alpha & \xrightarrow[]{} A \otimes B
\end{align*}
$$

$(A \otimes \mathbf{1}) \otimes B = (A \cup \emptyset) \cup B$ is mapped to $A \cup B$ by $\rho_A \otimes \text{id}_B$ via equality. Then $\alpha$ maps $(A \cup \emptyset) \cup B$ to $A \cup (\emptyset \cup B)$ via equality. This is then mapped to $A \cup B$ by $\text{id}_A \otimes \lambda_B$ via equality, and the diagram commutes.

Now let us check the second diagram, for $A, B, C, D \in \text{Ob}(C)$:

$$
\begin{align*}
((C \otimes A) \otimes B) \otimes D & \xrightarrow{\alpha \otimes \text{id}_D} (C \otimes (A \otimes B)) \otimes D \\
\alpha & \xrightarrow[]{} (C \otimes (A \otimes B)) \otimes D \\
\alpha & \xrightarrow[]{} C \otimes ((A \otimes B) \otimes D) \\
\text{id}_C \otimes \alpha & \xrightarrow[]{} C \otimes ((A \otimes B) \otimes D)
\end{align*}
$$

The object $((C \otimes A) \otimes B) \otimes D = ((C \cup A) \cup B) \cup D$ is mapped to $C \cup (A \cup (B \cup D))$ by $(\text{id}_C \otimes \alpha) \circ (\alpha \otimes \text{id}_D)$ via equality. Similarly, it is mapped to $C \cup (A \cup (B \cup D))$ by $\alpha \circ \alpha$ via equality. This diagram also commutes and so $C$ is a monoidal category. Furthermore, since $\alpha, \lambda,$ and $\rho$ are equalities, $C$ is a strict monoidal category.

The braiding for $C$ is a map $\imath_{A,B} : A \otimes B \to B \otimes A, A, B \in \text{Ob}(C)$. It is given by the matrix

$$
\begin{bmatrix}
B & A \\
A & 0 \\
B & 0
\end{bmatrix}.
$$

We need to check that the following diagrams commute for $A, B, C \in \text{Ob}(C)$:

$$
\begin{align*}
(B \otimes A) \otimes C & \xrightarrow{\imath_{A,B} \otimes \text{id}_C} B \otimes (A \otimes C) \\
\text{id}_B \otimes \imath_{A,C} & \xrightarrow[]{} (B \otimes C) \otimes A
\end{align*}
$$
The first diagram commutes by noting that
\[ (B \otimes A) \otimes C \xrightarrow{\alpha} B \otimes (A \otimes C) \]
\[ (A \otimes B) \otimes C \xrightarrow{\alpha} A \otimes (B \otimes C) \]
\[ B \otimes (C \otimes A) \]

The second diagram commutes since \( c_{A,B,C}^{-1} \) for any \( A,B \) is symmetric (in fact, an isomorphism) of dagger symmetric traced categories. Thus while computing a trace in \( \text{Vect}_C \) is in general \#P-hard, in the image of \( \text{sDet} \) it can be computed in polynomial time.

Theorem 3.2. The map \( \text{sDet} \) defines a strict monoidal functor which is an equivalence (in fact, an isomorphism) of dagger symmetric traced categories. Thus while computing a trace in \( \text{Vect}_C \) is in general \#P-hard, in the image of \( \text{sDet} \) it can be computed in polynomial time.

We prove this in two parts as Lemmata 3.3 and 3.4.

Lemma 3.3. The map \( \text{sDet} \) defines a strict monoidal functor which is an equivalence of monoidal categories.

Proof. We have defined how \( \text{sDet} \) acts on matrices, and so morphisms in \( C \). It takes an object \( A \in \text{Ob}(C) \) to \( \text{sDet}(A) = V_A = \bigotimes_{i \in A} V_i \).

First we must show that \( \text{sDet} \) is a functor, i.e., that it respects composition and that \( \text{sDet}(\text{id}_A) = \text{id}_{\text{sDet}(A)} \). Suppose \( X \in \text{Hom}_C(I,J) \), \( Y \in \text{Hom}_C(J,K) \) so \( X \) is a matrix with row labels \( I \) and column labels \( J \), \( Y \) has row labels \( J \) and column labels \( K \):

\[ \text{sDet}(Y) \circ \text{sDet}(X) = \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} \text{det}(X_{ij}) \text{det}(Y_{jk}) |i\rangle \langle k| \]

\[ = \sum_{i \in I} \sum_{j \in J} \text{det}(XY_{ik}) |i\rangle \langle k| = \text{sDet}(XY) \]

The dagger for \( C \) is given by matrix transpose and the identity on objects. Clearly \( \text{id}_{A}^T = \text{id}_{A} \). Given \( X,Y \in \text{Mor}(C) \), \( X : A \to B \), \( Y : B \to C \), \( (X \circ Y)^\dagger = \left((XY)^T \right) = Y^T X^T = Y^\dagger \circ X^\dagger : C \to A \). Lastly \( X^{\dagger \dagger} = X^{TT} = X \).

We also need the dagger to satisfy two extra properties since we are working in a monoidal category. First, given \( X,Y \in \text{Mor}(C) \), \( (X \otimes Y)^\dagger = (X \odot Y)^T = X^T \odot Y^T = X^\dagger \otimes Y^\dagger \). Secondly, \( \alpha, \lambda \), and \( \rho \) should all be unitary (its inverse is equal to its dagger). Since they are all the identity morphism, this is also satisfied. Thus \( C \) is indeed a strict dagger symmetric monoidal category. \( \square \)
where the middle equality is the Cauchy-Binet formula. Now in $C$, $\id_A$ is the identity matrix with row and column labels $A$. Then $\text{sDet}(\id_A) = \sum_{I \subseteq A} |I \times I|$ which is the identity morphism for the object $\text{sDet}(A)$ in $D$, and $\text{sDet}$ is indeed a functor.

For $\text{sDet}$ to be a monoidal functor, we must demonstrate two additional properties. First we must show that $\text{sDet}(A \oplus B) = \text{sDet}(A) \otimes \text{sDet}(B)$. Let $I$ and $J$ be the rows and columns of $A$, respectively. Let $I'$ and $J'$ be likewise for $B$. A straightforward calculation gives

$$\text{sDet}(A \oplus B) = \sum_{U \subseteq I, V \subseteq J} \sum_{U' \subseteq I', V' \subseteq J'} \det(A \oplus B)_{UV \setminus |U \setminus V|}$$

$$= \sum_{U \subseteq I, V \subseteq J} \sum_{U' \subseteq I', V' \subseteq J'} \det(A_{U \setminus I, V \setminus J}) \det(B_{V \setminus J', V \setminus J'})_{|U \setminus I'| \setminus |V \setminus J'|}$$

$$= \sum_{U \subseteq I, V \subseteq J} \sum_{U' \subseteq I', V' \subseteq J'} \det(A_{UV}) \det(B_{U'V'})_{|U' \setminus |V'\setminus |U\setminus V|} = \text{sDet}(A) \otimes \text{sDet}(B).$$

Secondly we must show there are morphisms $f_0 : \mathbf{1}_D \to \text{sDet}(\mathbf{1}_D)$ (the unit in $D$ is the base field $C$) and for any $A, B \in \text{Ob}(C)$, $f_1 : \text{sDet}(A) \otimes \text{sDet}(B) \to \text{sDet}(A \otimes C B)$ satisfying certain axioms expressed as commutative diagrams.

Since $\text{sDet}(\mathbf{1}) = \otimes_{i \in \mathbf{1}} V_i = C$, $f_0$ is simply equality. Similarly for objects $A$ and $B$,

$$\text{sDet}(A \otimes C B) = \text{sDet}(A \cup B) = \otimes_{i \in A \cup B} V_i = (\otimes_{i \in A} V_i) \otimes (\otimes_{j \in B} V_j) = \text{sDet}(A) \otimes \text{sDet}(B),$$

so $f_1$ is equality. In the following diagrams, we shall call $\text{sDet}$ simply $\mathbf{F}$.

Let $\alpha', \lambda', \rho'$ be the natural transformations for $D$. Note that all three are equalities. For $A, B, C \in \text{Ob}(C)$, the following must commute:

$$\mathbf{F}(A) \otimes (\mathbf{F}(B) \otimes \mathbf{F}(C)) \xrightarrow{\alpha'} (\mathbf{F}(A) \otimes \mathbf{F}(B)) \otimes \mathbf{F}(C).$$

The diagrams trivially commute as all of the maps are identities. So $\text{sDet}$ is a strong monoidal functor. Since $f_0, f_1$ are equalities, it is a strict monoidal functor.

Lastly, we want to say that $C$ and $D$ are equivalent as monoidal categories. By definition of $D$, $\text{sDet}$ surjects onto objects and morphisms, so it is a full functor. Now consider $\text{Hom}(A, B)$ for objects $A, B \in \text{Ob}(C)$. Let $X \in \text{Hom}(A, B)$. $\text{sDet}(X)$ contains all the entries of $X$ as coefficients in the sum since the entries of $X$ are $1 \times 1$ minors, and $X$ is determined by its image $\text{sDet}(X)$. Thus $\text{sDet}$ induces an injection on $\text{Hom}(A, B) \to \text{Hom}(\text{sDet}(A), \text{sDet}(B))$, and the functor is faithful. Thus it is an equivalence. However, it is not quite an isomorphism as $\text{sDet}$ does not give a bijection on objects. \hfill $\square$
We have yet to define the braiding and dagger for $\mathcal{D}$ required to state Theorem 3.2. For $F = s\text{Det}$ to respect the braiding, we need the following diagram to commute:

$$
\begin{array}{c}
F(A) \otimes F(B) \xrightarrow{f_1} F(A \otimes B) \\
\downarrow_{c_{F(A),F(B)}} \hspace{2cm} \downarrow_{F(c_{A,B})}
\end{array}
$$

Recalling the matrix $c_{A,B}$ as defined in Theorem 3.1, we define the braiding for $D$ to be $F(c_{A,B}) = s\text{Det}(c_{A,B}) = |00\rangle\langle 00| + |01\rangle\langle 10| + |10\rangle\langle 01| - |11\rangle\langle 11|$, which makes the diagram commute trivially. We do not check the diagrams that ensures this is a valid braiding for $D$ since it is equivalent to $C$. For the dagger, consider $X \in \text{Mor}(\mathcal{C})$ with row labels $I$ and column labels $J$, and note

$$
s\text{Det}(X^\dagger) = \sum_{i \in I, j \in J} \text{det}(X_{ij}^T) \langle i| \langle j| = \sum_{i \in I, j \in J} \text{det}(X_{ij}) \langle j| \langle i| = s\text{Det}(X)^T.
$$

So the dagger for $D$ is the normal dagger in $\text{Vect}_k$. Again for $f : A \to A \in \mathcal{C}$, define $\text{tr}(f) = \text{det}(I + f)$ and define trace in $D$ in the usual way.

**Lemma 3.4.** The map $s\text{Det}$ defines a strict monoidal functor which is an equivalence of dagger symmetric traced categories.

**Proof.** By construction, $s\text{Det}$ respects the braiding. We also showed that this functor respects the normal dagger for linear transformations. Theorem 3.6 and Proposition 3.7 below shows that $s\text{Det}$ induces the identity map from $\text{Hom}(I_{\mathcal{C}}, I_{\mathcal{C}}) \to \text{Hom}(I_D, I_D)$ and thus respects the trace.

**Remark 3.5.** This braiding is not the usual braiding for $\text{Vect}_\mathcal{C}$. Thus while the functor $s\text{Det}$ is count-preserving, the count will not be the same as if the standard braiding $u \circ v \mapsto v \circ u$ is used.

Using the operations of $\oplus$ and matrix multiplication, we can transform any string diagram in $\mathcal{C}$ into a diagram with a single matrix, $M$, and thus evaluate the determinantal circuit efficiently.

A determinantal circuit is the trace of a linear map defined by an expression of the form $(f_{1,1} \otimes \cdots \otimes f_{1,n_1}) \circ \cdots \circ (f_{m,1} \otimes \cdots \otimes f_{m,n_m})$. Let $d_k$ be the dimension of the domain of the $k$th linear map $(f_{k,1} \otimes \cdots \otimes f_{k,n_k})$, with $k = 1, \ldots, m$. The maximum width of such a circuit is $\max_{k=1,\ldots,m} d_k$ and the depth is $m$.

**Theorem 3.6.** The time complexity of computing the trace of a determinantal circuit in $\mathcal{C}$ is $O(dw^\omega) = O(dw^\omega + c^\omega)$ where $d$ is the depth of the circuit, $w$ is the maximum width, $c$ is width at the input and output (so can be chosen to be the minimum width), and $\omega$ is the exponent of matrix multiplication.

**Proof.** We have an $n \times n$ matrix with equal row and column labels, which we may assume to be $1, \ldots, n$. Then

$$
s\text{Det}(M) = \sum_{I,J \subseteq [n]} \text{det}(M_I|J) |I \rangle \langle J|
$$
and contracting this against itself gives
\[ \sum_{I,J \subseteq [n]} \det(M_{I,J}) \langle J| \langle J| = \sum_{I \subseteq [n]} \det M_{I,I}. \]
That is, the trace of a matrix \( M \) in \( C \) is the exponentially large sum of its \( 2^n \) principal minors; we claim that \( \det_p I^A \) is precisely this sum (Proposition 3.7).
This enables us to compute this number in time \( n^\omega \). □

The following identity is well-known (e.g. it can be derived from results in \([8]\)); we include a proof for completeness.

**Proposition 3.7.** Given an \( n \times n \) matrix \( M \),
\[ \det(I + M) = \sum_{J \subseteq [n]} \det(M_J) \]
where \( M_J = M_{J,J} \).

**Proof.** Let \( u_i \) be the columns of \( M \) and \( e_i \) the standard basis vectors, \( i \in [n] \). Then \( \det(I + M) = \bigwedge_{i=1}^n (e_i + u_i) \). Expanding this gives the sum of the determinants of all \( 2^n \) matrices with \( i \)th column either \( u_i \) or \( e_i \).

Consider one of these matrices, \( W \). Let \( J \subseteq [n] \) be the set of indices of the \( u_j \) appearing as columns in \( W \). Then for any \( j \notin J \), \( e_j \) is a column of \( W \). Using the Laplace expansion, \( \det(W) = \det(W_J) \), where \( W_J \) is \( W \) with the \( j \)th row and column omitted. Then iterating the Laplace expansion gives us that \( \det(W) = \det(M_J) \). □

A monoidal category is said to have duals for objects or be closed if each object \( A \) has a dual object \( A^* \) related by an adjunction \((A, A^*, i_A, e_A)\). Note that while \( D \) could be equipped with the object duality structure \((A, A^*, i_A, e_A)\) from the category of finite-dimensional vector spaces to obtain a dagger closed compact category, the matrix category \( C \) is not a closed compact category: it lacks the morphisms \( i_A \) (coevaluation) and \( e_A \) (evaluation). The morphism \( e_A : A \otimes A^* \to I \) would have to be the sDet of a 2 × 0 matrix, or the composition of several morphisms to obtain one of this type.

**Proposition 3.8.** The category \( C \) does not have duals for objects.

**Proof.** We cannot have \( e_A = \text{sDet}(M) \) for any \( M \). The morphism we want is \(|00\rangle + |11\rangle \), but there is a unique 2 × 0 matrix \( M \) and \( \text{sDet}(M) = |00\rangle \). □

As a consequence, we really do have to work with traced categories rather than the more convenient dagger closed compact categories \([10]\).

A diagram in the equivalent categories \( C, D \) is called a determinantal circuit; when the morphism represented is a field element, it computes the partition function, i.e. counts the weighted number of solutions to the weighted counting constraint satisfaction problem it represents. Because these categories have a traced, dagger braided monoidal category structure, they come with a corresponding graphical language \([19]\).

It is also a question of interest which tensors are determinantal. One can test whether a vector can be the set of determinants of minors from a matrix using the Plücker relations to obtain the relations among general minors of matrices. On the other hand, for minors of a fixed size this is an open problem \([3]\).
4. Applications

4.1. Multicycles. We now discuss a diagrammatic language and describe what determinantal circuits count in terms of multicycles. Our aim is to facilitate the application of determinantal circuits to specific counting problems.

Our convention shall be that tensors will be composed from right to left and that tensoring will be from top to bottom. A determinantal circuit is given as the trace of a composition of linear maps \((f_{1,1} \otimes \cdots \otimes f_{1,n_1}) \circ \cdots \circ (f_{m,1} \otimes \cdots \otimes f_{m,n_m})\). Let \(S_i = f_{i,1} \otimes \cdots \otimes f_{i,n_i}\). Let \(M^{S_i}\) be the matrix such that \(sDet(M^{S_i}) = S_i\). We call the \(S_i\) or associated \(M^{S_i}\) stacks. Pictorially, the situation is as follows:

\[
\begin{array}{c}
\text{Forgetting, for a moment, the categorical structure of the circuit, we consider the above as a graph.}
\end{array}
\]

**Definition 4.1.** A multicycle of a graph is an edge-disjoint union of cycles in the graph. We consider the empty graph a multicycle.

We are interested in whether a subgraph can be interpreted as several cycles, not which edges are in which particular cycles. Call two multicycles equivalent if they contain the same edges, and denote an equivalence class of multicycles by \(\mathcal{C}\).

**Definition 4.2.** A weighted multicycle of a determinantal circuit is a multicycle of the underlying graph where each cycle in the multicycle is assigned a scalar. The weight of the multicycle is the product of these scalars.

**Proposition 4.3.** Given a determinantal circuit, let \(\mathcal{M}\) be the set of all equivalence classes of its multicycles. There exists an assignment of a weight \(W_{\mathcal{C}}\) to every \(\mathcal{C} \in \mathcal{M}\) such that the value of the determinantal circuit is \(\sum_{\mathcal{C} \in \mathcal{M}} W_{\mathcal{C}}\).

**Proof.** A determinantal circuit with a single \(n \times n\) matrix \(M\) has value

\[
\det(I + M) = \sum_{I \subseteq [n]} \det(M_I) = \text{Tr}(sDet(M)) = \text{Tr}\left(\sum_{I \subseteq [n]} \det(M_I)|I\rangle\langle I|\right).
\]

A general determinantal circuit is the trace of a composition of stacks \(S_1 \circ \cdots \circ S_m\). Let \(E_k\) be the set of edges entering \(S_k\) from the left and exiting \(S_{k-1}\) to the right, and observe that

\[
\text{Tr}(sDet(M^{S_1} \circ \cdots \circ M^{S_m})) = \text{Tr}\left(\sum_{I_k \subseteq E_k} \prod_{k \subseteq [m]} \det(M^{S_k}_{I_k})|l_1\rangle\langle l_2|\cdots\langle l_m|l_m\rangle|l_1\rangle\right).
\]
We want to describe (1) as a sum over equivalence classes of multicycles of $S_1 \circ \cdots \circ S_m$. Consider the subgraph of the determinantal circuit whose edges are those in the sets $I_k$. We claim that if the subgraph does not correspond to an equivalence class of multicycles, $\prod \det(M_{I_k}^{S_k}) = 0$.

Each summand $\prod \det(M_{I_k}^{S_k})$ in (1) will be non-zero only if $|I_1| = \cdots = |I_m|$ as the determinant of a non-square matrix is zero. This implies that the number of edges entering a vertex from the left in the underlying graph must equal the number of edges exiting it to the right. This is sufficient for the circuit subgraph given by the subsets $I_k$ to be viewable as a multicycle.

We have not specified a cycle decomposition of the multicycle, so each circuit subgraph represents an equivalence class of multicycles with weight $\prod \det(M_{I_k}^{S_k})$.

**Example 4.4.** Suppose we are given the following determinantal circuit:

\[
\begin{matrix}
  & a & b \\
  \downarrow & & \downarrow \\
  c & b & d \\
  \uparrow & & \uparrow \\
  a & b \\
\end{matrix}
\]

Its value is the sum of the principal minors of the matrix: $1 + a + d + ad - bc$. In the picture below we draw the weighted multicycles in bold on the circuit:

- weight=1
- weight=a
- weight=d
- weight=ad − bc

**4.2. Recovering the matrix tree theorem and variants.** We describe how to recover the Chung-Langlands rooted spanning forest theorem by expressing it as a determinantal circuit.

**Theorem 4.5.** (Chung-Langlands [6]) Given a graph $G$, let $B$ be its incidence matrix endowed with an arbitrary orientation. Then $\det(I + BB^T)$ is the number of rooted spanning forests.

**4.2.1. Construction of a string diagram whose multicycles correspond to subgraphs of $G$.** We construct a string diagram $ZZ^!$ in $C$ which can be reduced to a determinantal circuit consisting of only the matrix $BB^T$ using the operations of $\oplus$ and matrix multiplication. An example of a graph is given in Figure 2(a) and the determinantal circuit constructed for it in Figure 2(b).
Choose an arbitrary orientation on the given graph $G = \{V, E\}$. We first build a string diagram, $Z$, from a collection of $C$-morphisms (nodes); there is one node for every edge and vertex of $G$. Denote an edge of $G$ by $\epsilon$, the edge node in $Z$ corresponding to it by $e$ and the edge node in $Z^\dagger$ corresponding to it by $e^\dagger$. Denote a vertex in $G$ by $\nu$ and its node in $D_G$ by $v$. An edge node is connected to a vertex node if the edge and vertex are incident in $G$.

Define an orientation on $Z$ which has no categorical meaning, but is used in the proof. An wire in $Z$ connecting an edge and vertex node is oriented towards the
vertex node if that vertex is a sink for the edge in \( G \); otherwise the wire is oriented towards the edge node. Arrange \( Z \) into two stacks: the first consists of the edge nodes, the second of the vertex nodes. The dashed box in Figure 2(b) gives an example of this construction.

Edge nodes are \( 1 \times 2 \) matrices, vertex nodes are \( d(\nu) \times 1 \) matrices, where \( d(\nu) \) is the degree of \( \nu \). The matrix \( M_e \) associated to an edge node \( e \) in \( Z \) is either \( [1 \ -1] \) or \( [-1 \ 1] \); it has a \(-1\) in the column corresponding to the output wire oriented away from \( e \) and a 1 in the other column. Let \( v \) be a vertex node. The matrix \( M_v \) associated with a vertex node \( v \) is a \( d(\nu) \times 1 \) matrix with every entry equal to 1. Although in general we suppress it in pictures, whenever two wires cross, we put the braiding matrix \( \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \) on the crossing.

**Lemma 4.6.** Using the operations of matrix multiplication and \( \oplus \), the matrices in \( Z \) collapse to the incidence matrix of \( G \) with some orientation placed on it.

**Proof.** Let \( E \) be the matrix equal to the direct sum of all the matrices on the edge nodes and \( V \) be the direct sum of all the matrices on vertex nodes. Then \( Z \) reduces to the matrix \( A = EPV \) where \( P \) is the permutation matrix obtained from crossed wires. Let \( e \) be an edge node and let \( r_e \) be the row vector of \( E \) corresponding to \( e \). For any column vector \( c_v \) of \( PV \) associated with vertex node \( v \), \( r_e \cdot c_v \neq 0 \) if and only if \( e \) is incident to \( v \). In fact, \( r_e \cdot c_v \) is equal to the number of wires \( v \to e \) minus the number of wires \( e \to v \) in \( Z \). This implies that \( A = B \), the incidence matrix. 

Reflect \( Z \) across a vertical line, transposing all node matrices, to obtain \( Z^\top \), which collapses to the matrix \( B^\top \). Our final circuit \( ZZ^\top \) is the composition of \( Z \) with \( Z^\top \). Figures 2(a) and (b), show an example of a graph \( G \) and its transformation into a circuit \( ZZ^\top \). Figure 2(c) is equivalent to 2(b) but we have composed the two stacks of vertex nodes. We denote the determinantal circuit like in Figure 2(c) associated to a graph \( G \), \( D_G \).

Let us turn our attention to the kinds of multicycles our circuit admits. Consider \( s\text{Det}(M_e) \), for an edge node \( e \). It is of the form \( |0\rangle\langle 0| \pm |1\rangle\langle 1| \mp |1\rangle\langle 0| \). This says that two distinct cycles in any multicycle cannot contain \( e \).

Let \( M_v \) be the matrix associated with the vertex node \( v \) (from here on out the matrices on vertex nodes will be as in Figure 2(c)). Let \( n \) be the degree of the corresponding vertex \( \nu \) in \( G \). Since \( M_v \) is a rank one matrix, the only nonzero principal minors are \( 1 \times 1 \) minors and the empty matrix. This means that for any multicycle, at most one of its cycles contains \( v \). So the conclusion as that for this determinantal circuit, all multicycles are vertex disjoint.

**Lemma 4.7.** Every multicycle of \( ZZ^\top \) can be associated with a subgraph of \( G \).

**Proof.** Here we are considering multicycles without weights to show that every multicycle of \( ZZ^\top \) may be associated to a subgraph of \( G \). For a particular multicycle \( \mu \), call a node in \( Z \) activated if there is a cycle in the multicycle that contains it. Let \( \mu(E) \) be the set of activated edge nodes in \( Z \) and \( \mu(E)^\top \) the activated edge nodes in \( Z^\top \). It must be that \( \mu(E) = \mu(E)^\top \). Then we associate \( \mu \) to the subgraph of \( G \) with edges \( \mu(E) \) along with those vertices incident to these edges.
4.2.2. Weighted multicycles and spanning forests.

**Observation 4.8.** In general, there may be more than one multicycle that is associated with a particular subgraph of $G$. Furthermore, these different multicycles may have different weights.

In our case, $D_G$ will have three stacks. We can break up a multicycle $\mu$ into three parts: $\mu_1, \mu_2,$ and $\mu_3$, where $\mu_i$ are the edges between the $i$th and $(i+1)$th (modulo 3) stacks.

Since a multicycle $\mu$ must have $\mu(E) = \mu(E)^\dagger$, so we can view $\mu$ as a permutation on a subset of the edges in $G$. This is well defined as the allowed multicycles are both edge and vertex-disjoint. Let $\text{sgn}(\mu)$ be the sign of this permutation.

**Lemma 4.9.** The weight of cycle in $D_G$ is:
- $-1$ if $p_1$ and $p_2$ are oriented in the same direction (left-to-right or right-to-left).
- $1$ if $p_1$ and $p_2$ are oriented in opposing directions.

The unassociated weight of a multicycle $\mu$, is the product of the weights of its constituent cycles. Its weight is its unassociated weight times $\text{sgn}(\mu)$.

**Proof.** This follows directly from how we defined $\text{sDet}$, the braiding used in $\mathcal{D}$, and the diagrammatic rules for tensor products in a monoidal category. □

**Definition 4.10.** A *straightline* multicycle $\mu$ is multicycle such that permutation associated with $\mu(E)$ is the identity permutation.
Figure 3(a) gives an example of a straightline multicycle whereas Figure 3(b) gives a non-example.

**Lemma 4.11.** Every straightline multicycle has weight 1.

**Proof.** Let $\mu$ be a straightline multicycle. Viewing $\mu$ as a permutation, the weight of $\mu$ is the unassociated weight of $\mu$. Now let us consider a cycle $C \in \mu$. Let $C$ connect the edge nodes $e$ and $e'$. Since $ZZ^\top$ is symmetric about the vertical axis, $C_1$ and $C_2$ are oriented in opposing directions. So $C$ has weight equal to 1 and multiplying all weights of cycles gives that the weight of $\mu$ is 1. 

**Lemma 4.12.** Every spanning forest is represented only by straightline multicycles. Furthermore every spanning forest has a representation as a multicycle.

**Proof.** Since any spanning forest can be broken into the disjoint union of trees, it is sufficient to show trees only have straightline multicycle representations. So let us suppose that there is a multicycle $\mu$ that is not straightline. Since $\mu$ is not the identity permutation on the edges activated, we can write it as the product of disjoint cycles. Let us consider one of these cycles, $\sigma$. This will be represented by a submulticycle $\mu'$. Let $E' = \{e_i\} = \mu'(E) = \mu'(E)^\top$, where $\sigma(e_i) = e_{i+1}$. Then $e_i$ and $e_{i+1}$ share a vertex for all $i$. Since $\sigma$ is a cycle, this implies that $\mu'$ represents cycle inside of the subgraph represented by $\mu$ and thus $\mu$ is not a tree.

In any spanning forest, one may associate any edge with a vertex adjacent to it such that no two edges are associated with the same vertex. So if $F$ is a spanning forest, let $e$ be an edge in $F$, and $e(\nu)$ be the edge associated with it. Then the straightline multicycle where each cycle is of the form $e \to e(\nu) \to e$ represents $F$ in $G$. 

**Lemma 4.13.** If $\mu$ represents a cyclic permutation on $\mu(E)$, its weight is -1

**Proof.** Lemma 4.12 shows that $\mu$ represents a cycle in $G$. Let $\sigma$ be the permutation on $\mu(E)$. Suppose the chosen orientation on $G$ was such that the arrows travel around the cycle in a uniform direction. For some $e$, let $\nu$ be the vertex it shares with $\sigma(e)$. In $\mu$ there is a cycle $C$ that goes from $e$ to $\sigma(e)^\top$ passing through $\nu$. $C_1$ has the opposite orientation from $C_2$, so the path between the two has weight -1.

So the unassociated weight of this cycle is $(-1)^{||\mu(E)||}$ and the weight is $\text{sgn}(\sigma)(-1)^{||\mu(E)||}$. Note, however, that if $\sigma$ is a cyclic permutation of an even number of edges, $\text{sgn}(\sigma) = -1$. Otherwise, $\text{sgn}(\sigma) = 1$. Thus the weight of this cycle is -1.

Now suppose our cycle has an arbitrary orientation and we reverse the arrow on an edge in the cycle, $e$. Let $v_1$ and $v_2$ be its endpoints. Then in $\mu$, the cycles $\sigma^{-1}(e) \to v_1 \to e^\top$ and $e \to v_2 \to \sigma(e)^\top$ both change weights. Thus changing the orientation does not affect the unassociated weight. It also does not change the sign of $\sigma$ and thus does not change the weight. Since we showed already that for a convenient orientation, $\mu$ has weight -1, all $\mu$ representing cycles have weight -1. 

Thus we obtain our version of Theorem 2 of [6] as follows.
**Theorem 4.14.** The value $\text{tr}(ZZ^T) = \det(I + B^TB)$ is the number of rooted spanning forests of $G$. Furthermore, $\det(Ix + BB^T)$ gives a polynomial where the coefficient of $x^k$ is the number of rooted spanning forests of $G$ with $k$ roots.

**Proof.** Consider a straightline multicycle that represents a spanning forest in $G$. If a multicycle $\mu$ represents a tree in the forest with $t$ edges, there are $t$ cycles in $\mu$. So one of the $t+1$ vertex nodes associated with a vertex in the tree does not have a path going through it. This will be the root of the tree.

Since $\mu$ is straightline, it has weight 1. So every multicycle representing an rooted spanning forest contributes a +1 to the sum. There is exactly one straightline multicycle for every rooted spanning forest and this is the only way to represent a particular rooted spanning forest making use of Lemma 4.12.

Every cycle has four multicycle representations; two have weight 1 and two have weight -1. This will be sufficient for proving the first part of the theorem. Let us consider a cycle in $G$ given by a cyclic permutation, $\sigma$, on $E' = \{e_i\} \subseteq E$, where $\sigma(e_i) = e_{i+1}$. Let $e_i(v)$ be the vertex shared by $e_i$ and $e_{i+1}$. Then we can represent this cycle with multicycles $\{e_i \rightarrow e_i(v) \rightarrow \sigma(e_i)\}$ and $\{e_i \rightarrow \sigma^{-1}(e_i)(v) \rightarrow \sigma^{-1}(e_i)\}$, which will both have weight -1 by Lemma 4.13. These are the only two multicycles representing a cycle which do not induce the identity permutation.

There are exactly two straightline multicycles which represent as cycle: $\{e_i \rightarrow e_i(v) \rightarrow e_i^1\}$ and $\{e_i \rightarrow \sigma^{-1}(e_i)(v) \rightarrow e_i^1\}$. Both of these multicycles have weight 1 by Lemma 4.11. Thus, subgraphs with cycles contribute 0 to the overall sum.

In the above setup, the Laplacian would actually be $Z^T Z$. However, note that in the circuit, we can move the $Z^T$ portion around so that the circuit has $Z^T$ to the left of $Z$. This is to say that the trace of the category satisfies the trace property: $\det(I + XY) = \det(I + YX)$. The multicycles will be the same (albeit deformed) and will still retain the combinatorial meaning discussed above. When considering a minor of $BB^T$ of size $k$, that corresponds to a multicycle in $Z^T Z$ with $k$ paths. We can think of this as activating $k$ vertices in $G$.

We have already shown that $\det(I + B^TB) = \det(I + BB^T)$ is the total number of rooted spanning forests. Now consider

$$\det(Ix + BB^T) = \sum_{k=0}^{V} \left( \sum_{I \subseteq [V]} \sum_{|I|=|V|-k} BB^T_I \right) x^k$$

So a minor of $BB^T$ of size $|V|-k$ is one where exactly $k$ of the vertices don't have a path passing through them. By our correspondence, this means those $k$ vertices are roots in the rooted spanning forest. So indeed this polynomial is a generating function for the number of rooted spanning forests where the exponent denotes the number of roots.

$\square$

Now we discuss how to obtain Kirchoff’s Matrix Tree Theorem as a corollary. This is a statement about the cofactors of the Laplacian of a graph, so we again consider the determinantal circuit $Z^T Z$ with trace $\det(I + BB^T)$.

Suppose we have a determinantal circuit that collapses to a matrix $M$. Then take the determinant of some principal minor $M'$. If $M'$ has labels $I$, this corresponds to the value of activating just the external edges (those that wrap around the circuit) labeled $I$ in the circuit. Then taking $\det(I + M)$ is just summing up the values
over all possible subsets of the external edges. Since the matrix tree theorem deals with cofactors, in our circuit \( Z^\dagger Z \), we will consider multicycles with all edges in the diagram turned on except for one.

**Corollary 4.15. (Kirchoff’s Matrix Tree Theorem)** The number of spanning trees of a graph is the absolute value of any cofactor of its Laplacian.

**Proof.** Consider the absolute value of an arbitrary cofactor of \( Z^\dagger Z \). This will be the value of \( Z^\dagger Z \) will all the edges activated except for one. This corresponds to multicycles where all vertex nodes of our graph are activated except for one. Let the activated vertices be denoted by \( V' \). We only need to consider straightline multicycles as all other multicycles will be canceled out.

Take a given straightline multicycle. For every vertex node, there will be an associated edge node. So if \( |V| \) is the number of vertices in our graph, we have an rooted spanning forest with \( |V| - 1 \) edges. This is a spanning tree. So the number of rooted spanning forests on \( V' \) is the number of spanning trees of our graph. \( \Box \)

### 4.3. Simulating quantum circuits in the presence of closed timelike curves.

Determinantal circuits define a class of tensor networks with a polynomial-time contraction algorithm. An immediate consequence is that certain types of quantum circuits (or more generally tensor networks possibly including preparations and postselection) can be simulated efficiently using this technique. Essentially these are the tensor networks of the type shown in Figure 1 (with arbitrarily many wires and transformations).

The loop in such a circuit corresponds to a postselected closed timelike curve (P-CTC) [14]. The resulting logical category of circuits represent physical experiments (which, if they contain an embedded contradiction, have count zero [17]).

### 5. Relation to Pfaffian circuits

Pfaffian Circuits were introduced as a reformulation of matchcircuits [16, 20]. We present a slightly different definition using category theory. We want to know what the relation of determinant circuits is with respect to Pfaffian circuits. We prefer to do this via functoriality.

We now define the category that gives us Pfaffian circuits. Consider the set \( \mathcal{M} \times \{0, 1\} \), where \( \mathcal{M} \) is the set of labeled skew-symmetric matrices. Furthermore, the columns and rows should have the same labels in the same order. The label sets are subsets of \( \mathbb{N} \). As before, for \( i \in \mathbb{N} \), let \( V_i \equiv \mathbb{C}^2 \) be spanned by an orthonormal basis (with inner product) \( v_{i,0}, v_{i,1} \) and for \( N \subset \mathbb{N} \) write \( V_N := \otimes_{i \in N} V_i \). Now let us consider the following function:

\[
sPf : \mathcal{M} \times \{0, 1\} \rightarrow V_N^{*} \otimes V_N
\]

\[
sPf(M, 0) = \sum_{I \subset \mathbb{N}} Pf(M_I)|I\rangle
\]

\[
sPf(M, 1) = \sum_{I \subset \mathbb{N}} Pf(M_I^*)|I\rangle
\]

where \( |I\rangle = \otimes_{i \in I} v_{i, \chi(i, I)} \), \( \langle J| = \otimes_{i \in J} v_{i, \chi(i, J)}^* \) and the indicator function \( \chi(i, I) = 0 \) if \( i \notin I \) and \( 1 \) if \( i \in I \). We denote by \( M_I \) the principal minor of \( M \) with row and column labels \( I \). \( M_I \) means the principal minor of \( M \) with the rows and columns labeled \( I \).
removed. We will use the convention that \( \text{sPf}(M, 0) \) will be denoted \( \text{sPf}^\circ(M) \) and \( \text{sPf}(M, 1) \) will be denoted \( \text{sPf}^\circ(M) \).

The \( \text{sPf} \) function lets us define a monoidal subcategory of \( \text{Vect}_C \). Let \( \mathcal{P} \) be the free monoidal category defined as follows. The objects are of the form \( V^N \) for ordered subsets of \( N \), the tensor product being the usual one. The morphisms of \( \mathcal{P} \) are generated by elements from the image of \( \text{sPf} \). Composition and tensor product will be inherited from \( \text{Vect}_C \).

**Theorem 5.1.** \( \mathcal{P} \) is a strict monoidal category with daggers.

**Proof.** By our definition of \( \mathcal{P} \), it will be the smallest monoidal subcategory of \( \text{Vect}_C \) containing the generating morphisms with the specified objects. A monoidal category \( (\mathcal{C}, \otimes, \lambda, \rho, \alpha) \) is strict if the natural transformations \( \lambda, \rho, \alpha \) are identities.

It is a theorem that every monoidal category is equivalent to a strict one \[15\]. So we can assume without loss of generality that we are working with a strict category equivalent to \( \text{Vect}_C \) instead. So the \( \alpha, \lambda, \rho \) maps that \( \mathcal{P} \) inherits will be identities. We want to show that the identity morphism is actually generated by our specified morphisms. Consider the following matrix for an object \( A \):

\[
I_A = A \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.
\]

Let \( L_A = \text{sPf}(I_A) = |0_A0_A\rangle + |1_A1_A\rangle \) and \( R_A = \text{sPf}^\circ(I_A) = \langle 0_A0_A| + \langle 1_A1_A| \). Then we can contract these two morphisms along a single edge as in the following picture:

\[
\begin{array}{c}
A \\
L_A \hspace{1cm} R_A \hspace{1cm} A
\end{array}
\]

This gives us the morphism \( |0_A\rangle\langle 0_A| + |1_A\rangle\langle 1_A| \) which is the identity morphism on \( A \). To prove that \( \mathcal{P} \) has daggers, we need to prove a few other things first. \(\square\)

**Definition 5.2.** The anti-transpose of a matrix \( N \), denoted by \( \check{N} \), is \( N \) flipped across the non-standard diagonal.

**Lemma 5.3.** \( \text{Pf}(\check{N}) = \text{Pf}(N) \).

**Proof.** Let \( N = \{\eta_{ij}\} \) be an \( n \times n \) matrix. If \( n \) is odd, the above is trivial, so let \( n \) be even. Now let \( \mathcal{F} \) be the set of partitions of \( [n] \) into pairs, \((i_k,j_k), i_k < j_k \). If \( \pi \in \mathcal{F} \) we can define the sign of \( \pi \), \( \text{sgn}(\pi) \). This is done by considering the set \( [n] \) as a sequence of nodes laid out horizontally and labeled \( 1, \ldots, n \) from left to right. Then if two nodes are paired in \( \pi \), connect them with an edge. Then \( \text{sgn}(\pi) \) is \((-1)^k \) where \( k \) is the number of places where lines cross. Now we can define \( \text{Pf}(N) \) as follows:

\[
\text{Pf}(N) = \sum_{\pi \in \mathcal{F}} \text{sgn}(\pi) \prod_{(i_k,j_k) \in \pi} \eta_{i_k,j_k}.
\]

Now let \( \eta'_{ij} = \eta_{n-j+1,n-i+1} \) be the entries of \( \check{N} \) and suppose \( \pi \in \mathcal{F} \). Then the mapping \( \mathcal{F} \to \mathcal{F} : \pi \mapsto \pi' \) given by \((i_k,j_k) \mapsto (n-j_k+1, n-i_k+1) \) is a bijective involution. Note that \( \pi' \) is the matching formed from \( \pi \) by relabeling the nodes as
n, . . . , 1 from left to right. This preserves the number of crossings of edges so that sgn(π') = sgn(π). Thus we get
\[ \text{Pf}(\hat{N}) = \sum_{\pi \in \mathcal{F}} \text{sgn}(\pi) \prod_{(i_k, j_k)} \eta_{i_k, j_k} = \sum_{\pi' \in \mathcal{F}} \text{sgn}(\pi') \prod_{(n-j_k+1, n-i_k+1)} \eta_{n-j_k+1, n-i_k+1} = \text{Pf}(N). \]

\[ \square \]

**Definition 5.4.** If I is a bitstring, let \( \hat{I} \) be the bitstring reflected across a vertical axis. If \( I \in \mathbb{N} \), \( \hat{I} \) is formed by considering \( I \) as a bitstring representing a characteristic function. Then \( \hat{I} \) is a characteristic function defining another subset of \( \mathbb{N} \). Then \( |\hat{I}|^y = \sum_{i,j} v_{i,j}(i,j) \) and \( \langle \hat{I} \rangle = \sum_{i,j} v^*_{i,j}(i,j) \)

**Corollary 5.5.** Let \( N \) be a skew symmetric matrix with labels \( M \). Let \( \hat{N} \) also have labels \( M \). Then \( \text{sPf}(\hat{N}) = \sum_{I \in M} \text{Pf}(N_I) |\hat{I}| \)

**Proof.** Let \( I \in M \). Note that \( N_I = \hat{N}_I \). Then \( \text{Pf}(N_I) = \text{Pf}(\hat{N}_I) \). This gives the result. \( \square \)

**Example 5.6.** Consider the following matrix:
\[
N = \begin{pmatrix}
0 & 0 & a & 0 \\
0 & 0 & 0 & b \\
-a & 0 & 0 & 0 \\
0 & -b & 0 & 0
\end{pmatrix}
\]

sPf(\( \hat{N} \)) = |0000\rangle + b|1010\rangle + a|0101\rangle - ab|1111\rangle

= Pf(\( N_{\emptyset} \))|0000\rangle + Pf(\( N_{\{2, 4\}} \))|1010\rangle + Pf(\( N_{\{1, 3\}} \))|0101\rangle + Pf(\( N \))|1111\rangle

= \sum_{I \in M} \text{Pf}(N_I) |\hat{I}|.

**Proposition 5.7.** For any skew-symmetric matrix \( M \),
\[
\sum_I \text{Pf}(M_I) |I| \quad \sum_I \text{Pf}(M_I) |\hat{I}|
\]
are morphisms of \( \mathcal{P} \). This implies that \( \mathcal{P} \) is a dagger monoidal category.

**Proof.** Let \( M \) have labels \( A = \{A_1, \ldots, A_n\} \). Then \( \hat{M} \) will have labels \( \hat{A} = \{A_n, \ldots, A_1\} \). Let \( R_A \) be defined as:
\[
R_A = \text{sPf} \hat{A} \begin{pmatrix}
\hat{A} & A \\
0 & \hat{i} & \hat{I} & 0
\end{pmatrix}
\]
where \( \hat{I} \) is the identity matrix reflected over a vertical axis. Then consider the following morphism in \( \mathcal{P} \):
This diagram represents the morphism
\[
\left( \sum_{i \in A} \text{Pf}(M_i) |i| \right) \left( \sum_{i \in \{AA\}} \langle i|l \rangle \right) = \sum_{i \in A} \text{Pf}(M_i) |i|.
\]

We can similarly form \( \sum \text{Pf}(M_i) |i| \) by instead using \( s\text{Pf}^\tau(M) \) and \( s\text{Pf}(R_A) \). Now since every generating morphism has a dagger, the entire category has a dagger and it is the usual vector space dagger. \( \square \)

Note that there are two primary types of morphisms in \( \mathcal{P} \), namely those of the form \( s\text{Pf}(M) \) and those of form \( s\text{Pf}^\tau(M) \). We will see the diagrams form bipartite graphs.

Suppose we are given a Pfaffian circuit \( \Gamma \). Let \( \Xi \) be the morphisms of the form \( s\text{Pf}(M) \) and \( \Theta \) be the morphisms of the form \( s\text{Pf}^\tau(M) \). We define \( \Xi' \) and \( \Theta' \) likewise. \( \Xi' \) is the direct sum with the row and columns reordered as follows: The ordering is found by drawing a planar curve through the Pfaffian circuit such that every edge is intersected by the curve once and exactly once. Since a Pfaffian circuit is planar and bipartite, such a curve always exists and the result is independent of the choice of curve. The edges are then labeled based on when the curve intersects them. This is ordering used to define \( \Xi' \). \( \Theta' \) is defined to be \( \{(-1)^{i+j+1} \theta_{ij}\} \).

**Theorem 5.8.** The value of a Pfaffian circuit \( \Gamma \) is given by \( \text{Pf}(\Xi + \tilde{\Theta}) \)\(^{10}\)

Thus Pfaffian circuits can be computed in polynomial time. Now we seek a functor transforming determinantal circuits into Pfaffian circuits. Such a functor should preserve the trace so that the resulting Pfaffian circuit solves the same problem as the original determinantal circuit. The functor should also be faithful.

Given a morphism in \( \mathcal{D} \), we now construct it in \( \mathcal{P} \). For an \( n \times n \) matrix \( M \),
\[
\text{Pf} \begin{bmatrix} 0 & M \\ -M^T & 0 \end{bmatrix} = (-1)^{n(n-1)/2} \det(M).
\]

We must embed a matrix \( M \) into a skew-symmetric matrix such that the Pfaffian corresponds to the determinant. Define \( \tilde{M} \) as the matrix \( M \) reflected across a vertical axis, and define \( S(M) \) to be
\[
S(M) = \begin{bmatrix} 0 & \tilde{M} \\ -\tilde{M}^T & 0 \end{bmatrix}.
\]

**Proposition 5.9.** For an \( n \times n \) matrix \( M \),
\[
\text{Pf}(S(M)) = \text{Pf} \begin{bmatrix} 0 & \tilde{M} \\ -\tilde{M}^T & 0 \end{bmatrix} = \det(M).
\]
Proof. In general, \( \tilde{M} \) can be made from \( M \) with \( \left\lfloor \frac{n}{2} \right\rfloor \) column swaps. So if \( n \equiv 0, 1 \mod 4 \), then \( \text{Pf}(S(M)) = (-1)^{n(n-1)/2} \det(M) = \det(M) \). If \( n \) is congruent to 2 or 3 modulo 4, then \( \left\lceil \frac{n}{2} \right\rceil \) is an odd number so \( \det(M) = - \det(M) \) and \( \text{Pf}(S(M)) = (-1)^{n(n-1)/2} \det(M) = - \det(M) = \det(M) \).

\[ \square \]

Theorem 5.10. Every morphism in \( D \) is a morphism in \( P \). Thus there is a trace-preserving faithful strict monoidal functor from \( D \to P \) given by inclusion.

Proof. First suppose that \( M \) is an \( n \times n \) matrix. The labels of \( S(M) = R \cup \tilde{C} \) where \( R \) is the row labels of \( M \) and \( C \) are the column labels of \( M \). Now let \( K \) be a subset of the labels. Then let \( I = K \cap R \) and \( \tilde{J} = K \cap \tilde{C} \). Then we get

\[
\text{Pf}(S(M)_K) = \text{Pf} \left[ \begin{array}{cc}
0 & \tilde{M}_{I,J} \\
\tilde{M}_{I,J} & 0
\end{array} \right] = \det(M_{I,J}),
\]

so that

\[
\text{sPf}(S(M)) = \sum_{I \subseteq R, J \subseteq C} \det(M_{I,J}) |I\rangle \langle \tilde{J}|,
\]

\[
\text{sPf}^\vee(S(M)) = \sum_{I \subseteq R, J \subseteq C} \det(M_{I,J}) |I\rangle \langle J|.\]

The identity morphism on \( A_n \otimes \cdots \otimes A_1 \) in \( C \) is given by the matrix

\[
I \otimes A_1 = \begin{pmatrix}
A_n & A_{n-1} & \cdots & A_1 \\
A_n & 0 & \cdots & 0 \\
A_{n-1} & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots \\
A_1 & 0 & \cdots & 1
\end{pmatrix}.
\]

Suppose we have an \( n \times n \) matrix \( M : B_1 \otimes \cdots \otimes B_n \to A_1 \otimes \cdots \otimes A_n \). Then we define \( M_* = \text{sPf}(S(M)) \) and \( R \otimes A_1 = \text{sPf}^\vee(S(I \otimes A_1)) \). Let us consider the morphism in \( P \) given by

\[
\begin{array}{c}
B_1 \\
B_n \\
\vdots \\
R \otimes A_1 \\
M_* \\
A_n \\
\vdots \\
A_1 \\
A_n
\end{array}
\]

For \( I \subseteq \{B_1, \ldots, B_n\}, \tilde{J}, \tilde{J}' \subseteq \{A_n, \ldots, A_1\} \); and \( J' \subseteq \{A_1, \ldots, A_n\} \), we can represent this tensor as

\[
\left( \sum \det(M_{I,J}) |I\rangle \langle \tilde{J}| \right) \left( \sum |\tilde{J}'\rangle \langle J'| \right) = 
\sum \det(M_{I,J}) |I\rangle \langle J| = \text{sDet}(M).
\]
So for any square matrix $M$, $\text{sDet}(M)$ is a morphism in $\mathcal{P}$. Now not every morphism in $\mathcal{C}$ is a square matrix. However, if we have an $n \times m$ matrix $M$, we can make it square. If $n < m$, then let $M' = M \oplus Z_{m-n}$ where $Z_{m-n}$ is the $(m-n) \times 0$ matrix. If $m < n$, then let $M' = M \oplus Z'_{n-m}$ where $Z'_{n-m}$ is the $0 \times (n-m)$ matrix. What this amounts to is either adding rows or columns of zeros as needed.

Now note that $\text{sPf}([0]) = |0\rangle$. $|0\rangle$ is also a morphism in $\mathcal{P}$. Consider $\text{sPf}'(K) = \langle 0_A0_B | + \langle 1_A1_B |$ where

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

and contracting this with the morphism $|0_B\rangle$, we obtain $\langle 0_A |$.

Let $M$ be an arbitrary $n \times m$ matrix. Then let us consider $S(M')$ where $M'$ is defined as above. Suppose $n < m$. Then

$$\text{sPf}(S(M')) = \sum_{I,J} \det(M_{I,J}) |I0_{n+1} \cdots 0_m\rangle \langle J|$$

Consider the following diagram in $\mathcal{P}$:

$$\begin{array}{c}
1 \\
\vdots \\
\langle 0 | \\
\langle n | \\
\langle 0 | \\
\vdots \\
\text{sPf}(S(M')) \\
\vdots \\
\langle 0 | \\
\langle m | \\
1 \\
\end{array}$$

The morphism this represents will obviously come out to be $\text{sDet}(M)$. If $n > m$, then copies of $|0\rangle$ are added to the extra output wires of $\text{sPf}(S(M'))$. Thus we have finished the proof of theorem. Every morphism of $\mathcal{D}$ is in fact a morphism in $\mathcal{P}$. Furthermore, the reinterpretation of a determinantal circuit as a Pfaffian circuit can obviously be done in polynomial time.

Despite this fact, determinantal circuits still have some advantages. If a Pfaffian circuit can be represented as a determinantal circuit, its evaluation will be more efficient. Also, given the non-intuitive nature of the inclusion, viewing a circuit as determinantal may prove more useful than viewing it as a Pfaffian circuit. In the applications above, this is certainly true. Finally, the practical complexity of implementation is lessened by the determinantal approach since there is less need to track a complex ordering on the objects.

**Acknowledgments**

J.M. and J.T. were supported in part by the Defense Advanced Research Projects Agency under Award No. N66001-10-1-4040. Portions of J.T.'s work were sponsored by the Applied Research Laboratory's Exploratory and Foundational Research Program.

**References**

[1] S. Arora and B. Barak. *Computational complexity: a modern approach*. Cambridge University Press, 2009.

[2] V. Bergholm and J.D. Biamonte. Categorical quantum circuits. *Journal of Physics A: Math. Theor.*, 44:245304, 2011.
[3] W. Bruns, A. Conca, and M. Varbaro. Relations between the minors of a generic matrix. Arxiv preprint arXiv:1111.7263, 2011.
[4] A. Bulatov. The complexity of the counting constraint satisfaction problem. Automata, Languages and Programming, pages 646–661, 2010.
[5] J.Y. Cai and X. Chen. Complexity of counting CSP with complex weights. In Proceedings of the 44th symposium on Theory of Computing, pages 909–920. ACM, 2012.
[6] F. R. K. Chung and Robert P. Langlands. A combinatorial Laplacian with vertex weights. J. Combin. Theory Ser. A, 75(2):316–327, 1996.
[7] C. Damm, M. Holzer, and P. McKenzie. The complexity of tensor calculus. Computational Complexity, 11(1):54–89, 2003.
[8] R.A. Horn and C.R. Johnson. Matrix analysis. Cambridge University Press, 1990.
[9] A. Joyal and R. Street. The geometry of tensor calculus. I. Advances in Mathematics, 88(1):55–112, 1991.
[10] A. Joyal, R. Street, and D. Verity. Traced monoidal categories. In Mathematical Proceedings of the Cambridge Philosophical Society, volume 119, pages 447–468. Cambridge University Press, 1996.
[11] C. Kassel. Quantum groups, volume 155. Springer, 1995.
[12] Y. Lafont. Towards an algebraic theory of Boolean circuits. Journal of Pure and Applied Algebra, 184(2):257–310, 2003.
[13] JM Landsberg, J. Morton, and S. Norine. Holographic algorithms without matchgates. Linear Algebra and its Applications, 438(15), 2013.
[14] S. Lloyd, L. Maccone, R. Garcia-Patron, V. Giovannetti, Y. Shikano, S. Pirandola, L.A. Rozema, A. Darabi, Y. Soudagar, L.K. Shalm, and A. Steinberg. Closed timelike curves via postselection: theory and experimental test of consistency. Physical Review Letters, 106(4):40403, 2011.
[15] S. Mac Lane. Categories for the working mathematician. Springer verlag, 1998.
[16] J. Morton. Pfaffian circuits. Arxiv preprint arXiv:1101.0129, 2010.
[17] J. Morton and J. Biamonte. Undecidability in tensor network states. Physical Review A, 86(3):030301, 2012.
[18] K.D. Mulmuley and M. Sohoni. Geometric complexity theory i: An approach to the P vs. NP and related problems. SIAM Journal on Computing, 31(2):496–526, 2001.
[19] P. Selinger. A survey of graphical languages for monoidal categories. New Structures for Physics, pages 275–337, 2009.
[20] L. Valiant. Quantum circuits that can be simulated classically in polynomial time. SIAM J. Comput., 31(4):1229–1254, 2002.

1Department of Mathematics, Pennsylvania State University, University Park PA 16802