

The Pennsylvania State University

The Graduate School

The Eberly College of Science

THE INVARIANT THEORY AND GEOMETRY PERTAINING TO  
TENSOR NETWORKS AND SOME FURTHER APPLICATIONS

A Dissertation in

Mathematics

by

Jacob W. Turner

© 2015 Jacob W. Turner

Submitted in Partial Fulfillment  
of the Requirements  
for the Degree of

Doctor of Philosophy

May 2015

The dissertation of Jacob W. Turner was reviewed and approved\* by the following:

Jason Morton  
Assistant Professor of Mathematics  
Dissertation Adviser  
Chair of Committee

A. Kirsten Eisenträger  
Associate Professor of Mathematics

Carina Curto  
Associate Professor of Mathematics

Sean Hallgren  
Professor of Computer Science

Yuxi Zheng  
Professor of Mathematics  
Head of the Department of Mathematics

\* Signatures are on file in the Graduate School.

## Abstract

The main objects of study in this work are tensor networks. We study applications of these objects to problems in computer science and physics using methods from algebraic geometry, representation theory, and geometric invariant theory. The main results are:

- (1) Descriptions of several classes of tensor networks that can be efficiently contracted and some counting problems that they can model.
- (2) A classification of the invariant ring of a product of groups acting by conjugation, in particular the adjoint action of

$$\times_{i=1}^n \mathrm{GL}(V_i) \curvearrowright \left( \bigotimes_{i=1}^n \mathrm{End}(V_i) \right)^{\oplus m}.$$

- (3) A sufficient condition for a point in  $\mathrm{End}(V)^{\oplus m}$  to have a Zariski closed orbit under this action.
- (4) Applications to the study of quantum entanglement on density operators.

# Contents

|   |            |
|---|------------|
| <b>List of Figures</b>  | <b>vi</b>  |
| <b>Acknowledgements</b>   | <b>vii</b> |
| <b>1 Introduction</b>   | <b>1</b>   |
| 1.1 Bra-Ket Notation . . . . .  | 2          |
| <b>2 An Introduction to Monoidal Categories and Tensor Networks</b>                               | <b>4</b>   |
| 2.1 Monoidal Categories . . . . .   | 4          |
| 2.1.1 Types of Monoidal Categories . . . . .  | 6          |
| 2.1.2 Functors Between Monoidal Categories . . . . .  | 7          |
| 2.1.3 Diagrammatic Language for Monoidal Categories . . . . .                                     | 9          |
| 2.2 Tensor Networks . . . . .   | 11         |
| <b>3 The Complexity of Contracting Tensor Networks</b>  | <b>14</b>  |
| 3.1 Modeling #SAT Problems as Tensor Networks . . . . .   | 15         |
| 3.1.1 Complexity Theory Via Tensor Networks . . . . .   | 17         |
| 3.2 Criteria for Tractability of #SAT Problems . . . . .  | 18         |
| 3.2.1 Some SAT Instances That Are Always Satisfiable . . . . .                                    | 19         |
| 3.2.2 Efficient Tensor Contraction Algorithm for Boolean States With Restricted Fan-out . . . . . | 21         |
| 3.2.3 Relation to Other Algorithms . . . . .  | 24         |
| 3.2.4 Rényi Entropy and Complexity . . . . .  | 25         |
| <b>4 Determinantal, Pfaffian, and Match Circuits</b>  | <b>28</b>  |
| 4.1 Pfaffian circuits . . . . .   | 29         |
| 4.2 Determinantal Circuits . . . . .  | 34         |
| 4.3 Determinantal Circuits are Pfaffian Circuits . . . . .  | 43         |

|          |  |            |
|----------|--|------------|
| <b>5</b> | <b>The Combinatorics of Determinantal Circuits</b>                                   | <b>47</b>  |
| 5.1      | Equivalence Classes of Multicycles . . . . .   | 47         |
| 5.2      | Recovering the matrix tree theorem . . . . .   | 50         |
| 5.3      | Computing the Tutte Polynomial of Lattice Path Matroids .                            | 52         |
| 5.3.1    | Weighted Lattice Paths . . . . .   | 53         |
| 5.3.2    | Phrasing the Tutte Polynomial as a Determinantal<br>Circuit . . . . .                | 55         |
| 5.3.3    | Lattice Path Matroids . . . . .  | 56         |
| 5.3.4    | Algorithm for Computing the Tutte Polynomial . . . .                                 | 57         |
| <b>6</b> | <b>An Introduction to Invariant Theory</b>   | <b>60</b>  |
| 6.1      | Centralizers of Semisimple Algebras . . . . .  | 62         |
| 6.2      | The Categorical Quotient . . . . .   | 65         |
| 6.3      | An Extended Example: The Adjoint Action of $GL(V)$ on<br>$End(V)$ . . . . .          | 67         |
| <b>7</b> | <b>The Invariants of Local Conjugation</b>   | <b>70</b>  |
| 7.1      | Multilinear Invariants of Local Groups . . . . .                                     | 72         |
| 7.1.1    | Description of the Invariant Ring . . . . .  | 74         |
| 7.2      | Tensor Network Representations of Invariants . . . . .                               | 75         |
| 7.3      | Bounding the Degree of the Generators . . . . .                                      | 76         |
| 7.3.1    | Girth at most $(2, \dots, 2)$ invariants operating on rank-<br>one tensors . . . . . | 78         |
| 7.3.2    | General girth, rank-one tensors . . . . .  | 82         |
| 7.3.3    | General case, arbitrary tensors . . . . .  | 84         |
| <b>8</b> | <b>A Complete Set of Invariants for Local Unitary Equivalence</b>                    | <b>85</b>  |
| 8.1      | Closed Orbits . . . . .  | 86         |
| 8.2      | Description of the Null Cone . . . . .   | 91         |
| 8.3      | The Relations Among the Invariants . . . . .   | 95         |
| <b>9</b> | <b>Applications to the Study of Quantum Entanglement</b>                             | <b>99</b>  |
| 9.1      | Example . . . . .  | 100        |
|          | <b>Bibliography</b>  | <b>102</b> |

# List of Figures

|     |  |    |
|-----|--|----|
| 2.1 | Basic String Diagrams . . . . .  | 10 |
| 2.2 | Further String Diagrams . . . . .  | 11 |
| 3.1 | Tensor Networks the Count the Number of Satisfying Solutions for a Boolean Formula . . . . .   | 16 |
| 3.2 | A Boolean Function Expressible as a Tree. . . . .  | 19 |
| 4.1 | Viewing $\mathcal{P}$ as a Traced Category. . . . .  | 34 |
| 4.2 | 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 $\text{sDet}$ to each matrix. Letting $V = \mathbb{C}^2$ , they lie in $(V^*)^{\otimes 2} \otimes V^{\otimes 2}$ , $(V^*)^{\otimes 2} \otimes V$ , $V^* \otimes V^{\otimes 2}$ , and $(V^*)^{\otimes 2} \otimes V^{\otimes 2}$ respectively. . . . . | 42 |
| 5.1 | Transforming a rooted graph to a determinantal circuit. . . .  | 51 |
| 5.2 | Lattices . . . . .   | 53 |
| 5.3 | A Determinantal Circuit . . . . .  | 55 |
| 5.4 | Weighted lattice with Tutte polynomial $(x^2 + xy + y^2 + x + y)(x)(x + y + y^2)$ as its value. . . . .  | 56 |
| 7.1 | The network on the left factors as $\text{Tr}_{(12),(23)}^{1,2,1} = \text{Tr}_{(12),(12)}^{1,2} \text{Tr}_{(),()}^1$ into generators of smaller degree, while the network $\text{Tr}_{(12),(23)}^{2,1,1}$ on the right does not factor. . . . .  | 77 |
| 7.2 | For $M = \{1, 1, 1, 2, 2, 3\}$ and $\sigma_i = (m_1)(m_2)(m_3 m_4)(m_5 m_6)$ . . .   | 80 |

## Acknowledgements

This work would not have been possible without the help and support of several people, only a few of whom it is possible to mention here. I owe a great deal to my adviser, Jason Morton, who was a source of constant encouragement and always willing to take chances with me. I was able to follow my own interests and learn so much in the writing of this Dissertation. I also want to thank Jacob Biamonte, for allowing me to come work with him in Italy, giving me good research problems, and keeping life interesting. I would also like to thank Becky Halpenny, Flossie Dunlop, Erica Workman, and Amy Stover for helping me keep my affairs in order as I cavorted around the world and navigating any administrative/bureaucratic nightmares that I unwittingly stumbled into. To my friends and family, I can't begin to describe how essential you've been to me. The past few years would have been an absolute misery if not down right impossible without you. You've seen me through a lot and I am eternally grateful.

# Chapter 1

## Introduction

In 1971, Roger Penrose introduced what he called “spin networks” to model interactions of particles in quantum systems [84]. These models were given as graphs where edges were systems of particles and vertices represented events or relations among different systems. Later formalization came to associate vector spaces to edges and tensors to vertices. These models have been invented a few different times with different names including birdtracks ([23]) and the term by which they are now known: tensor networks.

Tensor networks have seen applications to various different aspects of physics over the years. They have very recently been used to model ground states of Hamiltonians and quantum systems more generally [104, 53, 35, 33]. They can also represent channels, maps, states and processes appearing in quantum theory [40, 41, 42, 11]. They also have applications in the study of quantum gravity [93, 5].

Tensor networks have showed up independently in several other disciplines. In complexity theory, tensor networks can be viewed as a generalized notion of circuit. The tensors associated to vertices are the analog of logic gates in classical circuits. As an example, all quantum circuits can be interpreted as tensor networks. A different class of tensor networks were given by Leslie Valiant, who restricted the allowed tensors to lie in a particular variety. He was able to show any circuit built up from this restricted class of gates could be evaluated in polynomial time [98, 63, 79, 78].

In algebraic statistics, tensor networks appear under the name of “graphical models”. Graphical models are a tool for modeling dependencies among random variables. An active area of research is to understand which tensors can be modeled using a particular graphical model. These models give a polynomial parameterization of sets of tensors and the motivating problem



is to find the ideal of the image of this parameterization. A famous example is the problem posed by E. Allman (now solved set-theoretically) to determine the equations defining  $\sigma_4(\mathbb{P}^3 \times \mathbb{P}^3 \times \mathbb{P}^3)$ , which arose in the area of phylogenetics, specifically by looking at representations of graphs modeling phylogenetic trees [2, 37, 7, 29].

Tensor networks are a particular case of a diagrammatic language for a monoidal category [96, 55, 54]. Every monoidal category has a corresponding diagrammatic language and tensor networks arise from considering the category of finite dimensional vector spaces over some field. Studying monoidal categories more generally allowed for many more mathematical questions to be treated in a fashion similar to tensor networks.

In fact, studying representations of monoidal categories have proven to be very useful. By a representation of a monoidal category, we mean a functor from said category into the category of finite dimensional vector spaces. Examples include topological quantum field theory, knot invariants, typed graphs, and many algebraic combinatorial models such as edge and vertex coloring models.

One of the interesting aspects of tensor networks is that they possess a lot of internal symmetry. Tensor networks admit natural actions of products of general linear groups under which the tensor they represent is fixed. In physics, these are sometimes called *gauge symmetries* on graphs [5]. These problems naturally lend themselves to techniques from invariant theory and algebraic geometry and are deeply related to work done by Hilbert, Noether, Mumford, Procesi, Kraft, Popov, and Derksen, to name a few.

In this dissertation, we focus on two main aspects of tensor networks. Chapters 4 and 5 treat tensor networks as generalizations of circuits. We discuss tensor networks that can be evaluated efficiently and some applications to known counting problems. In Chapters 7 and 8, we discuss the invariant theory behind the internal symmetries of tensor networks. We also discuss how the invariant theory relates to understanding quantum entanglement.

## 1.1 Bra-Ket Notation

Throughout this dissertation, we use the so called bra-ket notation for tensors that is common in physics, but less so in mathematics. We briefly explain this notation here.

Given vector spaces  $V_1, \dots, V_{r+s}$ , let us consider the vector space  $V_1^* \otimes \dots \otimes V_r^* \otimes V_{r+1} \otimes \dots \otimes V_{r+s}$ , where each  $V_i$  has a specified orthonormal basis  $v_{i,1}, \dots, v_{i,d_i}$ , where  $d_i = \dim(V_i)$ . Let  $v_{i,j}^*$  be the associated dual basis

element to  $v_{i,j}$ . We then express the vector

$$v_{1,i_1}^* \otimes \cdots \otimes v_{r,i_r}^* \otimes v_{r+1,i_{r+1}} \otimes \cdots \otimes v_{r+s,i_{r+s}} =: |i_1 \cdots i_r \rangle \langle i_{r+1} \cdots i_{r+s}|$$

where  $i_1 \cdots i_r$  and  $i_{r+1} \cdots i_{r+s}$  are numeric strings of indices. Sometimes, when the vector spaces involved are not clear, the indices in the string may have a subscript indicating which vector space it is associated to. For example, considering the vector space  $A \otimes B = (\mathbb{C}^2)^* \otimes \mathbb{C}^2$  with an orthonormal basis  $v_0, v_1$ , the vector  $v_1^* \otimes v_0$  be written as  $|1_A \rangle \langle 0_B|$  or simply  $|1 \rangle \langle 0|$  if the vector spaces involved are clear.

Taking the tensor product of two vectors is straightforward, simply add indices in the appropriate places in the numeric strings. We denote the bilinear pairing between  $V_1 \otimes \cdots \otimes V_n$  and  $V_1^* \otimes \cdots \otimes V_n^*$  by

$$\langle i_{1,V_1} \cdots i_{n,V_n} | j_{1,V_1} \cdots j_{n,V_n} \rangle := v_{1,i_1}^*(v_{1,j_1}) \cdots v_{n,i_n}^*(v_{n,j_n}).$$

## Chapter 2

# An Introduction to Monoidal Categories and Tensor Networks

In this chapter, we introduce the formalism behind the diagrammatic languages that we use throughout this work, in particular, tensor networks. We first describe what a monoidal category is as well as axiomatizing several special types of monoidal categories.

### 2.1 Monoidal Categories

A monoidal category is one where a notion of product exists both for objects and morphisms and is typically thought of as a tensor product. There also exists a special object which acts like a unit for this product, hence the term “monoidal”.

**Definition 2.1.** A monoidal category,  $\mathcal{C}$ , is a tuple  $(\text{Ob}(\mathcal{C}), \text{Hom}(\mathcal{C}), \otimes, \alpha, \lambda, \rho, \mathbf{1})$  where

1.  $\text{Ob}(\mathcal{C})$  is a class of objects and  $\text{Hom}(\mathcal{C})$  a class of morphisms.
2. A bifunctor  $\otimes : \mathcal{C} \times \mathcal{C} \rightarrow \mathcal{C}$  such that
  - $\text{id}_A \otimes \text{id}_B = \text{id}_{A \otimes B}$  for all  $A, B \in \text{Ob}(\mathcal{C})$  and
  - $(f \otimes g) \circ (h \otimes i) = (f \circ h) \otimes (g \circ i)$  for  $f \in \text{Hom}(A, B)$ ,  $h \in \text{Hom}(B, C)$ ,  $g \in \text{Hom}(D, E)$ , and  $i \in \text{Hom}(E, F)$ .

3. A natural isomorphism  $\alpha_{ABC} : (A \otimes B) \otimes C \rightarrow A \otimes (B \otimes C)$  for all  $A, B, C \in \text{Ob}(\mathcal{C})$ .
4. Natural isomorphisms  $\rho_A : (A \otimes \mathbb{1}) \rightarrow A$  and  $\lambda_A : (\mathbb{1} \otimes A) \rightarrow A$ , for all  $A \in \text{Ob}(\mathcal{C})$ .

**Theorem 2.2** (MacLane's Coherence Theorem [73]). *A tuple  $(\text{Ob}(\mathcal{C}), \text{Hom}(\mathcal{C}), \otimes, \alpha, \lambda, \rho, \mathbb{1})$  is a monoidal category if and only if  $\mathcal{C}$  is a category and the following diagrams commute for all  $A, B, C, D \in \text{Ob}(\mathcal{C})$ .*

$$\begin{array}{ccc}
(A \otimes \mathbb{1}) \otimes B & \xrightarrow{\alpha} & A \otimes (\mathbb{1} \otimes B) \\
& \searrow \rho_A \otimes \text{id}_B & \downarrow \text{id}_A \otimes \lambda_B \\
& & A \otimes B
\end{array}$$
  

$$\begin{array}{ccc}
((C \otimes A) \otimes B) \otimes D & \xrightarrow{\alpha \otimes \text{id}_D} & (C \otimes (A \otimes B)) \otimes D \\
& \searrow \alpha & \downarrow \alpha \\
(C \otimes A) \otimes (B \otimes D) & & C \otimes ((A \otimes B) \otimes D) \\
& \searrow \alpha & \uparrow \text{id}_C \otimes \alpha \\
& & C \otimes (A \otimes (B \otimes D))
\end{array}$$

There are many important examples of monoidal categories and their study has applications to many different areas. In this dissertation, we are primarily interested the category of finite dimensional vector spaces over a field  $k$ ,  $\text{Vect}_k$ , or more generally subcategories that are also monoidal. The category  $\text{Vect}_k$  is monoidal when  $\otimes$  is defined by the usual tensor product on vector spaces,  $\mathbb{1} = k$ , and the natural isomorphisms  $\alpha, \lambda$ , and  $\rho$  are the usual ones.

**Example 2.3.** Some other important examples of monoidal categories are

- **FinRel**, the category of finite sets and relations, where  $\otimes$  is defined by cartesian product and  $\mathbb{1} = \{\emptyset\}$ .
- **Top**, the category of topological spaces, where  $\otimes$  is the cartesian product and  $\mathbb{1}$  is the 1-point space.
- **Braid**, the category of braids, where  $\otimes$  is the disjoint union of braids and  $\mathbb{1}$  is the empty braid.

- $\mathbf{Cat}$ , the category of locally small categories, where  $\otimes$  is the product of categories and  $\mathbf{1}$  is the category with only one object and the identity.

There are several other additional structures that may be present in a monoidal category. For example, note that in the Definition 2.1, there is no relationship between  $A \otimes B$  and  $B \otimes A$ . However, in all the examples listed above, these two objects are isomorphic. We present now the types of structures that will be relevant for this dissertation.

### 2.1.1 Types of Monoidal Categories

As we already pointed out, it is often natural for a monoidal category  $\mathcal{C}$  to have the property that  $A \otimes B$  is naturally isomorphic to  $B \otimes A$  for all  $A, B \in \text{Ob}(\mathcal{C})$ . Let  $\sigma_{AB} : A \otimes B \rightarrow B \otimes A$  be a natural isomorphism. Then  $\mathcal{C}$  is called a *braided* monoidal category if the following two diagrams commute for all  $A, B, C \in \text{Ob}(\mathcal{C})$ .

$$\begin{array}{ccccc}
& (B \otimes A) \otimes C & \xrightarrow{\alpha} & B \otimes (A \otimes C) & \\
\sigma_{A,B} \otimes \text{id}_C \nearrow & & & & \searrow \text{id}_B \otimes \sigma_{A,C} \\
(A \otimes B) \otimes C & & & & B \otimes (C \otimes A) \\
& \searrow \alpha & & \nearrow \alpha & \\
& A \otimes (B \otimes C) & \xrightarrow{\sigma_{A,(B \otimes C)}} & (B \otimes C) \otimes A &
\end{array}$$
  

$$\begin{array}{ccccc}
& (B \otimes A) \otimes C & \xrightarrow{\alpha} & B \otimes (A \otimes C) & \\
\sigma_{A,B}^{-1} \otimes \text{id}_C \nearrow & & & & \searrow \text{id}_B \otimes \sigma_{A,C}^{-1} \\
(A \otimes B) \otimes C & & & & B \otimes (C \otimes A) \\
& \searrow \alpha & & \nearrow \alpha & \\
& A \otimes (B \otimes C) & \xrightarrow{\sigma_{A,(B \otimes C)}^{-1}} & (B \otimes C) \otimes A &
\end{array}$$

Furthermore, if  $\sigma^{-1} = \sigma$ , then the category is called *symmetric*.

Since we are interested in the category  $\mathbf{Vect}_k$ , we wish to also axiomatize the notions of dual vector spaces, traces, and adjoints of linear maps. These correspond to autonomous, traced, and dagger categories respectively.

In an *autonomous category*, for every  $A \in \text{Ob}(\mathcal{C})$ , there exists left and right duals of  $A$  denoted  ${}^*A$  and  $A^*$ , respectively. Furthermore, there are

maps  $\eta_A : \mathbb{1} \rightarrow A^* \otimes A$ ,  $\eta'_A : \mathbb{1} \rightarrow A \otimes^* A$  and  $\epsilon_A : A \otimes A^* \rightarrow \mathbb{1}$ ,  $\epsilon'_A : {}^*A \otimes A \rightarrow \mathbb{1}$  so that

$$(\eta_A \otimes \text{id}_A) \circ (\text{id}_A \otimes \epsilon_A) = \text{id}_A. \quad (2.1a)$$

$$(\text{id}_{{}^*A} \otimes \eta'_A) \circ (\epsilon'_A \otimes \text{id}_{{}^*A}) = \text{id}_{{}^*A}. \quad (2.1b)$$

For all categories we consider,  ${}^*A = A^*$ . This will be a consequence of our categories being symmetric. Often, instead of calling a monoidal category autonomous, we say it has *duals for objects*. A symmetric autonomous category is often called a *closed compact category*.

For a *traced monoidal category*, we have maps  $\text{Tr}_C : \text{Hom}(A \otimes C, B \otimes C) \rightarrow \text{Hom}(A, B)$ . In the case of  $\text{Vect}_k$ , this operation corresponds to a partial trace over  $C$ . This function must satisfy the following four axioms:

1.  $\text{Tr}_C((g \otimes \text{id}_C) \circ f \circ (h \otimes \text{id}_C)) = g \circ \text{Tr}_C(f) \circ h$ .
2.  $\text{Tr}_C(f \circ (\text{id}_A \otimes g)) = \text{Tr}_C((\text{id}_B \otimes g) \circ f)$  for  $f : A \otimes D \rightarrow B \otimes C$  and  $g : C \rightarrow D$ .
3.  $\text{Tr}_{\mathbb{1}}(f) = f$  and  $\text{Tr}_{A \otimes B}(f) = \text{Tr}_A(\text{Tr}_B(f))$ .
4.  $\text{Tr}_C(g \otimes f) = g \otimes \text{Tr}_C(f)$ .
5. If  $\mathcal{C}$  is a symmetric monoidal category, then  $\text{Tr}_C(\sigma_{C,C}) = \text{id}_C$  for all  $C \in \text{Ob}(\mathcal{C})$ .

We should mention that if a category is autonomous, it is traced. This will be made clear in the diagrammatic language below.

In a *dagger monoidal category*, we have a contravariant functor  $\dagger : \mathcal{C} \rightarrow \mathcal{C}$  that is the identity on objects and  $f \mapsto f^\dagger$  for all  $f \in \text{Hom}(\mathcal{C})$ . Furthermore,  $f^{\dagger\dagger} = f$ , that is, the functor  $\dagger$  is an involution. We also require  $(f \otimes g)^\dagger = f^\dagger \otimes g^\dagger$  and the isomorphisms  $\alpha, \lambda$ , and  $\rho$  are unitary (that is, their image under the  $\dagger$  functor is their inverse). If  $\mathcal{C}$  is braided, then the map  $\sigma$  must be unitary. Lastly, if  $\mathcal{C}$  is a traced category, then  $\text{Tr}_C(f^\dagger) = (\text{Tr}_C(f))^\dagger$ .

There are many other notions that can be considered when studying monoidal categories, but we shall not need them for this dissertation. For a more complete description of types of monoidal categories, see [96].

### 2.1.2 Functors Between Monoidal Categories

We now need to describe the functors between monoidal categories and how structures on monoidal categories are to be preserved. Let

$$\mathcal{C} = (\text{Ob}(\mathcal{C}), \text{Hom}(\mathcal{C}), \otimes_{\mathcal{C}}, \alpha_{\mathcal{C}}, \lambda_{\mathcal{C}}, \rho_{\mathcal{C}}, \mathbb{1}_{\mathcal{C}})$$

and

$$\mathcal{D} = (\text{Ob}(\mathcal{D}), \text{Hom}(\mathcal{D}), \otimes_{\mathcal{D}}, \alpha_{\mathcal{D}}, \lambda_{\mathcal{D}}, \rho_{\mathcal{D}}, \mathbb{1}_{\mathcal{D}})$$

be two monoidal categories.

**Definition 2.4.** A *(strong) monoidal functor*,  $F : \mathcal{C} \rightarrow \mathcal{D}$  is a functor with natural isomorphisms  $F_0 : F(\mathbb{1}_{\mathcal{C}}) \rightarrow \mathbb{1}_{\mathcal{D}}$ , and  $F_1 : F(A \otimes_{\mathcal{C}} B) \rightarrow F(A) \otimes_{\mathcal{D}} F(B)$  such that the following diagrams commute:

$$\begin{array}{ccc} F(A) \otimes_{\mathcal{D}} (F(B) \otimes_{\mathcal{D}} F(C)) & \xrightarrow{\alpha_{\mathcal{D}}} & (F(A) \otimes_{\mathcal{D}} F(B)) \otimes_{\mathcal{D}} F(C) \\ \text{id}_{F(A)} \otimes F_1 \downarrow & & F_1 \otimes_{\mathcal{D}} \text{id}_{F(C)} \downarrow \\ F(A) \otimes_{\mathcal{D}} (F(B \otimes_{\mathcal{C}} C)) & & (F(A \otimes_{\mathcal{C}} B) \otimes_{\mathcal{D}} F(C)) \\ F_1 \downarrow & & F_1 \downarrow \\ F(A \otimes_{\mathcal{C}} (B \otimes_{\mathcal{C}} C)) & \xrightarrow{F(\alpha_{\mathcal{C}})} & F((A \otimes_{\mathcal{C}} B) \otimes_{\mathcal{C}} C) \end{array}$$
  

$$\begin{array}{ccc} F(B) \otimes_{\mathcal{D}} \mathbb{1}_{\mathcal{D}} & \xrightarrow{\rho_{\mathcal{D}}} & F(B) \\ \text{id}_{F(B)} \otimes F_0 \downarrow & & F(\rho_{\mathcal{C}}) \uparrow \\ F(B) \otimes_{\mathcal{D}} F(\mathbb{1}_{\mathcal{C}}) & \xrightarrow{F_1} & F(B \otimes_{\mathcal{C}} \mathbb{1}_{\mathcal{C}}) \end{array} \quad \begin{array}{ccc} \mathbb{1}_{\mathcal{D}} \otimes_{\mathcal{D}} F(B) & \xrightarrow{\lambda_{\mathcal{D}}} & F(B) \\ F_0 \otimes_{\mathcal{D}} \text{id}_{F(B)} \downarrow & & F(\lambda_{\mathcal{C}}) \uparrow \\ F(\mathbb{1}_{\mathcal{C}}) \otimes_{\mathcal{D}} F(B) & \xrightarrow{F_1} & F(\mathbb{1}_{\mathcal{C}} \otimes_{\mathcal{C}} B) \end{array}$$

The functor  $F$  is called a *strict monoidal functor* if  $F_0$  and  $F_1$  are equalities.

We may also want a monoidal functor to preserve the structures on  $\mathcal{C}$ , such as the braiding, duals, daggers, and traces. If  $\mathcal{C}$  is an autonomous category, then any strong monoidal functor automatically induces a natural isomorphism  $F(A^*) \cong F(A)^*$  [56]. So strong monoidal functors preserve autonomy automatically. If  $F(A^\dagger) = F(A)^\dagger$ , then  $F$  is a monoidal functor between dagger categories. If  $\sigma$  is the braiding for  $\mathcal{C}$ , and  $\tau$  the braiding for  $\mathcal{D}$ , then  $F$  is a functor between braided categories if the following diagram commutes:

$$\begin{array}{ccc} F(A) \otimes_{\mathcal{D}} F(B) & \xrightarrow{F_1} & F(A \otimes_{\mathcal{C}} B) \\ \tau_{F(A)F(B)} \downarrow & & F(\sigma_{AB}) \downarrow \\ F(B) \otimes_{\mathcal{D}} F(A) & \xrightarrow{F_1} & F(B \otimes_{\mathcal{C}} A). \end{array} \quad (2.2)$$

If  $\mathcal{C}$  and  $\mathcal{D}$  are both symmetric, the same condition holds. Lastly, if  $\mathcal{C}$  and  $\mathcal{D}$  are both traced categories, then  $F$  is a functor of traced categories if

$$F(\text{Tr}_{\mathcal{C}}(f)) = \text{Tr}_{F(\mathcal{C})}(F_1^{-1} \circ F(f) \circ F_1).$$

We can now make precise a useful notion for monoidal categories. A monoidal category is called *strict* if the natural isomorphisms  $\alpha$ ,  $\lambda$ , and  $\rho$  are equalities. Recall the following notion of equivalence for categories.

**Definition 2.5.** A functor  $F : \mathcal{C} \rightarrow \mathcal{D}$  is called an *equivalence of categories* if the image of  $F$  contains at least one object from every isomorphism class of objects in  $\mathcal{D}$  (essentially surjective) and for every pair of objects  $A, B \in \text{Ob}(\mathcal{C})$ ,  $F$  induces a bijection  $\text{Hom}(A, B) \rightarrow \text{Hom}(F(A), F(B))$ .

**Theorem 2.6** ([56]). *Every monoidal category is equivalent to a strict monoidal category.*

### 2.1.3 Diagrammatic Language for Monoidal Categories

In a monoidal category, well-formed expressions formed from morphisms, objects, and the symbols  $(, \otimes, *, \dagger, \circ, \text{ and } )$  can all be expressed as diagrams, called *string diagrams*. Furthermore, it has been shown that the value of these diagrams remain unchanged under the allowed homotopies [55, 54]. This makes expressing formulae in monoidal categories more intuitive and makes many of the axioms above very natural. We describe the appropriate graphical notions for the types of monoidal categories listed above.

For each of the string diagrams now described, see Figure 2.1. Objects in a monoidal category  $\mathcal{C}$  are denoted by labeled arrows (which we also sometimes call wires or edges) which, by our convention, we shall have pointing left. The exception is the distinguished object  $\mathbf{1}$ , which is denoted simply by empty space. The tensor product of two objects,  $A \otimes B$  is drawn as two arrows, one above the other. Our convention shall be that the tensor product is taken from top to bottom.

A morphism  $f : (\otimes_{i=1}^n A_i) \rightarrow (\otimes_{i=1}^m B_i)$  shall be depicted as a box with the arrows labeled  $B_i$  on its left and the arrows labeled  $A_i$  on its right. Composition is achieved by horizontally stringing morphisms together and tensor product is once again vertical juxtaposition. The morphism  $\text{id}_A$  is the same way as the object  $A$ . It may be viewed as stretching the arrow.

The string diagrams we have described thus far are the basic building blocks for all string diagrams. However, as we add structures to our monoidal categories, we must add other pictorial representations. The first is for braided categories. The morphism  $\sigma_{AB}$  is denoted by crossing the arrows for the objects  $A$  and  $B$ , see Figure 2.2 (a). If the category is symmetric, there is no notion of which wire is under the other, so we use the picture in Figure 2.2 (b).



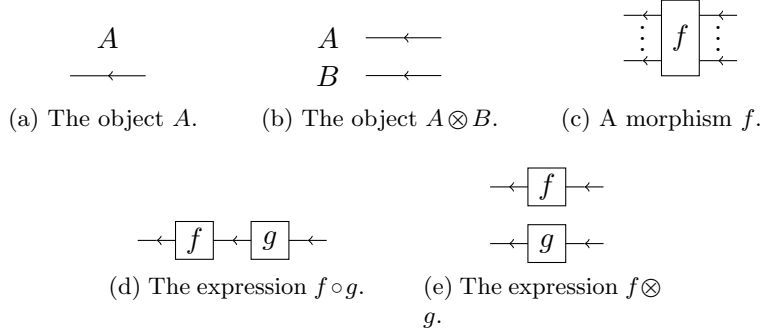


Figure 2.1: Basic String Diagrams

The arrows on objects have so far had no importance. However, in an autonomous category, where a notion of duals for objects exists, the arrows are used to keep track of primal and dual objects. The morphisms  $\eta_A$  and  $\epsilon_A$ , sometimes called the cup and cap morphisms, are the maps that bend wires, reversing the arrows. The map  $\eta_A$  can be thought of as the bilinear pairing of  $A$  and  $A^*$ . Indeed, this will be the case when working with a subcategory of  $\text{Vect}_k$ . The map  $\epsilon_A$  is the adjoint of this map. The diagrams for these maps are given in Figure 2.2 (c). The axioms given in Equations 2.1 state that if the cup and cap morphisms are paired in such a way to make an “s” shape, they can be pulled straight into the identity morphism. These are sometimes called the “yank axioms”.

Note that composing the maps  $\eta_A$  and  $\epsilon_A$  gives a cycle. In the category  $\text{Vect}_k$ , this morphism can be easily checked to be the trace of identity map on the vector space  $A$ . Cycles correspond to traces more generally. For a traced monoidal category, the operation  $\text{Tr}_C$  should be thought of as a partial trace over the object  $C$  and is depicted as in Figure 2.2 (d). Note that since cycles can be formed using the cup and cap morphisms, the previous assertion that autonomous categories are traced is justified.

The first three axioms for a traced monoidal category then say that one can stretch cycles without affecting the expression it represents, move morphisms around a cycle, and the order of forming cycles does not matter. For a symmetric monoidal category,  $\text{Tr}_A(\sigma_{A,A})$  is the identity arrow on  $A$  with a loop in it. By straightening out the loop, we see that  $\text{Tr}_A(\sigma_{A,A}) = \text{id}_A$ .

For dagger monoidal categories, we do not need to add any more diagrammatic rules. Thus we have all of the necessary ingredients for building the string diagrams that will be relevant for this dissertation. At various

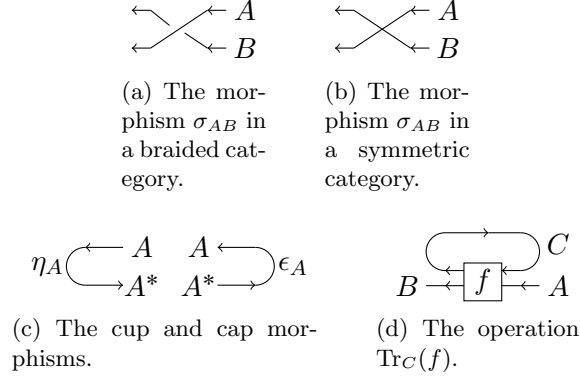


Figure 2.2: Further String Diagrams

points we may introduce further diagrammatic conventions e.g., for specific tensors, but these shall be defined as they are needed.

## 2.2 Tensor Networks

Given a monoidal category with a specified set of structures (e.g. braided, closed compact), the set of allowed diagrams forms a strict monoidal category with the same set of structures. The above process of turning formulae in a structured monoidal category  $\mathcal{C}$  actually describes a functor  $\phi$  from  $\mathcal{C}$  to a category of diagrams  $\mathcal{D}$ . This category is a free category in the following sense. For every functor from  $F : \mathcal{C} \rightarrow \mathcal{F}$  that preserves a set of structures, there exists a functor  $\Psi$  such that the following diagram commutes

$$\begin{array}{ccc} \mathcal{C} & \xrightarrow{\phi} & \mathcal{D} \\ & \searrow F & \downarrow \exists \Psi \\ & & \mathcal{F} \end{array}$$

Therefore, every monoidal category can be viewed as a quotient category of a category of diagrams. In particular, the category  $\text{Vect}_k$  is a dagger closed compact monoidal category. We are interested in functors from the category of diagrams into  $\text{Vect}_k$ .

**Definition 2.7.** A *tensor network* is the image of a diagram in the category  $\text{Vect}_k$ .

**Proposition 2.8.** The category  $\text{Vect}_k$  is a dagger closed compact category.

*Proof.* This category is clearly symmetric as  $V \otimes W \cong W \otimes V$  and the isomorphism is an involution. As previously mentioned, the map  $\eta_A$  is the map from  $A \otimes A^* \rightarrow k$  by  $a \otimes \alpha \mapsto \alpha(a)$ ;  $\epsilon_A$  is the adjoint map. The function  $\text{Tr}_C$  is the partial trace over the vector space  $C$ . The  $\dagger$  functor is the usual adjoint of a linear transformation. We do not check the commutativity of the appropriate diagrams here.  $\square$

When working with general tensor networks, we often specify a basis for each vector space for convenience. By choosing a basis, a vector space and its dual can be identified. Then we drop the arrows from the tensor network, instead working with undirected wires. There are some categories that we shall work with, however, where the arrows matter.

When two tensors are connected by a wire in a tensor network, this can be turned into another tensor via an operation called *tensor contraction*. Given a tensor  $T \in V_1 \otimes \cdots \otimes V_s \otimes (V_{s+1})^* \otimes \cdots \otimes (V_r)^*$  written in a chosen basis  $\{v_j^i\}$  for each  $V_i$  and  $\{\eta_j^i\}$  the associated dual basis, denote the coefficient of basis vector  $v_{j_1}^1 \otimes \cdots \otimes v_{j_s}^s \otimes \eta_{j_{s+1}}^{s+1} \otimes \cdots \otimes \eta_{j_r}^r$  in  $T$  by  $T_{j_{s+1}, \dots, j_r}^{j_1, \dots, j_s}$ .

Making use of the fact that indices can be raised or lowered as we wish as we have identified vector spaces with their duals, let our two tensors, expressed in a basis, be

$$T = \sum_j T_j^{i_1, \dots, i_t} (v_{i_1}^1 \otimes \cdots \otimes v_{i_t}^t \otimes \eta_j^{t+1}) \quad \text{and} \\ S = \sum_{k_1, \dots, k_s} S_{k_1, \dots, k_s}^j (v_j^{t+1} \otimes v_{k_1}^{t+2} \otimes \cdots \otimes v_{k_s}^{t+s+1}),$$

where  $j$  is the index they share by being connected by a wire. Then we contract along this shared index to form the new tensor with coefficients

$$U_{k_1, \dots, k_s}^{i_1, \dots, i_t} = \sum_j T_j^{i_1, \dots, i_t} S_{k_1, \dots, k_s}^j.$$

Tensor contraction can be similarly expressed in bra-ket notation, as detailed in Section 1.1, in the following way

$$U = \sum_{\substack{i_1, \dots, i_t \\ k_1, \dots, k_s}} U_{k_1, \dots, k_s}^{i_1, \dots, i_t} |i_1 \cdots i_t\rangle \langle k_1 \cdots k_s| = \\ \sum_j T_j^{i_1, \dots, i_t} S_{k_1, \dots, k_s}^j |i_1 \cdots i_t\rangle \langle j| \langle j| \langle k_1 \cdots k_s| = \\ \sum_j T_j^{i_1, \dots, i_t} S_{k_1, \dots, k_s}^j |i_1 \cdots i_t\rangle \langle k_1 \cdots k_s|$$

noting that for an orthonormal basis,  $\langle i|j\rangle = \delta_{ij}$ .

In this way, every tensor network can be viewed as an element of  $k^{\otimes n}$ , where  $n$  is the number of dangling (or open) wires. We can also view a tensor network as a multilinear polynomial. Let  $T$  be a tensor network and denote the boxes in the diagram  $M_1, \dots, M_\ell$ . Then by varying the tensors placed in the boxes  $M_1, \dots, M_\ell$ , with each tensor  $M_i \in W_i$ , for the appropriate vector space  $W_i$ , we get a multilinear polynomial function  $F_T(M_1, \dots, M_\ell) : \bigotimes_{i=1}^\ell W_i \rightarrow k^{\otimes n}$ .

In fact, the polynomial  $F_T$  has even more structure. Let  $E$  be the set of edges in the tensor network  $T$  and  $V_e$  the associated vector space. Then consider the group  $\mathrm{GL}_T := \bigoplus_{e \in E} \mathrm{GL}(V_e)$ . It acts on the polynomial  $F_T$  by applying to every  $e \in E$ ,  $g_e g_e^{-1}$  where  $g_e \in \mathrm{GL}(V_e)$ . The clearly leaves  $F_T$  invariant. However, it induces an action on  $\bigotimes_{i=1}^\ell W_i$  and we see that  $F_T$  is a covariant of this group action.

**Observation 2.9.** Thus each coordinate of the map  $F_T$  is a polynomial invariant of the group action  $\mathrm{GL}_T$ .

A tensor network that is an element of  $\mathrm{Hom}(\mathbb{1}, \mathbb{1}) \cong k$  is called *closed* tensor network. This observation will be studied further in Chapters 7 and 8.

## Chapter 3

# The Complexity of Contracting Tensor Networks

In this chapter, as well as Chapters 4 and 5, we are concerned with the following question: working in the category  $\text{Vect}_{\mathbb{C}}$ , if we are given a tensor network  $T \in \text{Hom}(\mathbb{1}, \mathbb{1}) \cong \mathbb{C}$ , which complex number does it represent? This is called the *tensor contraction problem*. It turns out that for general tensor networks, answering this problem is hard.

**Definition 3.1.** If  $\mathcal{L} \in \text{NP}$ , let  $F_{\mathcal{L}} : \mathcal{L} \rightarrow \mathbb{N}$  be the function that takes a problem  $P \in \mathcal{L}$  to the number of accepting solutions. A problem is in the complexity class  $\#P$  if it is of the form “compute  $F_{\mathcal{L}}(P)$  for some  $\mathcal{L} \in \text{NP}$  and any  $P \in \mathcal{L}$ .”

The complexity class  $\text{FP}$  is the subset of problems in  $\#P$  that can be solved in polynomial time.

The class  $\#P$  was introduced by Leslie Valiant to classify the difficulty of computing the permanent of a matrix [100]. The counting version of any  $\text{NP}$ -hard problem is immediately  $\#P$ -hard as well. A bit surprisingly, while 2-SAT can be decided to have a satisfying assignment in polynomial time, counting the number of such satisfying assignments is also  $\#P$ -complete. However, as with the class  $\text{NP}$ , the problem that is arguably most studied is  $\#SAT$ , the problem of counting the number of satisfiable solutions to a SAT problem. Much work focuses particularly on  $\#3\text{-SAT}$  or  $\#2\text{-SAT}$ .

It turns out that the tensor contraction problem is  $\#P$ -hard. In the following section we discuss the reduction of  $\#SAT$  problems to tensor contraction problems. Later in the chapter, we exploit graphical properties of a tensor network modeling a  $\#SAT$  problem to find a class of efficiently tractable Boolean formulas.

### 3.1 Modeling #SAT Problems as Tensor Networks

We describe a way to write a Boolean satisfiability problem as a tensor network, as in [52, 62]. Suppose we are given a SAT formula  $f$  and we wish to express it as a tensor  $\psi_f$ . Let  $x_1, \dots, x_n$  be the variables appearing in this formula. We further suppose that  $f$  is given to us in the following form.

**Definition 3.2.** A Boolean formula is in *Conjunctive Normal Form* (or CNF form for short) if it is of the form

$$\bigwedge_{i=1}^{\alpha} \left( l_{i,1} \vee \dots \vee l_{i,k_i} \right)$$

where each literal  $l_{i,j}$  is of the form  $x_k$  or  $\neg x_k$ . A problem is in  $r$ -CNF form if each clause has  $r$  literals.

We assign an open wire to each variable  $x_i$ . If a variable  $x_i$  appears in  $k$  clauses, we make  $k$  copies of  $x_i$  via the COPY-tensor  $|0\rangle\langle 0|^{\otimes k} + |1\rangle\langle 1|^{\otimes k}$ . We call  $k$  the *degree* of the COPY-tensor. It is depicted as a solid black dot in a tensor network.

A Boolean gate (or clause)  $\varphi : \{0, 1\}^m \rightarrow \{0, 1\}$  is expressed as the tensor

$$\sum_{\mathbf{x} \in \{0,1\}^m} |\mathbf{x}\rangle\langle \varphi(\mathbf{x})|.$$

Then the COPY-tensor associated to each variable is connected to each tensor representing a clause which that variable appears in. This gives a tensor network with  $n$  dangling wires corresponding to the variables and  $\alpha$  dangling wires corresponding to the output of each clause. Since we want each clause to have the value of 1, we attach to each of these wires the tensor  $|1\rangle$ . This is equivalent to composing with the tensor  $|1_{\alpha}\rangle = |1\rangle^{\otimes \alpha}$ . It is not hard to check that the entire network can be contracted to the tensor

$$\psi_f = \sum_{\mathbf{x}} |\mathbf{x}\rangle\langle f(\mathbf{x})|1_{\alpha}\rangle = \sum_{\mathbf{x}} f(\mathbf{x})|\mathbf{x}\rangle.$$

We call a state of the form  $\psi_f$  a *Boolean state*.

**Remark 3.3** (Counting SAT solutions). *Let  $f$  be a SAT instance. Then  $\|\psi_f\|^2$ , using the standard norm, counts the number of satisfying solutions of  $f$ .*

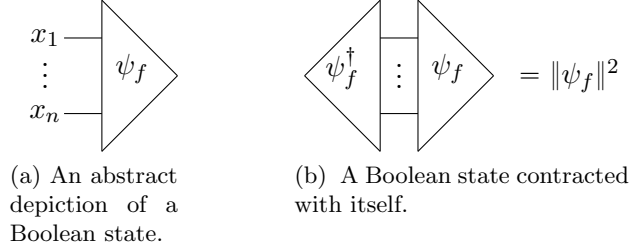


Figure 3.1: Tensor Networks the Count the Number of Satisfying Solutions for a Boolean Formula

We calculate the inner product of this state with itself viz

$$\|\psi_f\|^2 = \sum_{\mathbf{x}, \mathbf{y}} f(\mathbf{x})f(\mathbf{y})\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{\mathbf{x}} f(\mathbf{x}), \quad (3.1)$$

which gives the number of satisfying inputs. This follows since  $\langle \mathbf{x}, \mathbf{y} \rangle = \delta_{\mathbf{x}\mathbf{y}}$ .

We note that solving the counting problem for general formula is known to be #P-complete [101]. The condition

$$\|\psi_f\| > 0 \quad (3.2)$$

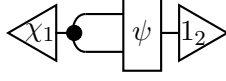
implies that the SAT instance  $f$  has a satisfying assignment. Determining if this condition holds for general Boolean states is an NP-complete decision problem.

The tensor network contraction for the counting problem is depicted in Figure 3.1: (a) gives a network realization of the function and (b) is the contraction that represents the norm of  $\psi_f$ . Note that we represent vectors and covectors as triangles pointing in opposite directions. This is meant to resemble the shape of a “bra” and “ket”.

Alternatively, we can count the number of solutions to a Boolean formula  $f$  in the following way. For an  $n$ -variable Boolean formula  $f$ , let  $\chi_n = (\langle 0| + \langle 1|)^{\otimes n}$ . Then

$$\langle \chi_n, \psi_f \rangle = \sum_{\mathbf{x} \in \{0,1\}^n} f(\mathbf{x})\langle \mathbf{x}, \mathbf{x} \rangle = \sum_{\mathbf{x} \in \{0,1\}^n} f(\mathbf{x}).$$

**Example 3.4.** As a first example, we consider the simple non-satisfiable Boolean formula  $x \wedge (\neg x)$ , which is in Conjunctive Normal Form. The tensor representing the literal  $x$  is  $|0\rangle\langle 0| + |1\rangle\langle 1|$  and the tensor representing the literal  $\neg x$  is  $|0\rangle\langle 1| + |1\rangle\langle 0|$ . The tensor  $\chi_1 = \langle 0| + \langle 1|$ . The picture is



The dot is the aforementioned COPY-tensor. The tensor  $\psi = (|0\rangle\langle 0| + |1\rangle\langle 1|) \otimes (|0\rangle\langle 1| + |1\rangle\langle 0|)$ . Then contracting, we get

$$(\langle 0| + \langle 1|) \circ (|0\rangle\langle 00| + |1\rangle\langle 11|) \circ (|01\rangle\langle 01| + |10\rangle\langle 10| + |00\rangle\langle 01| + |11\rangle\langle 10|) \circ (|11\rangle) = 0.$$

Seeing that any #SAT problem can be easily rephrased as a tensor contraction problem, we can conclude the following well-known fact: **the tensor contraction problem is #P-hard.**

### 3.1.1 Complexity Theory Via Tensor Networks

Relationships between multilinear maps and counting problems have long been known in computer science and physics [100, 85]. This observation makes natural the generalization from classical to quantum computing. Quantum circuits are in fact tensor networks with a measurement, where the only morphisms allowed are unitary matrices [8].

It is not hard to see that tensor networks actually provide an alternative model of computation using the ideas outlined above. Consider a function  $f : \{0, 1\}^n \rightarrow \mathbb{C}$ , then consider the tensor  $\psi_f = \sum_{\mathbf{x} \in \{0, 1\}^n} f(\mathbf{x}) |\mathbf{x}\rangle$ . Then  $f(\mathbf{x}) = \langle \mathbf{x} | \psi_f$  for any  $\mathbf{x} \in \{0, 1\}^n$ . So computing any binary function can be expressed as a tensor contraction problem.

Because of this, much work has been done to understand which tensor networks can be contracted in polynomial time. Of particular interest are the following generalization of Boolean formulas.

**Definition 3.5.** A *constraint satisfaction problem* (or CSP for short) is a triple  $(X, D, C)$  where  $X = \{x_1, \dots, x_n\}$  denotes a set of variables,  $D = \{D_1, \dots, D_n\}$  are the domains for the respective  $x_i$ , and  $C = \{C_1, \dots, C_k\}$  are a set of constraints  $C_j = (X_j, R_j)$ , where  $X_j \subseteq X$  and  $R_j \subseteq \times_{x_i \in X_j} D_i$  is a  $|X_j|$ -ary relation.

A solution to a CSP is an assignment to each variable  $x_j := t_j$  such that for every  $C_j$ ,  $X_j$  satisfies the relation  $R_j$ . A complex valued CSP is one where the assignment of the variables are drawn from  $\mathbb{C}$ . A complex valued #CSP problem is the problem of counting the number of satisfying solutions to a CSP.



A complex-valued  $\#CSP$  problem can be modeled as a tensor contraction problem similarly to our construction of Boolean states. Each  $D_i$  can be associated with a the set  $[|D_i|]$ . Then the variable tensor  $x_i$  is  $\langle 0|^{\otimes |D_i|} + \langle 1|^{\otimes |D_i|}$ . The clauses are now removed of any constraint. The vector space associated to a wire connecting a variable  $x_i$  to a particular clause has dimension  $|D_i|$ . In this way, every  $\#CSP$  problem can be expressed in terms of tensor networks.

In general, solving  $\#CSP$  problems is  $\#P$ -hard [19, 25]. However, several dichotomy theorems exist for when a tensor network modeling a  $\#CSP$  problem is either in  $FP$  or  $\#P$ -hard [16, 32, 17]. Furthermore, great emphasis has been placed on how easily a given  $\#CSP$  problem can be determined to be tractable or not.

One can also restrict to  $\#CSP$  problems with extra constraints, such as the restriction that its tensor network representation be planar. An important class of efficiently tractable tensor networks are Valiant's matchcircuits which have a planarity restriction [103, 98, 99]. Work has been done to show that this tensor networks are precisely the planar tractable tensor networks [18]. These particular tensor networks will be discussed further in Chapter 4.

Throughout the rest of this chapter, we focus on  $\#SAT$  problems, which every  $\#P$  problem can easily be expressed as, and discuss different criteria for tractability of such problems. We also discuss how effective each of these criterion are, in the sense that testing whether or not a given  $\#SAT$  problem satisfies a given criterion can be done in polynomial time.

## 3.2 Criteria for Tractability of $\#SAT$ Problems

When considering a criterion for tractability of a  $\#SAT$  problem, there are two natural questions: How does this property affect the computation of the problem and how easy is it to determine the value of said property. In this chapter we consider the following criteria:

1. Tensor networks with the structure of a tree,
2. The fan-out of the variables,
3. Treewidth of the tensor network,
4. Rényi entropy of tensor networks.

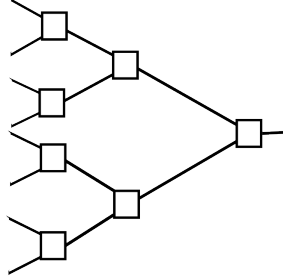


Figure 3.2: A Boolean Function Expressible as a Tree.

Tree tensor networks can always be efficiently contracted. The algorithm is the sum product algorithm and has been rediscovered many times. Within the tensor network formalism, the algorithm was provided in [75].

We measure the fan-out of the variables via the number of COPY-tensors in the Boolean state as well the degree of the COPY-tensors. The number of COPY-tensor networks restricts the number of variables that can appear in more than one clause. The degrees of the COPY-tensors measures how many clauses a variable may appear in. By combining these two conditions, we can find a new class of tractable #SAT problems. Theorem 3.15 gives an explicit polynomial time algorithm for such problems.

The related study of Boolean formulas in  $r$ -CNF form where the number clauses a variable can appear in is restricted have been studied previously [97, 30]. We will explain how this criteria relates to bounded treewidth, specifically which #SAT problems are tractable via the algorithm in Theorem 3.15 but not by the tree-width based algorithms. Lastly, we discuss the Rényi entropy as a measure of complexity, as suggested in [21]. We address the fact that it may not be defined or easily computed.

### 3.2.1 Some SAT Instances That Are Always Satisfiable

We focus on a class of SAT instances that are always satisfiable simply by the structure of their tensor networks, namely trees where every variable is a leaf. These are more familiarly known as read-once functions. Here, we do not demand that the tensor network representation of the be in the form described in Section 3.1. Instead, we look at tensor networks as depicted in Figure 3.2.1, where the variables are the left-side leaves, the right-side leaf is the output of the function, and each box is any Boolean function except the two constant functions. Such a network can always be rewritten into CNF form.

**Definition 3.6** (read-once). A function  $f$  is called *read-once* if it can be represented as a Boolean expression using the operations conjunction, disjunction and negation in which every variable appears exactly once. We call such a factored expression a read-once expression for  $f$ .

We call such a function ROF for short. The definition implies that there are no COPY-tensors since every variable can only appear in one of the Boolean tensors in the network. As such, it does indeed have a tree structure when expressed as a tensor network.

These formulas represent a special subclass of  $r, s$ -SAT, which is defined to be the decision problem for SAT formulas written in  $r$ -CNF form where each variable appears in at most  $s$  clauses. Read once formulas, given in CNF form, represent the cases where  $r$  is general and  $s = 1$ . These problems have been studied classically and we mention two seminal results:

**Theorem 3.7** (Tovey, [97]). *Every instance of  $r, r$ -SAT is satisfiable.*

**Theorem 3.8** (DuBois, [30]). *If every instance of  $r_0, s_0$ -SAT is satisfiable, then  $r, s$ -SAT is satisfiable for  $r = r_0 + \lambda$  and  $s \leq s_0 + \lambda[s_0/r_0]$ ,  $\lambda \in \mathbb{N}$ .*

This implies that  $r, s$ -SAT is satisfiable for  $s \leq r$  by Theorem 3.7 and letting  $\lambda = 0$  in Theorem 3.8. In particular, all ROF are satisfiable. Using tensor networks, we can give a very short proof of this.

**Definition 3.9.** Let  $f : \{0, 1\}^n \rightarrow \{0, 1\}$  be a Boolean function. Considering the tensor  $\tilde{\psi}_f = \sum_{\mathbf{x} \in \{0, 1\}^n} |\mathbf{x}\rangle \langle f(\mathbf{x})|$ , we call a tensor of the form  $\tilde{\psi}_f^\dagger \tilde{\psi}_f$  a *diagonal map*.

**Lemma 3.10.** *For a Boolean function  $f$ ,*

$$\tilde{\psi}_f^\dagger \tilde{\psi}_f = \#f^{-1}(0)|0\rangle\langle 0| + \#f^{-1}(1)|1\rangle\langle 1|$$

where  $\#f^{-1}(b)$  denotes the size of the pre-image of  $b$ . Furthermore, if  $f$  is not a constant function, we can normalize it to get a tensor  $\zeta_f$  such that  $\zeta_f^\dagger \zeta_f = \text{id}$ .

*Proof.*

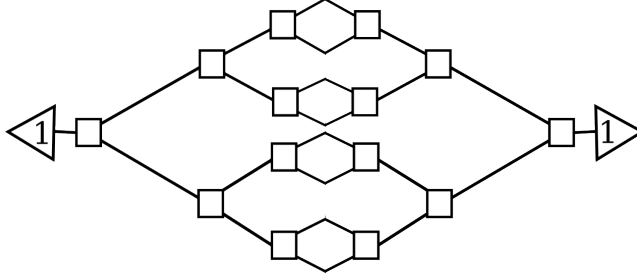
$$\begin{aligned} \tilde{\psi}_f^\dagger \tilde{\psi}_f &= \sum_{\mathbf{x}, \mathbf{y}} |f(\mathbf{x})\rangle \langle \mathbf{x}| \mathbf{y}\rangle \langle f(\mathbf{y})| \\ &= \sum_{\mathbf{x}} |f(\mathbf{x})\rangle \langle f(\mathbf{x})| \\ &= \#f^{-1}(0)|0\rangle\langle 0| + \#f^{-1}(1)|1\rangle\langle 1|. \end{aligned}$$

Now let  $\zeta_f = \sum_{\mathbf{x}} \sqrt{\#f^{-1}(f(\mathbf{x}))}^{-1} |\mathbf{x}\rangle\langle f(\mathbf{x})|$ . It is clear that  $\zeta_f$  has the desired property and is well defined since we assumed that  $f$  was non-constant.  $\square$

For a Boolean state representing an ROF, we replace every gate with its normalization. This gives positive weights to the different assignments of variables. The contraction then sums up the weights of the satisfying assignments. It is clear that normalizing a ROF does not change the fact that the norm will be zero if and only if it has no satisfying solution. The resulting scalar, however, will no longer reflect the number of satisfying solutions.

**Theorem 3.11.** *Every ROF built up from non-constant gates is satisfiable.*

*Proof.* To show that  $f$  is satisfiable, we must show that  $\|\tilde{\psi}_f|1\rangle\|^2 \geq 1$ . This tensor network is depicted in the picture below.



We normalize every gate  $\tilde{\psi}_g$  in the tree using Lemma 3.10. As discussed above, this new tensor network will have norm zero if and only if it is unsatisfiable. We observe a series of nested diagonal maps. As any ROF does not allow a constant Boolean gate, by Lemma 3.10 these maps are equivalent to the identity. So we can successively collapse them until we get the contraction  $\langle 1 | (\text{id}) | 1 \rangle = 1$ . So it is indeed satisfiable.  $\square$

### 3.2.2 Efficient Tensor Contraction Algorithm for Boolean States With Restricted Fan-out

Consider a tensor network with a known efficient, i.e. polynomial, contraction method. By inserting new wires and tensors into the network, one can construct a tensor network which is not necessarily easy to contract. Of interest are ways to reduce tensor networks to those that are computationally tractable in an efficient way.

In particular, we are interested in transforming Boolean states into other Boolean states with polynomial time evaluations. One approach is to restrict the number of tensors added to an efficiently contractible class of tensor networks to be at most  $O(\log n)$ , where  $n$  is the number of variables.

**Definition 3.12** (Tensor network correspondence). Let  $v$  be the minimum number of vertices needed to be removed from the network  $T$ , with their incident edges becoming dangling wires, to transform it into a network  $U$  known to be efficiently contractible. The *correspondence* of  $T$  with respect to  $U$  is defined to be  $1/v$ .

For our purposes, we will be interested in restricting our definition of correspondence to the number of COPY-tensors needed to be removed to recover a tensor network that can be quickly evaluated. In this way, we can construct a wide class of counting problems that can be solved efficiently using tensor contraction algorithms. Here we focus on the Boolean states that are close to tree tensor networks.

It is well known that tree tensor networks are polynomially contractible in the number of gates [43, 65, 75]. For Boolean states, the gate count includes the tensors  $\langle 0| + \langle 1|$  placed on the variable wires.

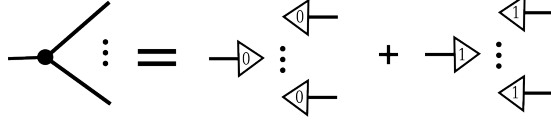
**Definition 3.13.** Given a rooted tree, a *limb* is a sequence of vertices  $v_1, \dots, v_k$  such that  $v_2, \dots, v_{k-1}$  each only have one child and  $v_{i+1}$  is a child of  $v_i$ .

**Proposition 3.14.** *An  $n$ -variable Boolean state that is a tree has  $O(n)$  gates.*

*Proof.* A Boolean state has a natural root: the output bit. We argue that the limbs of a Boolean state are of length at most three. Take a limb  $v_1, \dots, v_k$ . Then consider  $v_i$ ,  $i \neq 1, k$ . It is a unary operation on clauses, implying that it must either be the identity or negation operator. We can therefore assume that there is at most a single negation operator between  $v_1$  and  $v_k$  and nothing else. So at worst the tree is a perfect binary tree which has  $O(n)$  gates.  $\square$

So we can conclude that the norm of a Boolean state that is a tree can be computed in polynomial time in the number of variables.

Let  $X$  be a Boolean state. The multilinearity of tensor networks allows us to remove a COPY-tensor, resulting in a sum of tensor networks:



We note that there are two summands, where either 0 or 1 is attached to the dangling wires resulting from the removal of the COPY-tensor. Given a tensor network  $X$ , if we remove a COPY-tensor, we denote the two summands  $X_0$  and  $X_1$ . Furthermore, we have that

$$\mathcal{C}\{X, X\} = \sum_i \mathcal{C}\{X_i, X_i\}.$$

We can now give an algorithm for contracting a tensor network, assuming we have an algorithm for contracting a tree tensor network in polynomial time. Let  $C_1, \dots, C_m$  be the COPY-tensors appearing in  $X$ . Then

$$\mathcal{C}\{X, X\} = \sum_{i_1, \dots, i_m} \mathcal{C}\{X_{i_1, \dots, i_m}, X_{i_1, \dots, i_m}\}$$

where  $X_{i_1, \dots, i_m}$  is the tree formed by removing all the COPY-tensors and assigning the value  $i_k$  to the wires that were incident to  $C_k$ . So the algorithm for computing  $\mathcal{C}\{X, X\}$  computes the expression as a sum of contractions of trees.

**Theorem 3.15** (Upper bounding tensor contraction in terms of COPY-tensors). *Given a tensor network as described in Section 3.1, the complexity of evaluating this network is  $O((g + cd)^{O(1)} 2^c)$  where  $c$  is the number of COPY-tensors,  $g$  is the number of gates, and  $d$  is the maximal degree of any COPY-tensor.*

*Proof.* Contracting a tensor network is upper bounded by evaluating the expression

$$\mathcal{C}\{X, X\} = \sum_{i_1, \dots, i_m} \mathcal{C}\{X_{i_1, \dots, i_m}, X_{i_1, \dots, i_m}\}.$$

For each COPY-tensor removed, we double the number of summands. We also add a gate to each dangling wire created by removing a COPY-tensor. So the computation is bounded by summing over the contraction of  $2^c$  trees, each of which can be contracted in time  $O((g + cd)^{O(1)})$  [75].  $\square$

**Remark 3.16.** *We must mention that Theorem 3.15 only applies when the Boolean state is given by a tensor network as described in Section 3.1. Otherwise, there are many tensor networks to describe a given Boolean state, many of which invalidate the statement of Theorem 3.15.*

**Corollary 3.17** (A class of efficiently solvable #SAT instances). *Suppose that a Boolean state with  $n$  variables has inverse logarithmic correspondence with a tree tensor network and the maximal degree of the COPY-tensors is  $O(n^\alpha)$ . Then this Boolean state can be contracted in polynomial time.*

*Proof.* Inverse logarithmic correspondence implies that if the SAT instance has  $n$  variables, then it has  $O(\log(n))$  COPY-tensors. By Theorem 3.15, the complexity of contraction is  $O((g + n^\alpha \log(n))^{O(1)} 2^{\log(n)})$ . By Proposition 3.14,  $g$  is  $O(n)$ , so the algorithm presented will contract the tensor network in polynomial time.  $\square$

Theorems 3.7 and 3.8 only pertain to instances of  $r, s$ -SAT that are always satisfiable; they do not address the complexity of the corresponding counting problem. For instance, read-twice monotone 3-CNF formulas are always satisfiable by Theorems 3.7 and 3.8, but counting the number of solutions is #P-hard [106].

In contrast, Theorem 3.15 gives conditions for the tractability of the counting problem associated to  $r, s$ -SAT. One of these conditions is not implicit in the definition of  $r, s$ -SAT, namely that we bound the number of variables that can appear in more than one clause. To restate Corollary 3.17, we have shown that  $\#r, \text{poly}(n)$ -SAT is polynomial time countable if the number of COPY-tensors is bounded by  $\log(n)$ .

### 3.2.3 Relation to Other Algorithms

There have been several results relating the complexity of contracting tree tensor networks to various measures such as treewidth, clique-width, and branchwidth. An overview can be found in [83]. The general type of result is that the time to solve a #SAT instance is polynomial in the number of variables and exponential in the corresponding notion of width [4, 36].

We discuss how our algorithm compares with algorithms based on the treewidth of Boolean states and the relationship between treewidth and the number of COPY-tensors. Contracting a tree tensor network  $T$  takes time  $g^{O(1)} \exp(\text{tw}(T))$ , where  $\text{tw}(T)$  is the treewidth of  $T$  [75]. More explicitly, there exists the following results:

**Theorem 3.18** ([36]). *Given an  $n$ -variable SAT formula with treewidth  $k$ , there is an algorithm counting the number of solutions in time  $4^k(n + n^2 \log_2(n))$ .*

**Theorem 3.19** ([94]). *Given a SAT formula with treewidth  $k$ , largest clause of size  $l$ , and  $N$  the number of nodes of the tree-decomposition, the #SAT problem can be solved in time  $O(4^k l N)$ .*

**Proposition 3.20.** *Given a Boolean state, let  $k$  be the treewidth and  $c$  the number of COPY-tensors. Then  $k \leq c$ .*

*Proof.* It is well known that if  $H$  is a subgraph of  $G$ , then  $\text{tw}(H) \leq \text{tw}(G)$ . If we place the tensor  $\langle 0| + \langle 1|$  on each variable wire and then compose with the COPY-tensors to get a tensor network  $T$ , a variable will correspond to a leaf if and only if it appears in exactly one clause. The treewidth of  $T$  with  $n$  variables,  $n'$  of which are not leaves, and  $m$  clauses, is at most the treewidth of  $K_{n',m}$ , which is known to be  $\min(m, n')$ . This is because adding leaves to a graph does not increase the treewidth. On the other hand,  $c = n'$ .  $\square$

Note that if the number of variables is large with respect to the number of clauses, then  $c$  may be much larger than the treewidth,  $k$ . However, the algorithms are comparable as long as  $c$  is bounded by  $2^k$  by Theorems 3.15 and 3.18. We note that if the clause to variable ratio is extremely small or large, the tree width is small and the tensor network can be contracted efficiently. In the critical case when  $c \cong k$ , our algorithm runs *exponentially faster* in  $k$ . There is the added advantage that, unlike treewidth, calculating  $c$  is not NP-complete. As such,  $c$  is an attractive estimate for the complexity of the counting problem. However, the trade off is that it is a cruder measurement.

### 3.2.4 Rényi Entropy and Complexity

Rényi entropies [92] have also been proposed as an indicator of the complexity of counting problems [21]. The Rényi entropies are attractive as they are well understood and motivated from areas such as physics and statistics. We look at the question pertaining to the use of Rényi entropies as a measurement of counting complexity. A natural question is the efficiency of determining the Rényi entropy of a given state. Many proposed measures of complexity are difficult to determine. As we noted in Section 3.2.2, the number of COPY-tensors can be easily computed, whereas treewidth cannot.

Given a Boolean state  $\psi_f$ , we can instead look at the operator  $|\psi_f\rangle\langle\psi_f|$ . We choose a partition for the rows and columns into two disjoint subsystems,  $A$  and  $B$ . This is called a *bipartition* of the operator, denoted  $A : B$ . The



Rényi entropy of order  $q$  with respect to a bipartition  $A : B$  is defined to be

$$H_q^{AB}(\rho) = \frac{1}{1-q} \ln \text{Tr}\{\rho^q\} = \frac{1}{1-q} \ln \sum_i \lambda_i^q, \quad (3.3)$$

$$\rho = \frac{1}{Z_0} \text{Tr}_A(|\psi_f\rangle\langle\psi_f|).$$

Here,  $Z_0 = \text{Tr}(|\psi_f\rangle\langle\psi_f|)$ . The case  $q \rightarrow 0$  gives the rank of the bipartition and the positive sided limit  $q \rightarrow 1$  recovers the familiar von Neumann Entropy  $H_{q \rightarrow 1}^{AB}(\rho) = -\text{Tr}(\rho \ln \rho)$ .

In terms of the counting problem, note that the state  $|\psi_f\rangle$  could be given by the zero vector, in which case it is not possible to define the state  $\rho$ : so more generally we can consider the unnormalized variant of  $\rho$ , which is defined as  $\rho = \text{Tr}_A(|\psi_f\rangle\langle\psi_f|)$ . This has the following computational significance.

**Theorem 3.21** ( Rényi entropy implies satisfiability). *For any bipartition, deciding if the Rényi entropy is defined is NP-hard.*

*Proof.* Given an unnormalized Boolean state  $\rho$ ,

$$\rho = \text{Tr}_A(|\psi\rangle\langle\psi|)$$

for some bipartition  $A : B$ ,  $H_q^{AB}(\rho)$  takes a finite value if and only if the Boolean state is satisfiable. Note that this is independent of choice of bipartition as  $\ln \text{Tr}(\rho^q)$  is undefined if  $|\psi_f\rangle = 0$ . Thus determining if the Rényi entropy is defined is NP-hard.  $\square$

We can then consider instead the promise problem called Unambiguous-SAT: Given a SAT instance promised to have at most one solution, determine if it is satisfiable. Unambiguous-SAT is still a hard problem.

**Theorem 3.22** (Valiant-Vazirani, [102]). *If there is a polynomial time algorithm for solving Unambiguous-SAT, then  $\text{NP} = \text{RP}$ .*

**Theorem 3.23** (Rényi Entropy Reduction to Unambiguous-SAT). *Suppose you have a Boolean state with the promise that the Rényi entropies are all undefined or zero. Then deciding if the Rényi Entropies are undefined or zero is as hard as Unambiguous-SAT.*

*Proof.* The Rényi entropies are zero if and only if the density operator is a product state, which implies that the corresponding Boolean state has a

unique solution. It is undefined if and only if it has no solution as previously discussed. Therefore, if we are promised the Rényi entropies are zero or undefined for all partitions, we are promised that there is at most one solution. This is precisely Unambiguous-SAT.  $\square$

What we see is that while Rényi entropies may encode information about the complexity of a counting problem, they are unfortunately not easy to compute. Furthermore, they may not even be defined and determining this is computationally challenging.

## Chapter 4

# Determinantal, Pfaffian, and Match Circuits

In the previous chapter, we looked at a general class of problems, phrased as tensor contraction problems, and gave conditions for when such problems could be solved efficiently. In this chapter, we take the opposite approach. We will study classes of tensor networks whose corresponding tensor contraction problem can always be solved in polynomial time. Then one can ask which problems can be solved by such a particular class of tensor networks.

The classes of tensor networks considered will be subcategories of  $\text{Vect}_{\mathbb{C}}$ . By a *circuit* we mean a combinatorial counting problem expressed as a string diagram in a monoidal subcategory of  $\text{Vect}_{\mathbb{C}}$ .

Let  $\mathcal{M}$  be a (strict) monoidal category such that  $S_{\mathcal{M}} = \text{Hom}_{\mathcal{M}}(\mathbb{1}_{\mathcal{M}}, \mathbb{1}_{\mathcal{M}})$  is a semiring; call this a *semiringed category*. A strict monoidal functor  $\mathbf{F} : \mathcal{M} \rightarrow \mathcal{M}'$  between semiringed categories is *count preserving* if the induced map  $\mathbf{F} : S_{\mathcal{M}} \rightarrow S_{\mathcal{M}'}$  is an injective morphism of semirings.

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  $\text{Vect}_{\mathbb{C}}$ . Then let  $i : \mathcal{L} \rightarrow \mathcal{C}$  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  $l \in \mathcal{L}$ ,  $i(l)$  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  $\text{Hom}(\mathbb{1}_{\mathcal{C}}, \mathbb{1}_{\mathcal{C}})$  is represented by an arbitrary monoidal word. We also have an interpretation  $f : \mathcal{L} \rightarrow \mathcal{S}$ . Then we want a monoidal

functor  $\mathbf{F}$  such that the diagram

$$\begin{array}{ccc} \mathcal{L} & \xrightarrow{i} & \mathcal{C} \\ & \searrow f & \downarrow \mathbf{F} \\ & & \mathcal{S} \end{array}$$

commutes and  $\mathbf{F}$  is a count preserving functor.  $\mathcal{S}$  is the subcategory of  $\text{Vect}_{\mathcal{C}}$  generated by the morphisms in the image of either  $\mathbf{F}$  or  $f$ . The induced maps on  $S_{\mathcal{C}}$  and  $S_{\mathcal{S}}$  make  $S_{\mathcal{C}}$  a sub-semiring of  $S_{\mathcal{S}}$ .

Of course, it is important that the interpretation  $i$  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.

Throughout this chapter, we use the following notation: in most cases we use  $M, N$  for matrices,  $\mathbf{I}, \mathbf{J}$  for sets (especially of indices),  $f, g$  for morphisms,  $\mathbf{F}, \mathbf{G}$  for functors, and  $\mathcal{C}, \mathcal{M}$  for categories. For a matrix  $X$ , we let  $X_{\mathbf{I}\mathbf{J}}$  be the submatrix with rows in  $\mathbf{I}$  and columns in  $\mathbf{J}$ .

## 4.1 Pfaffian circuits

As a first example, we consider the counting category (and corresponding subcategory of  $\text{Vect}_{\mathcal{C}}$ ) that defines Pfaffian Circuits. Pfaffian circuits were introduced in [78, 63] as a reformulation of matchcircuits [98]. Both prior formulations were combinatorial in nature. We present a slightly different construction using category theory to incorporate the theory into the tensor network formalism.

We first define the counting category for Pfaffian circuits, denoted  $\mathcal{P}$ . Consider the category of planar diagrams where every box is a morphism such that its domain or codomain equals the unit  $\mathbb{1}_{\mathcal{P}}$ . Notice that this allows the cup and cap morphisms, so this category will have duals for objects. The objects are vertical arrays of wires and can be labeled by elements of  $\mathbb{N}$ . Therefore, objects are identified with ordered subsets of  $\mathbb{N}$  and  $\mathbb{1}_{\mathcal{P}} = \emptyset$ .

Consider the set  $\mathcal{M} \times \{0, 1\}$ , where  $\mathcal{M}$  is the set of labeled skew-symmetric matrices such that the columns and rows have the same labels in the same order. Then every box in a diagram has a label, where morphisms  $\mathbf{J} \rightarrow \emptyset$  are labeled by an element of  $\mathcal{M} \times \{1\}$  with label set  $\mathbf{J}$ , and morphisms  $\emptyset \rightarrow \mathbf{J}$  are labeled by elements of  $\mathcal{M} \times \{0\}$  with label set  $\mathbf{J}$ . Henceforth, we refer to morphisms simply as  $(M, b)$  instead of labeled boxes.

We define the tensor product of this category as follows:

$$(a) \quad (M, b) \otimes (N, b) := (M \oplus N, b).$$

$$(b) \quad (M, 1) \otimes (N, 0) := (N, 0) \circ (M, 1).$$

Note that every diagram with no dangling edges can be collapsed to a diagram representing the map  $(M, 1) \circ (N, 0) : \emptyset \rightarrow \emptyset$ . This describes our counting category  $\mathcal{P}$ .

We now define a functor  $\text{sPf}$  from  $\mathcal{P} \rightarrow \text{Vect}_{\mathbb{C}}$ . For  $i \in \mathbb{N}$ , let  $V_i \cong \mathbb{C}^2$  be spanned by an orthonormal basis (with inner product)  $v_{i,0}, v_{i,1}$  and for  $\mathbb{N} \subset \mathbb{N}$  write  $V_{\mathbb{N}} := \bigotimes_{i \in \mathbb{N}} V_i$ . Now let us consider the following function:

$$\text{sPf} : \mathcal{M} \times \{0, 1\} \rightarrow V_{\mathbb{N}}^* \otimes V_{\mathbb{N}}$$

$$\text{sPf}(M, 0) = \sum_{\mathbb{I} \subseteq \mathbb{N}} \text{Pf}(M_{\mathbb{I}}) |\mathbb{I}\rangle$$

$$\text{sPf}(M, 1) = \sum_{\mathbb{I} \subseteq \mathbb{N}} \text{Pf}(M_{\bar{\mathbb{I}}}) \langle \mathbb{I} |$$

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

The  $\text{sPf}$  function lets us define a monoidal subcategory of  $\text{Vect}_{\mathbb{C}}$ . Let  $\mathcal{P}$  be the free monoidal category defined as follows. The objects are of the form  $V_{\mathbb{N}}$  for ordered subsets of  $\mathbb{N}$ , 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  $\text{Vect}_{\mathbb{C}}$ . It turns out that  $\text{sPf}$  is a functor that gives an equivalence of categories between  $\mathcal{P}$  and  $\mathcal{P}$  [78].

We now define a Pfaffian circuit to be a diagram in either of the equivalent categories  $\mathcal{P}$  or  $\mathcal{P}$  representing a map from  $\mathbb{1}_{\mathcal{P}} \rightarrow \mathbb{1}_{\mathcal{P}}$ , equivalently a map  $\mathbb{C} \rightarrow \mathbb{C}$ . Classically, the vectors  $\text{sPf}(M)$  are called *gates* and the covectors  $\text{sPf}^{\vee}(M)$  are called *cogates*. Note that any Pfaffian circuit  $T \in \mathcal{P}$  has the natural group action of  $\text{GL}_T$  as mentioned in Observation 2.9. Therefore, a natural extension of the definition of a Pfaffian circuit is a tensor network  $T$  that is in the  $\text{GL}_T$ -orbit of a tensor network in  $\mathcal{P}$ . A Pfaffian (co)gate is a (co)vector that is in the  $\text{GL}_T$ -orbit of a (co)gate in  $\mathcal{P}$ .

This is the original definition used in [98, 63, 78]. However, this means that determining whether or not a circuit is Pfaffian, and the general expressiveness of Pfaffian circuits, is much more difficult. But this action allowed

for new polynomial time algorithms to be found for problems for which no such algorithms were previously known [99, 103]. The problem of determining if a tensor network is a Pfaffian circuit falls into geometric invariant theory which we investigate in later chapters. However, for the remainder of this chapter, we restrict ourselves to Pfaffian circuits built up from the standard set of (co)gates previously described.

Now suppose we are given a Pfaffian circuit  $\Gamma$ . Let  $\Xi_i$  be the morphisms of the form  $\text{sPf}(M)$  and  $\Theta_i$  be the morphisms of the form  $\text{sPf}^\vee(M)$ . We define  $\Xi = \bigoplus_i \Xi_i$  and  $\Theta$  likewise. The operation  $\bigoplus$  is the direct sum with the row and columns reordered as follows: draw a planar curve through the Pfaffian circuit such that every edge is intersected by the curve exactly once. Since a Pfaffian circuit is planar and bipartite, such a curve always exists and the induced ordering 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  $\bigoplus$ . The matrix  $\check{\Theta}$  is defined to be  $\{(-1)^{i+j+1}\theta_{ij}\}$ . This ordering allows for the possibility that the Pfaffian circuits are not drawn with arrows drawn vertically, as classically has not been required. However, Theorem 4.2 tells us that this does not affect the well definedness of the value a Pfaffian circuit represents.

**Theorem 4.1.** *The value of a Pfaffian circuit  $\Gamma$  is given by  $\text{Pf}(\Xi + \check{\Theta})$  [78]*

Thus Pfaffian circuits can be computed in polynomial time. Furthermore, we will show that  $\mathcal{P}$  is a traced category, with the trace provided by the expression in Theorem 4.1. First, we investigate the structures of the category  $\mathcal{P}$ .

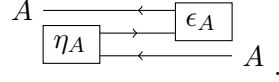
**Theorem 4.2.**  *$\mathcal{P}$  is a strict monoidal category with duals for objects.*

*Proof.* By our definition of  $\mathcal{P}$ , it will be the smallest monoidal subcategory of  $\text{Vect}_{\mathbb{C}}$  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. Recall from Theorem 2.6 that every monoidal category is equivalent to a strict one.

So we can assume without loss of generality that we are working with a strict category equivalent to  $\text{Vect}_{\mathbb{C}}$  instead. So the  $\alpha, \lambda$ , and  $\rho$  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  $A$ :

$$I_A = \begin{matrix} & A & A \\ A & \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \\ A & \end{matrix}.$$

Let  $\epsilon_A = \text{sPf}(I_A) = |0_A 0_A\rangle + |1_A 1_A\rangle$  and  $\eta_A = \text{sPf}^\vee(I_A) = \langle 0_A 0_A| + \langle 1_A 1_A|$ . Then we can contract these two morphisms along a single edge as in the following picture:



This gives us the morphism  $|0_A\rangle\langle 0_A| + |1_A\rangle\langle 1_A|$  which is the identity morphism on  $A$ . Furthermore,  $\epsilon_A$  and  $\eta_A$  are the cup and cap morphisms and we have just shown that they satisfy the yank axioms given by Equations 2.1. This shows that  $\mathcal{P}$  has duals for objects.  $\square$

**Definition 4.3.** The anti-transpose of a matrix  $N$ , denoted by  $\hat{N}$ , is  $N$  flipped across the non-standard diagonal.

**Lemma 4.4.**  $\text{Pf}(\hat{N}) = \text{Pf}(N)$ .

*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  $1, \dots, n$  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  $\hat{N}$  and suppose  $\pi \in \mathcal{F}$ . Then the mapping  $\mathcal{F} \rightarrow \mathcal{F} : \pi \mapsto \pi'$  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  $n, \dots, 1$  from left to right. This preserves the number of crossings of edges so that  $\text{sgn}(\pi') = \text{sgn}(\pi)$ . Thus we get

$$\begin{aligned} \text{Pf}(\hat{N}) &= \sum_{\pi \in \mathcal{F}} \text{sgn}(\pi) \prod_{(i_k, j_k)} \eta'_{i_k j_k} = \\ &= \sum_{\pi' \in \mathcal{F}} \text{sgn}(\pi') \prod_{(n-j_k+1, n-i_k+1)} \eta_{n-j_k+1, n-i_k+1} = \text{Pf}(N). \end{aligned}$$

$\square$

**Definition 4.5.** If  $l$  is a bitstring, let  $\tilde{l}$  be the bitstring reflected across a vertical axis. If  $l \subseteq N$ , where  $N$  is a finite ordered set,  $l$  can be viewed as a bitstring representing a characteristic function, where 1 in the  $i^{th}$  bit indicates inclusion of the  $i^{th}$  element of  $N$  in the set  $l$ . Then  $\tilde{l}$  is a characteristic function defining another subset of  $N$ . Then  $|\tilde{l}\rangle = \bigotimes_{i \in N} v_{i, \chi(i, \tilde{l})}$  and  $\langle \tilde{l}| = \bigotimes_{i \in N} v_{i, \chi(i, \tilde{l})}^*$

**Corollary 4.6.** Let  $N$  be a skew symmetric matrix with labels  $M$ . Let  $\hat{N}$  also have labels  $M$ .  $\text{sPf}(\hat{N}) = \sum_{l \subseteq M} \text{Pf}(N_l) |\tilde{l}\rangle$

*Proof.* Let  $l \subseteq M$ . Note that  $N_l = \hat{N}_{\tilde{l}}$ . Then  $\text{Pf}(N_l) = \text{Pf}(\hat{N}_{\tilde{l}})$ . This gives the result.  $\square$

**Example 4.7.** Consider the following matrix:

$$N = \begin{pmatrix} 0 & 0 & a & 0 \\ 0 & 0 & 0 & b \\ -a & 0 & 0 & 0 \\ 0 & -b & 0 & 0 \end{pmatrix}$$

$$\begin{aligned} \text{sPf}(\hat{N}) &= |0000\rangle + b|1010\rangle + a|0101\rangle - ab|1111\rangle \\ &= \text{Pf}(N_{\emptyset})|0000\rangle + \text{Pf}(N_{\{2,4\}})|1010\rangle + \text{Pf}(N_{\{1,3\}})|0101\rangle + \text{Pf}(N)|1111\rangle \\ &= \sum_{l \subseteq M} \text{Pf}(N_l) |\tilde{l}\rangle. \end{aligned}$$

**Proposition 4.8.** For any skew-symmetric matrix  $M$ ,

$$\begin{aligned} &\sum_l \text{Pf}(M_l) \langle l| \\ &\sum_l \text{Pf}(M_{\tilde{l}}) |l\rangle \end{aligned}$$

are morphisms of  $\mathcal{P}$ . This implies that  $\mathcal{P}$  is a dagger monoidal category.

*Proof.* Let  $M$  have labels  $A = \{A_1, \dots, A_n\}$ . Then  $\hat{M}$  will have labels  $\tilde{A} = \{A_n, \dots, A_1\}$ . Here the cup morphism  $\eta_A$  is given by:

$$\eta_A = \text{sPf} \begin{pmatrix} \tilde{A} & A \\ A & \tilde{I} \end{pmatrix}$$

where  $\tilde{I}$  is the identity matrix reflected over a vertical axis. Then consider the following morphism in  $\mathcal{P}$ :



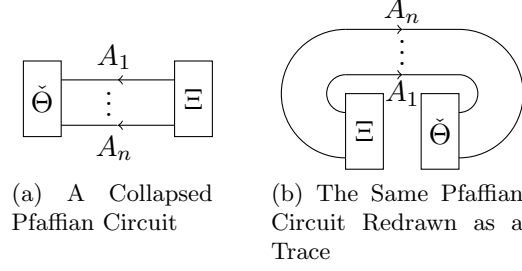
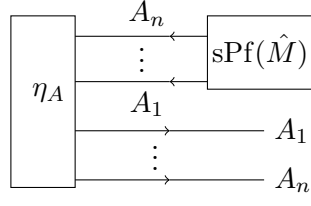


Figure 4.1: Viewing  $\mathcal{P}$  as a Traced Category.



This diagram represents the morphism

$$\left( \sum_{I \subseteq A} \text{Pf}(M_I) |\tilde{I}\rangle \right) \left( \sum_{I \subseteq \{\hat{A}A\}} \langle \tilde{I} | \langle I | \right) = \sum_{I \subseteq A} \text{Pf}(M_I) \langle I |.$$

We can similarly form  $\sum_I \text{Pf}(M_I) |I\rangle$  by instead using  $\text{sPf}^\vee(\hat{M})$  and the cap morphism  $\epsilon_A$ . Now since every generating morphism has a dagger, the entire category has a dagger and it is the usual vector space dagger.  $\square$

Now we can show that  $\mathcal{P}$  is a traced category. Recall that, as it is equivalent to the category  $\mathcal{P}$ , we can collapse every Pfaffian circuit to one of the form in Figure 4.1(a). But, because we have duals for objects in  $\mathcal{P}$  and  $\mathcal{P}$ , this can be recast as a trace, as in Figure 4.1(b). We know that this trace is equal to  $\text{Pf}(\Xi + \check{\Theta})$  in  $\mathcal{P}$ . Therefore, we define this to be the trace for  $\mathcal{P}$ .

## 4.2 Determinantal Circuits

We now formulate a class of circuits based on determinants, which we call determinantal circuits. We show that the corresponding tensor contraction

problem is solvable in polynomial time. The existence of such a class was conjectured in [63]. This was conjectured based on the following theorem:

**Theorem 4.9** ([22]). *Given a graph  $G$  endowed with an arbitrary orientation, let  $B$  be its incidence matrix. Then  $\det(I + BB^T)$  is the number of rooted spanning forests.*

For Pfaffian circuits, the polynomial time evaluation was due to the existence of a formula that exploited an exponential amount of cancellation in the sum resulting from the tensor contraction. Tensor networks are very well suited for counting structures in graphs and we have a formula for solving the tensor contraction problem in polynomial time already recommended to us:  $\det(I + X)$ , where  $X$  is the result of combining all of the gates in a determinantal circuit in some way.

Suppose  $X$  is an  $n \times m$  matrix of elements of  $\mathbb{C}$  with rows and columns labeled by finite disjoint subsets  $I$  and  $J$  of  $\mathbb{N} = \mathbb{Z}_{\geq 0}$ . As before, for  $i \in \mathbb{N}$ , let  $V_i = \mathbb{C}^2$  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  $\text{sDet}$  (which we later show to be a functor) by  $\text{sDet}(N) = V_N$  and

$$\begin{aligned} \text{sDet} : \text{Mat}_k(n, m) &\rightarrow V_N^* \otimes V_M \cong (\mathbb{C}^{2*})^{\otimes n} \otimes (\mathbb{C}^2)^{\otimes m} \\ \text{sDet}(X) &= \sum_{I \subset [n], J \subset [m]} \det(X_{IJ}) |I\rangle \langle J| \end{aligned}$$

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  $i \in I$ . Throughout this paper, we work with the understanding that  $\det(X_{IJ}) = 0$  if  $|I| \neq |J|$ .

This subdeterminant function  $\text{sDet}$  induces a strong monoidal functor  $\text{sDet} : \mathcal{C} \rightarrow \text{Vect}_{\mathbb{C}}$  from a counting category to a subcategory  $\mathcal{D} \subset \text{Vect}_{\mathbb{C}}$ . Let  $\mathcal{C}$  be the monoidal category described as follows freely generated from the following objects and morphisms. The objects of  $\mathcal{C}$  are finite ordered subsets of  $\mathbb{N}$  (which may have repeated elements), with monoidal product on objects defined by disjoint union. The morphisms are  $\mathbb{C}$ -valued matrices with rows and columns labeled by ordered 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  $N$ , 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 a 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  $\text{sDet}(M)$  for

$M$  a labeled matrix. Tensor product and composition/contraction are the usual operations.

**Proposition 4.10.**  $\mathcal{C}$  is a strict dagger symmetric monoidal category.

*Proof.* We first show that  $\otimes_{\mathcal{C}}$  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  $A, B \subset \mathbb{N}$ ,  $\text{id}_A \otimes_{\mathcal{C}} \text{id}_B = \text{id}_{A \otimes_{\mathcal{C}} B}$ . Now for morphisms  $W, X, Y, Z \in \text{Mor}(\mathcal{C})$ ,  $W \otimes_{\mathcal{C}} X \circ Y \otimes_{\mathcal{C}} Z = (W \oplus X)(Y \oplus Z) = WY \oplus XZ = (W \circ Y) \otimes_{\mathcal{C}} (X \circ Z)$ , so  $\otimes_{\mathcal{C}}$  is indeed a bifunctor.

The maps  $\alpha, \rho$ , and  $\lambda$  must be natural transformations. For  $A, B, C \in \text{Ob}(\mathcal{C})$ , the associator  $\alpha_{ABC} : (A \otimes_{\mathcal{C}} B) \otimes_{\mathcal{C}} C \rightarrow A \otimes_{\mathcal{C}} (B \otimes_{\mathcal{C}} C)$  is just equality by the associativity of matrix direct product. The unit for  $\mathcal{C}$ , denoted  $\mathbb{1}_{\mathcal{C}}$ , is the empty set. Then  $\lambda_A : \mathbb{1}_{\mathcal{C}} \otimes_{\mathcal{C}} A \rightarrow A$  and  $\rho_A : A \otimes_{\mathcal{C}} \mathbb{1}_{\mathcal{C}} \rightarrow A$  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 Theorem 2.2 commute. First let us check, for  $A, B \in \text{Ob}(\mathcal{C})$ :

$$\begin{array}{ccc} (A \otimes_{\mathcal{C}} \mathbb{1}_{\mathcal{C}}) \otimes_{\mathcal{C}} B & \xrightarrow{\alpha} & A \otimes_{\mathcal{C}} (\mathbb{1}_{\mathcal{C}} \otimes_{\mathcal{C}} B) \\ & \searrow \rho_A \otimes_{\mathcal{C}} \text{id}_B & \downarrow \text{id}_A \otimes_{\mathcal{C}} \lambda_B \\ & & A \otimes_{\mathcal{C}} B \end{array}$$

$(A \otimes_{\mathcal{C}} \mathbb{1}_{\mathcal{C}}) \otimes_{\mathcal{C}} B = (A \cup \emptyset) \cup B$  is mapped to  $A \cup B$  by  $\rho_A \otimes_{\mathcal{C}} \text{id}_B$  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  $\text{id}_A \otimes_{\mathcal{C}} \lambda_B$  via equality, and the diagram commutes.

Now let us check the second diagram, for  $A, B, C, D \in \text{Ob}(\mathcal{C})$  (as writing  $\otimes_{\mathcal{C}}$  simply as  $\otimes$ ):

$$\begin{array}{ccc} ((C \otimes A) \otimes B) \otimes D & \xrightarrow{\alpha \otimes \text{id}_D} & (C \otimes (A \otimes B)) \otimes D \\ \swarrow \alpha & & \downarrow \alpha \\ (C \otimes A) \otimes (B \otimes D) & & C \otimes ((A \otimes B) \otimes D) \\ \searrow \alpha & \xleftarrow{\text{id}_C \otimes \alpha} & C \otimes (A \otimes (B \otimes D)) \end{array}$$

The object  $((C \otimes_{\mathcal{C}} A) \otimes_{\mathcal{C}} B) \otimes_{\mathcal{C}} D = ((C \cup A) \cup B) \cup D$  is mapped to  $C \cup (A \cup (B \cup D))$  by  $(\text{id}_C \otimes_{\mathcal{C}} \alpha) \circ (\alpha) \circ (\alpha \otimes_{\mathcal{C}} \text{id}_D)$  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  $\mathcal{C}$  is a monoidal category. Furthermore, since  $\alpha, \lambda$ , and  $\rho$  are equalities,  $\mathcal{C}$  is a strict monoidal category.

The braiding for  $\mathcal{C}$  is a map  $\sigma_{A,B} : A \otimes_{\mathcal{C}} B \rightarrow B \otimes_{\mathcal{C}} A$ ,  $A, B \in \text{Ob}(\mathcal{C})$ . It is given by the matrix

$$\sigma_{A,B} = \begin{matrix} & B & A \\ \begin{matrix} A \\ B \end{matrix} & \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \end{matrix}.$$

We need to check that the diagrams in Subsection 2.1.1 for braided categories commute for  $A, B, C \in \text{Ob}(\mathcal{C})$  (once again writing  $\otimes$  in lieu of  $\otimes_{\mathcal{C}}$ ):

$$\begin{array}{ccccc} & (B \otimes A) \otimes C & \xrightarrow{\alpha} & B \otimes (A \otimes C) & \\ \sigma_{A,B} \otimes \text{id}_C \nearrow & & & & \searrow \text{id}_B \otimes \sigma_{A,C} \\ (A \otimes B) \otimes C & & & & B \otimes (C \otimes A) \\ \alpha \searrow & & & & \nearrow \alpha \\ & A \otimes (B \otimes C) & \xrightarrow{\sigma_{A,(B \otimes C)}} & (B \otimes C) \otimes A & \end{array}$$
  

$$\begin{array}{ccccc} & (B \otimes A) \otimes C & \xrightarrow{\alpha} & B \otimes (A \otimes C) & \\ \sigma_{A,B}^{-1} \otimes \text{id}_C \nearrow & & & & \searrow \text{id}_B \otimes \sigma_{A,C}^{-1} \\ (A \otimes B) \otimes C & & & & B \otimes (C \otimes A) \\ \alpha \searrow & & & & \nearrow \alpha \\ & A \otimes (B \otimes C) & \xrightarrow{\sigma_{A,(B \otimes C)}^{-1}} & (B \otimes C) \otimes A & \end{array}.$$

The first diagram commutes by noting that

$$\begin{array}{ccc} \begin{matrix} B & A & C \\ A \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} & B \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & B \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \end{matrix} = \\ \begin{matrix} A & B & C \\ A \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & A \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} & B \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{matrix} \end{array}$$

The second diagram commutes since  $\sigma_{B,A}^{-1} = \sigma_{A,B}$  (which implies the category is symmetric) for any  $A, B \in \text{Ob}(\mathcal{C})$  and so the second diagram is the same as the first.

The dagger for  $\mathcal{C}$  is given by matrix transpose and the identity on objects. We need to check the axioms for being a dagger symmetric monoidal category listed in Subsection 2.1.1. Clearly  $\text{id}_A^\dagger = \text{id}_A^T = \text{id}_A$ . Given  $X, Y \in \text{Mor}(\mathcal{C})$ ,  $X : A \rightarrow B$ ,  $Y : B \rightarrow C$ ,  $(X \circ Y)^\dagger = (XY)^T = Y^T X^T = Y^\dagger \circ X^\dagger : C \rightarrow A$ . Lastly  $X^{\dagger\dagger} = X^{TT} = X$ .

Given  $X, Y \in \text{Hom}(\mathcal{C})$ ,  $(X \otimes_{\mathcal{C}} Y)^\dagger = (X \oplus Y)^T = X^T \oplus Y^T = X^\dagger \otimes_{\mathcal{C}} Y^\dagger$ . 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  $\mathcal{C}$  is indeed a strict dagger symmetric monoidal category.  $\square$

**Theorem 4.11.** *The map  $\text{sDet}$  defines a strict monoidal functor which is an equivalence of dagger symmetric traced categories. Thus while computing a trace in  $\text{Vect}_{\mathbb{C}}$  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 4.12 and 4.13.

**Lemma 4.12.** *The map  $\text{sDet}$  defines a strict monoidal functor which is an equivalence of monoidal categories.*

*Proof.* 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  $X \in \text{Hom}_{\mathcal{C}}(I, J)$ ,  $Y \in \text{Hom}_{\mathcal{C}}(J, K)$  so  $X$  is a matrix with row labels  $I$ , column labels  $J$  and  $Y$  has row labels  $J$  and column labels  $K$ :

$$\begin{aligned} \text{sDet}(Y) \circ \text{sDet}(X) &= \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} \det(X_{ij}) \det(Y_{jk}) |i\rangle\langle k| \\ &= \sum_{i \in I} \sum_{j \in J} \det(XY_{ik}) |i\rangle\langle k| = \text{sDet}(XY) \end{aligned}$$

where the middle equality is the Cauchy-Binet formula. Now in  $\mathcal{C}$ ,  $\text{id}_A$  is the identity matrix with row and column labels  $A$ . Then  $\text{sDet}(\text{id}_A) = \sum_{I \subseteq A} |I\rangle\langle I|$  which is the identity morphism for the object  $\text{sDet}(A)$  in  $\mathcal{D}$ , so  $\text{sDet}$  is indeed a functor.

For  $\text{sDet}$  to be a monoidal functor, we must demonstrate two additional properties. We must show that for matrices  $M$  and  $N$ ,  $\text{sDet}(M \oplus N) = \text{sDet}(M) \otimes \text{sDet}(N)$ . Here  $\otimes$  denotes the usual tensor product in  $\text{Vect}_{\mathbb{C}}$ . Let  $I$  and  $J$  be the rows and columns of  $M$ , respectively. Let  $I'$  and  $J'$  be likewise for  $N$ . A straightforward calculation gives

$$\text{sDet}(M \oplus N) = \sum_{U \subseteq I \cup I'} \sum_{V \subseteq J \cup J'} \det(M \oplus N)_{UV} |U\rangle\langle V|$$

$$\begin{aligned}
&= \sum_{U \subseteq I \cup I'} \sum_{V \subseteq J \cup J'} \det(M_{U \cap I, V \cap J}) \det(N_{V \cap I', V \cap J'}) |U \cap I\rangle |U \cap I'\rangle \langle V \cap J| \langle V \cap J'| \\
&= \sum_{U \subseteq I} \sum_{U' \subseteq I'} \sum_{V \subseteq J} \sum_{V' \subseteq J'} \det(M_{UV}) \det(N_{U'V'}) |U\rangle |U'\rangle \langle V| \langle V'| = \text{sDet}(M) \otimes \text{sDet}(N).
\end{aligned}$$

Secondly we must show there are morphisms  $F_0 : \mathbb{1}_{\mathcal{D}} \rightarrow \text{sDet}(\mathbb{1}_{\mathcal{C}})$  and for any  $A, B \in \text{Ob}(\mathcal{C})$ ,  $F_1 : \text{sDet}(A) \otimes \text{sDet}(B) \rightarrow \text{sDet}(A \otimes_{\mathcal{C}} B)$  satisfying the commutative diagrams in Definition 2.4.

Since  $\text{sDet}(\emptyset) = \otimes_{i \in \emptyset} V_i = \mathbb{C}$ ,  $F_0$  is simply equality. Similarly for objects  $A$  and  $B$ ,

$$\begin{aligned}
\text{sDet}(A \otimes_{\mathcal{C}} B) &= \text{sDet}(A \cup B) = \otimes_{i \in A \cup B} V_i \\
&= (\otimes_{i \in A} V_i) \otimes (\otimes_{j \in B} V_j) = \text{sDet}(A) \otimes \text{sDet}(B).
\end{aligned}$$

Thus  $F_1$  is equality. In the following diagrams, we shall denote  $\text{sDet}$  by  $\mathbf{F}$ .

Let  $\alpha', \lambda', \rho'$  be the natural transformations for  $\mathcal{D}$ . Note that all three are equalities. For  $A, B, C \in \text{Ob}(\mathcal{C})$ , the following diagrams from Definition 2.4 must commute :

$$\begin{array}{ccc}
\mathbf{F}(A) \otimes (\mathbf{F}(B) \otimes \mathbf{F}(C)) & \xrightarrow{\alpha'} & (\mathbf{F}(A) \otimes \mathbf{F}(B)) \otimes \mathbf{F}(C) \\
\downarrow \text{id}_{\mathbf{F}(A)} \otimes F_1 & & \downarrow F_1 \otimes \text{id}_{\mathbf{F}(C)} \\
\mathbf{F}(A) \otimes (\mathbf{F}(B \otimes_{\mathcal{C}} C)) & & (\mathbf{F}(A \otimes_{\mathcal{C}} B) \otimes \mathbf{F}(C)) \\
\downarrow F_1 & & \downarrow F_1 \\
\mathbf{F}(A \otimes_{\mathcal{C}} (B \otimes_{\mathcal{C}} C)) & \xrightarrow{\mathbf{F}(\alpha)} & \mathbf{F}((A \otimes_{\mathcal{C}} B) \otimes_{\mathcal{C}} C)
\end{array}$$
  

$$\begin{array}{ccc}
\mathbf{F}(B) \otimes \mathbb{1}' & \xrightarrow{\rho'} & \mathbf{F}(B) \\
\downarrow \text{id}_{\mathbf{F}(B)} \otimes F_0 & & \uparrow \mathbf{F}(\rho) \\
\mathbf{F}(B) \otimes \mathbf{F}(\mathbb{1}) & \xrightarrow{F_1} & \mathbf{F}(B \otimes \mathbb{1})
\end{array}
\quad
\begin{array}{ccc}
\mathbb{1}' \otimes \mathbf{F}(B) & \xrightarrow{\lambda} & \mathbf{F}(B) \\
\downarrow F_0 \otimes \text{id}_{\mathbf{F}(B)} & & \uparrow \mathbf{F}(\lambda) \\
\mathbf{F}(\mathbb{1}) \otimes \mathbf{F}(B) & \xrightarrow{F_1} & \mathbf{F}(\mathbb{1} \otimes B)
\end{array}$$

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  $\mathcal{C}$  and  $\mathcal{D}$  are equivalent as monoidal categories. By definition of  $\mathcal{D}$ ,  $\text{sDet}$  surjects onto objects and morphisms, so it is a full functor. Now consider  $\text{Hom}(A, B)$  for objects  $A, B \in \text{Ob}(\mathcal{C})$ . 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  $\text{Hom}(A, B) \rightarrow \text{Hom}(\text{sDet}(A), \text{sDet}(B))$ , so 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 as all subsets of  $\mathbb{N}$  of size  $n$  map to  $(\mathbb{C}^2)^{\otimes n}$ .  $\square$

We have yet to define the braiding and dagger for  $\mathcal{D}$  required to state Theorem 4.11. For  $\mathbf{F} = \text{sDet}$  to respect the braiding, we need the following diagrams from Equation 2.1.2 to commute:

$$\begin{array}{ccc} \mathbf{F}(A) \otimes \mathbf{F}(B) & \xrightarrow{F_1} & \mathbf{F}(A \otimes B) \\ \sigma_{\mathbf{F}(A), \mathbf{F}(B)} \downarrow & & \downarrow \mathbf{F}(\sigma_{A,B}) \\ \mathbf{F}(B) \otimes \mathbf{F}(A) & \xrightarrow{F_1} & \mathbf{F}(B \otimes_{\mathcal{C}} A) \end{array} .$$

Recalling the matrix  $\sigma_{A,B}$  as defined in Theorem 4.10, we define the braiding for  $\mathcal{D}$  to be  $F(\sigma_{A,B}) = \text{sDet}(\sigma_{A,B}) = |00\rangle\langle 00| + |01\rangle\langle 10| + |10\rangle\langle 01| - |11\rangle\langle 11|$ , which makes the diagram commute trivially. We do not check the diagrams that ensures this is a valid braiding for  $\mathcal{D}$  since it is equivalent to  $\mathcal{C}$ . For the dagger, consider  $X \in \text{Hom}(\mathcal{C})$  with row labels  $\mathbf{l}$  and column labels  $\mathbf{j}$ , and note

$$\begin{aligned} \text{sDet}(X^\dagger) &= \sum_{\mathbf{i} \subseteq \mathbf{l}, \mathbf{j} \subseteq \mathbf{j}} \det(X_{\mathbf{ij}}^T) |\mathbf{i}\rangle\langle \mathbf{j}| = \\ &= \sum_{\mathbf{i} \subseteq \mathbf{l}, \mathbf{j} \subseteq \mathbf{j}} \det(X_{\mathbf{ij}}) |\mathbf{j}\rangle\langle \mathbf{i}| = \text{sDet}(X)^T. \end{aligned}$$

So the dagger for  $\mathcal{D}$  is the normal dagger in  $\text{Vect}_{\mathcal{C}}$ .

For  $f : A \rightarrow A \in \mathcal{C}$ , define  $\text{tr}(f) = \det(I + f)$  and define trace in  $\mathcal{D}$  in the usual way. This choice of trace may seem unusual, but it satisfies the axioms of a traced category given in Subsection 2.1.1 and its image under the  $\text{sDet}$  functor is the usual trace in  $\mathcal{D}$  (as we show in a moment). This is the most important aspect as it allows us to frame problems in  $\mathcal{C}$  and find the answer to the contraction problem without the need to pass over to the category  $\mathcal{D}$  which has exponentially larger tensors.

**Lemma 4.13.** *The map  $\text{sDet}$  defines a strict monoidal functor which is an equivalence of dagger symmetric traced categories.*

*Proof.* By construction,  $\text{sDet}$  respects the braiding. We also showed that this functor respects the normal dagger for linear transformations. Theorem 4.15 and Proposition 4.16 below shows that  $\text{sDet}$  induces the identity map from  $\text{Hom}(\mathbb{1}_{\mathcal{C}}, \mathbb{1}_{\mathcal{C}}) \rightarrow \text{Hom}(\mathbb{1}_{\mathcal{D}}, \mathbb{1}_{\mathcal{D}})$  and thus respects the trace.  $\square$

**Remark 4.14.** *This braiding is not the usual braiding for  $\text{Vect}_{\mathbb{C}}$ . Thus while the functor  $\text{sDet}$  is count-preserving, the count will not be the same as if the standard braiding  $u \otimes v \mapsto v \otimes u$  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  $k = 1, \dots, m$ . The maximum width of such a circuit is  $\max_{k=1, \dots, m} \log_2 d_k$  and the depth is  $m$ .

**Theorem 4.15.** *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  $1, \dots, n$ . Then

$$\text{sDet}(M) = \sum_{I, J \subseteq [n]} \det(M_{I, J}) |I\rangle\langle J|$$

and contracting this against itself gives

$$\sum_{I, J \subseteq [n]} \det(M_{I, J}) \langle J | I \rangle \langle J | I \rangle = \sum_{I \subseteq [n]} \det M_{I, I}.$$

That is, the trace of a matrix  $M$  in  $\mathcal{C}$  is the exponentially large sum of its  $2^n$  principal minors; we claim that  $\det(I + A)$  is precisely this sum (Proposition 4.16). This enables us to compute this number in time  $n^\omega$ .  $\square$

The following identity is well-known (e.g. it can be derived from results in [50]); we include a proof for completeness.

**Proposition 4.16.** *Given an  $n \times n$  matrix  $M$ ,*

$$\det(I + M) = \sum_{J \subseteq [n]} \det(M_J)$$

*where  $M_J = M_{J, J}$ .*



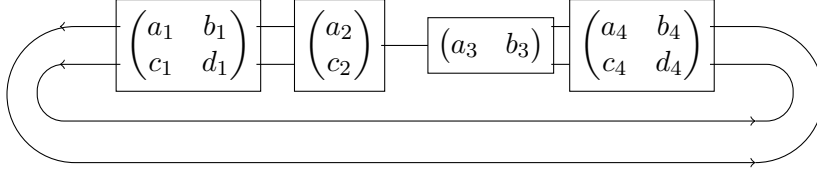


Figure 4.2: An example of a determinantal circuit (wires oriented clockwise). The four tensors in  $\text{Vect}_{\mathbb{C}}$ , from left to right, are obtained by applying  $\text{sDet}$  to each matrix. Letting  $V = \mathbb{C}^2$ , they lie in  $(V^*)^{\otimes 2} \otimes V^{\otimes 2}$ ,  $(V^*)^{\otimes 2} \otimes V$ ,  $V^* \otimes V^{\otimes 2}$ , and  $(V^*)^{\otimes 2} \otimes V^{\otimes 2}$  respectively.

*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,  $\det(W) = \det(W_{\bar{j}})$ , where  $W_{\bar{j}}$  is  $W$  with the  $j$ th row and column omitted. Then iterating the Laplace expansion gives us that  $\det(W) = \det(M_J)$ .  $\square$

Note that while  $\mathcal{D}$  could be equipped with cup and cap morphisms from the category of finite-dimensional vector spaces to obtain a dagger closed compact category, the matrix category  $\mathcal{C}$  is *not* a closed compact category: it lacks the morphisms  $\eta_A$  and  $\epsilon_A$ . The morphism  $e_A : A \otimes A^* \rightarrow \mathbb{C}$  would have to be the  $\text{sDet}$  of a  $2 \times 0$  matrix, or the composition of several morphisms to obtain one of this type.

**Proposition 4.17.** *The category  $\mathcal{C}$  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  $2 \times 0$  matrix  $M$  and  $\text{sDet}(M) = |00\rangle$ .  $\square$

As a consequence, we really do have to work with traced categories rather than the more convenient dagger closed compact categories.

A diagram in the equivalent categories  $\mathcal{C}, \mathcal{D}$  is called a *determinantal circuit*, an example is given in Figure 4.2. When the morphism represented is a field element, it computes the value of the tensor contraction problem. We discuss applications of determinantal circuits further in Chapter 5.

### 4.3 Determinantal Circuits are Pfaffian Circuits

In looking to find a new class of efficiently tractable tensor networks, we found that surprisingly, all determinantal circuits are Pfaffian circuits. The converse is not true, however. To show this, we find 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 so that  $\mathcal{C}$  is a subcategory of  $\mathcal{P}$ , consequently  $\mathcal{D}$  is a subcategory of  $\mathcal{P}$ .

The morphisms in  $\mathcal{D}$  from  $V_n \rightarrow V_m$  are isomorphic to the variety

$$D_{n,m} := \{(1, \dots, \det(M_{IJ}), \dots, \det(M)) | M \in \text{Mat}_{n \times m}\}$$

given by tuples of minors of  $n \times m$  matrices. Then define the variety

$$P_n := \{(1, \dots, \text{Pf}(M_{IJ}), \dots, \text{Pf}(M)) | M \in \mathcal{M}_n\},$$

the tuples of minors of  $n \times n$  skew-symmetric matrices. This variety is isomorphic to the image of the sPf functor on the set  $\mathcal{M}_n \times \{0\}$ . We first want to find a closed immersion  $D_{n,m} \hookrightarrow P_{n+m}$ .

We can assume that  $n = m$ , otherwise, we pad the matrix with columns or rows of zeros as necessary. So we want to find a map  $D_{n,n} \hookrightarrow P_{2n}$ . For an  $n \times n$  matrix  $M$ , the following formula is classically known:

$$\text{Pf} \begin{bmatrix} 0 & M \\ -M^T & 0 \end{bmatrix} = (-1)^{n(n-1)/2} \det(M).$$

This embedding of  $M$  into a skew-symmetric matrix is close to the map we are looking for, however this naïve way may change the sign on some of the minors of  $M$ . So we must modify this map slightly.

**Definition 4.18.** Let  $\tilde{M}$  be the matrix  $M$  reflected across a vertical axis, and define  $S(M)$  to be

$$S(M) := \begin{bmatrix} 0 & \tilde{M} \\ -\tilde{M}^T & 0 \end{bmatrix}.$$

If  $M$  has row labels  $R$  and column labels  $C$ , then we give  $S(M)$  the row and column labels  $R \cup \tilde{C}$ , where  $\tilde{C}$  is the reverse ordering of  $C$ .

**Proposition 4.19.** 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  $\lfloor \frac{n}{2} \rfloor$  column swaps. So if  $n \equiv 0, 1$  modulo 4,  $\lfloor \frac{n}{2} \rfloor$  is an even number and so  $\det(\tilde{M}) = \det(M)$ . Now if  $n$  is congruent to 0 or 1 modulo 4, then  $\text{Pf}(S(M)) = (-1)^{n(n-1)/2} \det(\tilde{M}) = \det(\tilde{M}) = \det(M)$ . If  $n$  is congruent to 2 or 3 modulo 4, then  $\lfloor \frac{n}{2} \rfloor$  is an odd number so  $\det(\tilde{M}) = -\det(M)$  and  $\text{Pf}(S(M)) = (-1)^{n(n-1)/2} \det(\tilde{M}) = -\det(\tilde{M}) = \det(M)$ .  $\square$

We must turn this closed immersion into a functor. The morphisms of  $\mathcal{D}$  and  $\mathcal{P}$  look quite different. Note that there are two primary types of morphisms in  $\mathcal{P}$ , namely those of the form  $\text{sPf}(M)$  and those of form  $\text{sPf}^\vee(M)$ . Thus Pfaffian circuits form bipartite graphs. Determinantal circuits, on the other hand, are not bipartite at all. There are morphisms from  $V_n \rightarrow V_m$  for any sets  $n$  and  $m$  of any size.

Given how different these circuits look on the surface, we must really look at the categorical properties of  $\mathcal{P}$  to construct the functor. We will need the ability to bend wires in Pfaffian circuits. So we will exploit the fact that  $\mathcal{P}$  is a monoidal category with a dagger structure and duals for objects, as proved in Theorem 4.2 and Proposition 4.8.

**Theorem 4.20.** *Every morphism in  $\mathcal{D}$  is a morphism in  $\mathcal{P}$ . Thus there is a trace-preserving faithful strict monoidal functor from  $\mathcal{D} \rightarrow \mathcal{P}$  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

$$\text{Pf}(S(M)_K) = \text{Pf} \begin{bmatrix} 0 & \tilde{M}_{I,J} \\ -\tilde{M}_{I,J}^T & 0 \end{bmatrix} = \det(M_{I,J}),$$

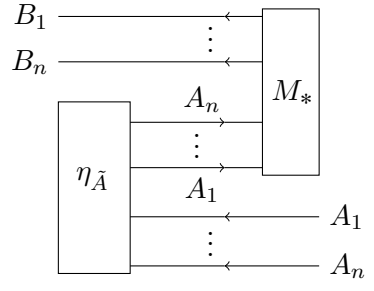
so that

$$\begin{aligned} \text{sPf}(S(M)) &= \sum_{I \subseteq R, J \subseteq C} \det(M_{I,J}) |I\rangle |\tilde{J}\rangle \\ \text{sPf}^\vee(S(M)) &= \sum_{I \subseteq R, J \subseteq C} \det(M_{\bar{I}, \bar{J}}) \langle I| \langle \tilde{J}|. \end{aligned}$$

The identity morphism on  $\tilde{A} := A_n \otimes \cdots \otimes A_1$  in  $\mathcal{C}$  is given by the matrix

$$I_{\tilde{A}} = \begin{matrix} & A_n & A_{n-1} & \cdots & A_1 \\ \begin{matrix} A_n \\ A_{n-1} \\ \vdots \\ A_1 \end{matrix} & \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \end{matrix}.$$

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



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

$$\left( \sum \det(M_{I,J}) |I\rangle |\tilde{J}\rangle \right) \left( \sum \langle \tilde{J}' | \langle J' | \right) = \sum \det(M_{I,J}) |I\rangle \langle J| = \text{sDet}(M).$$

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  $\text{sPf}([0]) = |0\rangle$  and thus  $\langle 0|$  is also a morphism in  $\mathcal{P}$ . Consider  $\text{sPf}^\vee(K) = \langle 0_A 0_B| + \langle 1_A 1_B|$  where

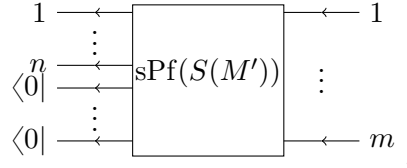
$$K = \begin{matrix} & A & B \\ \begin{matrix} A \\ B \end{matrix} & \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \end{matrix},$$

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

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

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



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.  $\square$

## Chapter 5

# The Combinatorics of Determinantal Circuits

In this chapter, we discuss how to determine what a determinantal circuit is counting in terms of its underlying graph. Knowing this allows us to phrase counting problems as determinantal circuits.

We discuss two applications of determinantal circuits in particular. In Section 5.2, we describe how to set up a determinantal circuit that counts the number of rooted spanning forests of a specified graph. We then show that collapsing this determinantal circuit to a single matrix yields that the number of rooted spanning forests is  $\det(I + BB^T)$ , recovering Theorem 4.9.

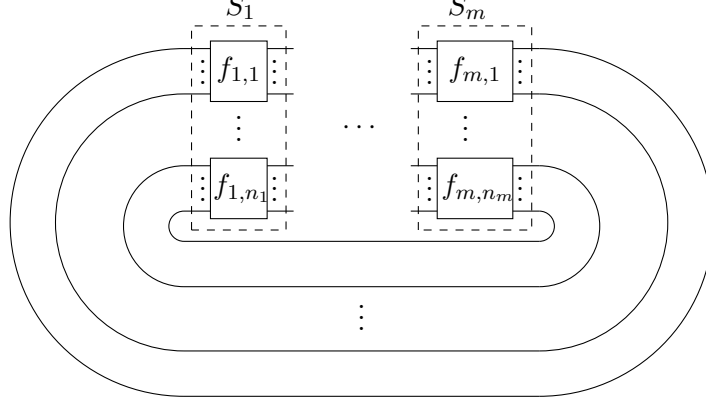
In Section 5.3, we define lattice path matroids. It is known that computing the Tutte polynomial of such a matroid can be done in polynomial time. However, using determinantal circuits we can make an improvement to this algorithm.

### 5.1 Equivalence Classes of Multicycles

We describe what the tensor contraction of a determinantal circuit counts in the most general setting. Considering the underlying weighted graph of the determinantal circuit, the value can be phrased as summing the weights associated to certain structures within the graph. These structures are equivalence classes of multicycles, which we define momentarily. Thus, when trying to count with determinantal circuits, one needs to embed the objects to be counted as equivalence classes of multicycles.

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  $\text{sDet}(M^{S_i}) = S_i$ . We call the  $S_i$  or associated  $M^{S_i}$  *stacks*. Pictorially, the situation is as follows:



Forgetting, for a moment, the categorical structure of the circuit, we consider the above as a graph.

**Definition 5.1.** A *multicycle* of a graph is an edge-disjoint union of cycles in the graph. We consider the empty graph a multicycle.

What matters is 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 5.2.** 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 5.3.** 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

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

A general determinantal circuit is the trace of a composition of stacks  $S_1 \circ \dots \circ S_m$ . Let  $E_k$  be the set of edges entering  $S_k$  from the right and

exiting  $S_{k-1}$  to the left, and observe that

$$\begin{aligned} \text{Tr}(\text{sDet}(M^{S_1} \circ \dots \circ M^{S_m})) &= \\ \text{Tr}\left(\sum_{\mathbf{l}_1, \dots, \mathbf{l}_m} \left[ \prod_{k=1}^m \det(M_{\mathbf{l}_k, \mathbf{l}_{k+1}}^{S_k}) \right] |\mathbf{l}_1\rangle\langle\mathbf{l}_2| |\mathbf{l}_2\rangle\langle\mathbf{l}_3| \dots \langle\mathbf{l}_m| \mathbf{l}_m\rangle\langle\mathbf{l}_1| \right) &= \\ \sum_{\mathbf{l}_1, \dots, \mathbf{l}_m} \left[ \prod_{k=1}^m \det(M_{\mathbf{l}_k, \mathbf{l}_{k+1}}^{S_k}) \right] & \end{aligned} \quad (5.1)$$

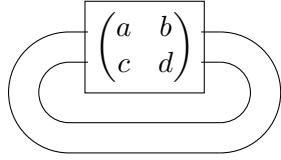
where each  $\mathbf{l}_k \subseteq E_k$ .

We want to describe (5.1) as a sum over equivalence classes of multicycles of  $S_1 \circ \dots \circ S_m$ . Consider the subgraph of the determinantal circuit whose edges are those in the sets  $\mathbf{l}_k$ . We claim that if the subgraph does not correspond to an equivalence class of multicycles,  $\prod \det(M_{\mathbf{l}_k, \mathbf{l}_{k+1}}^{S_k}) = 0$ .

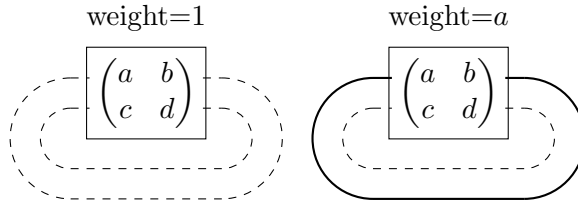
Each summand  $\prod \det(M_{\mathbf{l}_k, \mathbf{l}_{k+1}}^{S_k})$  in (5.1) will be non-zero only if  $|\mathbf{l}_1| = \dots = |\mathbf{l}_m|$  as the determinant of a non-square matrix is zero. This implies that the number of edges entering a vertex from the left in the underlying graph must equal the number of edges exiting it to the right. This is sufficient for the circuit subgraph given by the subsets  $\mathbf{l}_k$  to be viewable as a multicycle.

We have not specified a cycle decomposition of the multicycle, so each summand corresponds to an equivalence class of multicycles with weight  $\prod \det(M_{\mathbf{l}_k}^{S_k})$ .  $\square$

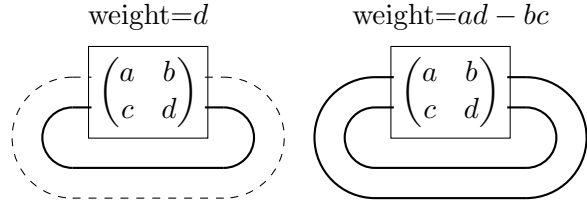
**Example 5.4.** Suppose we are given the following determinantal circuit:



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:







## 5.2 Recovering the matrix tree theorem

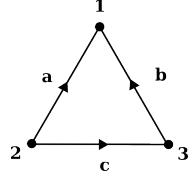
In this section, we describe the construction of a determinantal circuit from a given graph,  $G = \{V, E\}$ , such that its value counts the number of rooted spanning forests of  $G$ . Necessary for the theorem, choose an arbitrary orientation on the graph  $G$ . Let  $B$  be the directed incidence matrix of  $G$ .

We construct a string diagram  $ZZ^\dagger$  in  $\mathcal{C}$  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 5.1(a) and the determinantal circuit constructed for it in Figure 5.1(b).

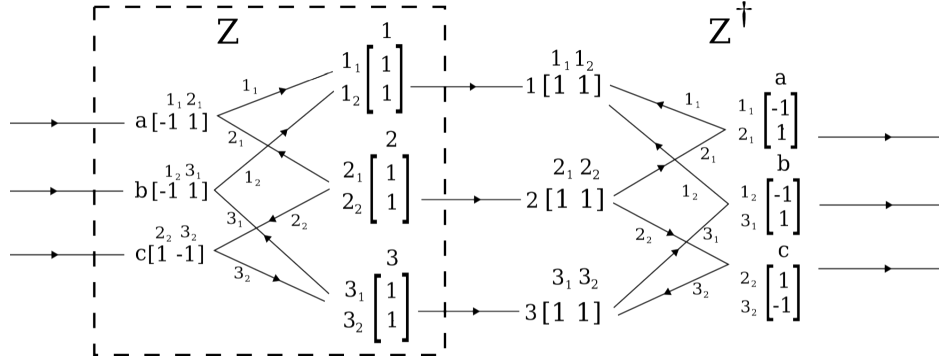
We first build a string diagram,  $Z$ , from a collection of  $\mathcal{C}$ -morphisms (nodes); there is one node for every edge and vertex of  $G$ . Denote an edge of  $G$  by  $\epsilon$ , the edge node in  $Z$  corresponding to it by  $e$  and the edge node in  $Z^\dagger$  corresponding to it by  $e^\dagger$ . 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  $G$ .

We induce an orientation on  $Z$  from the orientation placed on  $G$ , although this has no categorical meaning. 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 5.1(b) gives an example of this construction.

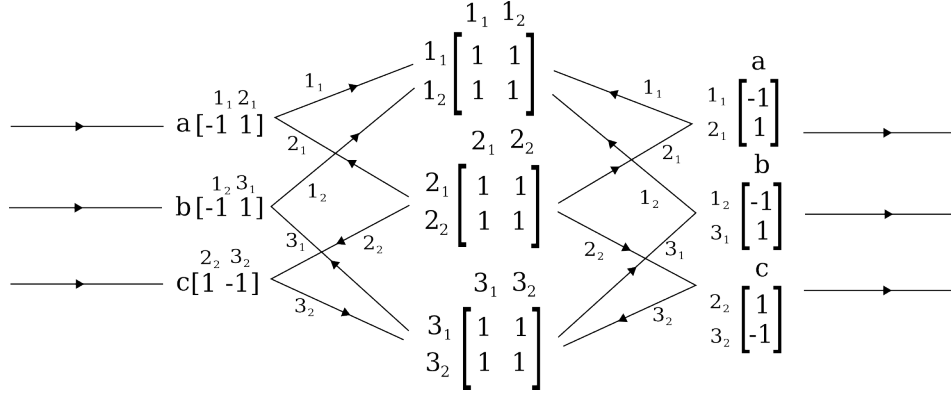
Edge nodes are  $1 \times 2$  matrices, vertex nodes are  $d(\nu) \times 1$  matrices, where  $d(\nu)$  is the degree of  $\nu$ . The matrix  $M_e$  associated to an edge node  $e$  in  $Z$  is either  $[1 \ -1]$  or  $[-1 \ 1]$ ; it has a  $-1$  in the column corresponding to the output wire oriented 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.



(a) A rooted graph



(b) Its corresponding circuit



(c) Equivalent circuit

Figure 5.1: Transforming a rooted graph to a determinantal circuit.

**Lemma 5.5.** *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  $v \rightarrow e$  minus the number of wires  $e \rightarrow v$  in  $Z$ . This implies that  $A = B$ , the incidence matrix.  $\square$

Reflect  $Z$  across a vertical line, transposing all node matrices, to obtain  $Z^\dagger$ , which collapses to the matrix  $B^T$ . Our final circuit  $ZZ^\dagger$  is the composition of  $Z$  with  $Z^\dagger$ . Figures 5.1(a) and (b), show an example of a graph  $G$  and its transformation into a circuit  $ZZ^\dagger$ . We denote the determinantal circuit like in Figure 5.1(b) associated to a graph  $G$  by  $D_G$ .

By analyzing the values of the multicycle classes of  $D_G$  and what they represent in the graph  $G$ , one can arrive at Theorem 4.9, although the proof via this method is quite tedious. We briefly sketch the correspondence, although we do not prove it.

There is a several to one, surjective map  $\phi$  from weighted multicycles classes of  $D_G$  to subgraphs of  $G$ . This is given by looking at the subgraph induced by the edge nodes included in the multicycle class. For a subgraph  $H$  that is not a forest,  $\phi^{-1}(H)$  contains an even number of multicycles, half with weight  $+1$  and half with weight  $-1$ . Thus subgraphs that are not forests do not contribute to the sum. If  $H$  is a forest, then  $\phi^{-1}(H)$  has as many multicycle classes as choice of roots for the forest. Furthermore, the weight of every multicycle class in  $\phi^{-1}(H)$  is  $+1$ . Every forest can be lifted to a unique spanning forest by adding the vertices of  $G$  not included in  $H$  (but none of their incident edges) and making all of the added vertices roots. This is sufficient to prove Theorem 4.9.

### 5.3 Computing the Tutte Polynomial of Lattice Path Matroids

Lattice path matroids were introduced in [12] as a particularly well-behaved and yet very interesting class of matroids. In the same paper, the authors prove that computation of the Tutte polynomials of lattice path matroids is

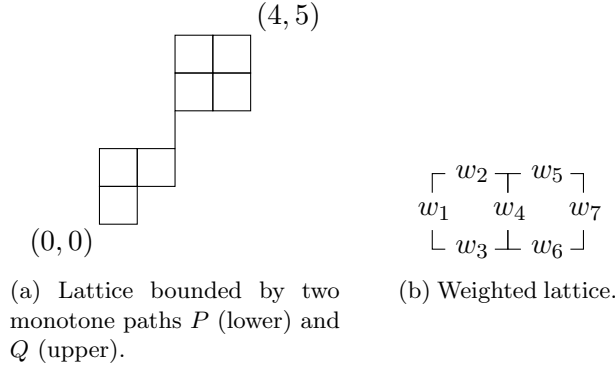


Figure 5.2: Lattices

polynomial time. In [13], they go on to prove that the time complexity of computing the Tutte polynomial is  $O(n^5)$ , where  $n$  is the size of the ground set of the matroid.

First we discuss weighted lattice paths and their relation to determinantal circuits, specifically as a special case of weighted equivalence classes of multicycles. Then we define lattice path matroids and state the relevant theorems from [12, 13]. Lastly, we give an explicit algorithm for computing the Tutte polynomial and analyze its time complexity. We give an algorithm that improves the complexity to  $O(n^4)$  using determinantal circuits. Then we show that evaluating the Tutte polynomial on a specific input (fixed values of  $x$  and  $y$ ) can be done in time  $O(n^2)$ .

### 5.3.1 Weighted Lattice Paths

Let us consider  $\mathbb{Z}^2$  as an infinite graph where two points are connected if they differ by  $(\pm 1, 0)$  or  $(0, \pm 1)$ . Suppose we are given two monotone paths on  $\mathbb{Z}^2$ ,  $P$  and  $Q$ , that both start at  $(0, 0)$  and end at  $(m, r)$ . Furthermore, suppose that  $P$  is never above  $Q$  in the sense that there are no points  $(p_1, p_2) \in P$ ,  $(q_1, q_2) \in Q$  such that  $p_1 - q_1 < 0$  and  $p_2 - q_2 > 0$ . We are interested in subgraphs of  $\mathbb{Z}^2$  bounded by such pairs of paths. From here on out, “lattice” means of subgraph of this form. An example is given by Figure 5.2(a).

Let  $E$  be the set of edges of a lattice  $G$ . Suppose for each  $e \in E$ , we assign it a weight,  $w(e)$ . We call this a *weighted lattice*. Given a monotone path  $C \subseteq G$ , we define the weight of  $C$  to be the product of its edge weights

$$w(C) := \prod_{e \in C} w(e).$$

**Definition 5.6.** Let  $G$  be a lattice bounded by two paths with common endpoint  $(m, r)$ . A *full path* in  $G$  is monotone path from  $(0, 0)$  to  $(m, r)$ .

**Definition 5.7.** Let  $\mathcal{F}$  be the set of full paths of a weighted lattice  $G$ . The *value* of  $G$  is defined to be

$$\sum_{C \in \mathcal{F}} w(C).$$

In Figure 5.2(b), there are three full paths of the weighted lattice. The value of this lattice is  $w_1 w_2 w_5 + w_3 w_4 w_5 + w_3 w_6 w_7$ .

We assign matrices to every vertex of a weighted lattice to encode the weights of each edge that will be the  $\mathcal{C}$ -morphisms for a determinantal circuit. Let  $G = \{V, E\}$  be a weighted lattice. For  $v \in V$ , we define the *incoming edges* of  $v$  to be those edges below or to the left of  $v$  incident to  $v$ . The other edges incident to  $v$  are the *outgoing edges* of  $v$ .

The matrix we associate to  $v$  has rows equal to the number of incoming edges and columns equal to the number of outgoing edges. We order the incoming edges of  $v$  counter-clockwise starting with the incoming edge closest to the negative  $x$ -axis. We order the outgoing edges of  $v$  clockwise starting with the outgoing edge closest to the positive  $y$ -axis. This order defines how to associate the edges of  $v$  with rows and columns of the matrix. We fill each column with the weight of the outgoing edge of  $v$  it corresponds to. For example:

$$\begin{array}{c} | \\ e_3 \\ | \\ - e_1 - \boxed{v} - e_4 - \mapsto \\ | \\ e_2 \\ | \end{array}$$

$$\begin{array}{cc} e_3 & e_4 \\ e_1 & \begin{pmatrix} w(e_3) & w(e_4) \\ w(e_3) & w(e_4) \end{pmatrix} \\ e_2 & \end{array}$$

Note that every edge in a lattice is the outgoing edge of precisely one vertex, so given the matrices associated to the vertices, the weight on the edges can be recovered. For a vertex  $v$ , we denote the matrix associated with it by  $M_v$ .

However, for the matrices associated with  $(0, 0)$  and  $(m, r)$ , we do something slightly different. As defined,  $M_{(0,0)}$  would be have zero rows and

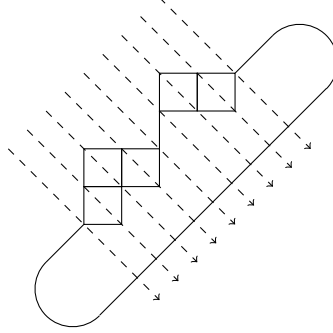


Figure 5.3: A Determinantal Circuit

$M_{(m,r)}$  zero columns. We define  $M_{(0,0)}$  to have one row with the weights of the outgoing edges in the appropriate columns. We define  $M_{(m,r)}$  to have one column, with all entries 1.

### 5.3.2 Phrasing the Tutte Polynomial as a Determinantal Circuit

Take a weighted lattice with the matrices described above assigned to each vertex. Figure 5.3.2 shows an example of how to determine the stacks (as described in Section 5.2) of the determinantal circuit from the lattice. For each of the diagonal arrows, take the matrix direct sum of the matrices  $M_v$  along the direction of the arrow, i.e.,  $M_v$  will be block diagonal. If  $G$  is a weighted lattice, let  $D_G$  be its associated determinantal circuit.

Let  $G$  be a weighted lattice bounded by two paths  $P$  and  $Q$  from  $(0,0)$  to  $(m,r)$ , and  $\mathcal{F}$  the set of full paths of  $G$ . If  $C \in \mathcal{F}$ , we describe it as a series of triples  $(v_i, e_i, e_{i+1})$  where  $e_i$  is the  $i$ th edge and  $v_i$  is the common vertex of  $e_i$  and  $e_{i+1}$ ,  $i$  ranging from 1 to  $n = m + r$ . The following is a special case of Proposition 5.3.

**Corollary 5.8.** *The value of  $D_G$  is given by the expression*

$$\sum_{C \in \mathcal{F}} \left( \det(M_{v_1, 1, e_1}) \det(M_{v_n, e_n, 1}) \prod_{(v_i, e_i, e_{i+1}) \in P} \det(M_{v_i, e_i, e_{i+1}}) \right)$$

where  $M_{v_i, e_i, e_{i+1}}$  is the minor of  $M_{v_i}$  specified by the edges  $e, e'$ .

**Proposition 5.9.** *The value of  $D_G - 1$  is equal to the value of  $G$ .*

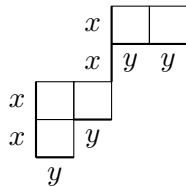


Figure 5.4: Weighted lattice with Tutte polynomial  $(x^2 + xy + y^2 + x + y)(x)(x + y + y^2)$  as its value.

*Proof.* By the way we constructed  $M_{v_i}$ ,  $M_{v_i, e_i, e_{i+1}}$  is a  $1 \times 1$  minor with entry  $w(e_{i+1})$ . Furthermore  $M_{v_1, 1, e_1}$  has single entry  $w(e_1)$  and  $M_{v_n, e_n, 1}$  has single entry 1. Thus

$$\begin{aligned} & \sum_{C \in \mathcal{F}} \left( \det(M_{v_1, 1, e_1}) \det(M_{v_n, e_n, 1}) \prod_{(v_i, e_i, e_{i+1}) \in P} \det(M_{v_i, e_i, e_{i+1}}) \right) \\ &= \sum_{C \in \mathcal{F}} \prod_{i=1}^n w(e_i) = \sum_{C \in \mathcal{F}} w(C). \end{aligned}$$

However, the case of the empty path is counted in this value, so if we subtract one from the value of  $D_G$ , we get the value of  $G$ .  $\square$

### 5.3.3 Lattice Path Matroids

Recall that a *matroid* is a pair  $(G, I)$  where  $G$ , the ground set, is a finite set and  $I$ , the independent sets, are a collection of subsets of  $G$  such that (i)  $\emptyset \in I$ , (ii) if  $A \in I$  and  $A' \subset A$ , then  $A' \in I$ , and (iii) if  $A, B \in I$  and  $|A| > |B|$ , then  $\exists a \in A \setminus B$  such that  $B \cup \{a\} \in I$ .

Lattice path matroids are defined with respect to a lattice bounded by two monotone paths  $P$  and  $Q$  from  $(0, 0)$  to  $(m, r)$  as described in Subsection 5.3.1. Since  $P$  and  $Q$  are both monotone, they can be described by a string of  $n = m + r$  0's and 1's where '1' corresponds to moving up one step and '0' corresponds to moving east one step.

**Definition 5.10** ([12]). Let  $P = p_1 \cdots p_n$  and  $Q = q_1 \cdots q_n$  be two lattice paths from  $(0, 0)$  to  $(m, r)$  with  $P$  never going above  $Q$ . Let  $u_1 < \cdots < u_r$  be the set of indices such that  $p_{u_i} = 1$ . Let  $l_1 < \cdots < l_r$  be similarly for  $Q$ . Then let  $N_i$  be the interval  $[l_i, u_i]$  of integers. Define  $M[P, Q]$  to be the matroid with ground set  $[n]$  and presentation  $(N_i : i \in [r])$ . A *lattice path matroid* is any matroid isomorphic to some  $M[P, Q]$ .

Lattice path matroids are a very nice example of matroids and of interest is the Tutte polynomial of such matroids. It turns out that this can be given as the value of a particular weighting of the lattice defining the matroid. The following is a slight restatement of the original theorem:

**Theorem 5.11** ([12]). *The Tutte polynomial of a lattice path matroid  $M[P, Q]$  is the value of the lattice  $G$  defined by  $P$  and  $Q$  with the north steps of  $Q$  having weight  $x$ , the east steps of  $P$  having weight  $y$ , and all other lattice weights equal to 1.*

An example is shown in Figure 5.4. Beside the edges in bold are the corresponding weights. The weight of each non-bold edge is simply 1.

### 5.3.4 Algorithm for Computing the Tutte Polynomial

We now give the algorithm for computing the Tutte polynomial in pseudocode. We assume that we are given as input two monotone paths  $P$  and  $Q$  from  $(0, 0)$  to  $(m, r)$  such that  $P$  never goes above  $Q$ . These paths are given to us as a list of 1's and 0's where a '1' denotes a step north and a '0' denotes a step east.

```

Distance(P,Q,i):
    Find the number of matrices in the  $i^{th}$  stack. This is
    how many more north steps  $Q$  has made than  $P$  by the  $i^{th}$  stack.

M(v):
    For input vertex  $v$ , return  $M_v$ .

Tutte(P,Q):
    Let  $T$  be the length-one row vector (1)
    for each stack  $i$ :
        Let  $A$  be the empty list
        for each node in the stack,  $v$ :
            Append  $M(v)$  to  $A$ 
        Update the vector  $T = (T_1A[1], \dots, T_iA[i], \dots)$ 
    After the last stack,  $T$  is 1 by 1,  $T=(t)$ .
    Return  $t$ 

```

Let  $n = m + r$  be the length of  $P$  and  $Q$ . The function **Tutte** iterates over the stacks. The function **Distance** calculates the number of lattice



points in a stack. For each vertex in the stack, the matrix  $M(v)$  associated to the vertex is found by a constant-time lookup.

After the matrices are calculated, we could take their direct sum and multiply it by  $T$ . For example, the matrices associated to the nine stacks in Figure 5.4, each of which is a direct sum of  $M(v)$ , are multiplied to obtain

$$(x \ y) \begin{pmatrix} x & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & y \\ 0 & 1 & y \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ x \end{pmatrix} (x \ y) \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & y \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix},$$

which equals the Tutte polynomial  $(x^2 + xy + y^2 + x + y)(x)(x + y + y^2)$ .

We can be more careful, however, to improve the running time since each matrix  $S_i$  is block-diagonal with block size at most 2.

$T$  starts out as a (row) vector and each iteration of the main loop updates  $T$  to a new vector. Since at each stage we are multiplying  $T$  by a block diagonal matrix, we can partition  $T$  into  $T = (T_1, \dots, T_i, \dots)$  such that multiplying  $T$  by  $\oplus M_{v_i}$  is the same as calculating  $(T_1 M_{v_1}, \dots, T_i M_{v_i}, \dots)$ . Eventually,  $T$  becomes a  $1 \times 1$  matrix and the algorithm then returns its sole entry.

Inside of the main **for** loop of the function **tutte**, there are three main contributors to the time complexity: the function **Distance**, finding the matrix associated to a vertex, and the matrix multiplication.

If we let  $P[i]$  be the  $i^{th}$  bit of  $P$  and  $Q[i]$  likewise. Then **Distance** finds  $\sum_{k \leq i} Q[k] - P[k]$ . This runs in time  $O(n)$ .

For the function  $M(v)$ , there are a fixed finite of number of cases that need to be checked depending on which entering and exiting edges it has and whether or not  $v$  lies on  $P$  or  $Q$ . This runs in time  $O(1)$ .

For any stack, there are at most  $n$  vertices in the stack, and each  $M_{v_i}$  has at most two rows and at most two columns. If we specify values for  $x$  and  $y$ , calculating  $(T_1 M_{v_1}, \dots, T_i M_{v_i}, \dots)$  takes at most  $16n$  computations. Thus it is  $O(n)$ .

Both the function **distance** and updating  $T$  runs in time  $O(n)$  inside the main loop. Since there are  $n$  iterations of the main **for** loop, the overall time complexity is  $O(n^2)$ .

Should we wish to compute the polynomial itself rather than its value, we need to account for multiplying intermediate polynomials by 1,  $x$ , or  $y$ , and adding them. The polynomials in the intermediate vector  $T$  after stack  $i$  are of degree at most  $i$  so have at most  $\binom{i+2}{i}$  terms. The last application of an  $S_i$  can involve adding quadratically large polynomials and so the additional cost is at most  $O(n^2)$ .

Then updating the vector can be done in time  $O(n^3)$  and this lies in a **for** loop that iterates at most  $n$  times. So the overall algorithm is  $O(n^4)$ . This is indeed an improvement over [13]

## Chapter 6

# An Introduction to Invariant Theory

In Chapter 4, we noted the difficulty of determining whether or not a tensor network is a Pfaffian or determinantal circuit. This motivates our attempt to study the invariant theory of the action of  $\mathrm{GL}_T$  on a tensor network  $T$ , viewed as an invariant polynomial. We present our results in subsequent chapters. This chapter serves to introduce many of the classical results from invariant theory that we use.

Let  $G \curvearrowright V$  be a group acting on a vector space or, more generally, a variety. For our purposes,  $V$  will always be an affine variety over a field  $k$  of characteristic 0, often algebraically closed. Many of the motivating examples come from working over the field  $\mathbb{C}$ . Throughout this dissertation, we always choose an basis for any vector space under consideration, although we may change this basis when it proves convenient.

Choosing a basis for  $V$ , if it is  $n$ -dimensional then we have the isomorphism  $SV^* \cong k[x_1, \dots, x_n]$ . Under a choice of basis, we have the coordinate ring of  $V$ , denoted  $k[V]$ , is isomorphic to  $k[x_1, \dots, x_n]$ . Since we always choose a basis, we simply define  $k[V] := k[x_1, \dots, x_n]$ .

There is an induced action by  $G$  on  $k[V]$ : For  $f \in k[V]$ ,  $g.f := f(g^{-1})$ , for  $g \in G$ . One can then form the ring of polynomials that are fixed by this action:

$$k[V]^G := \{f \in k[V] \mid g.f = f, \forall g \in G\}.$$

Note that this ring consists of the polynomials that are constant on orbits. The basic idea is to try and distinguish two orbits from one another by testing to see if they agree on all invariant polynomials.

While it is always the case that two orbits are distinct if they take a

different value on some invariant, the converse is not generally true. To understand when two orbits can be distinguished by invariants, how orbits relate to the Zariski topology must be considered.

Defining a polynomial on a Zariski dense set means that the polynomial can be uniquely lifted to one on the Zariski closure. So if an invariant is constant on an orbit, it is constant on the Zariski closure of that orbit. If the orbit closures of two orbits intersect, any invariant must take the same value on both orbits. Thus these two orbits cannot be distinguished by polynomials. This defines an equivalence class of orbits: two orbits are equivalent if their closures intersect. The study of orbit closures leads into algebraic geometry. We call such an equivalence class of orbits an *orbit class*.

**Example 6.1.** Let  $V = k^2$ , so  $k[V] \cong k[x, y]$ . Then consider the action of  $k^\times$  on  $V$  given by  $t.(x, y) = (tx, t^{-1}y)$ . Then it is easy to see that  $k[V]^{k^\times}$  is equal to the ring  $k[xy]$ . In this example, determining the orbits of this action is straightforward. If we set  $xy = c$  for  $c \neq 0$ , this specifies a unique orbit. Furthermore, each such orbit is closed as it is defined by an irreducible polynomial. However, if  $xy = 0$ , there are three possible orbits. These are the origin, the  $x$ -axis minus the origin, and the  $y$ -axis minus the origin. Since the invariant ring cannot distinguish between these three orbits, we claim that the closure of any two intersect. Indeed, it is easily seen that this is the case. What we see then is that each equivalence class of orbits is specified by the value of the single invariant polynomial. So the equivalent classes of orbits can be associated with  $\text{Spec} k[V]^G \cong k$ . We will see that something similar holds in general in Section 6.2.

The groups we consider are typically products of general or special linear groups which are algebraic groups. Recall that an *algebraic group* is a group that is also a variety such that multiplication and inversion are regular functions on the variety.

A rational representation of an algebraic group  $\phi : G \rightarrow \text{GL}(V)$  is one where  $\phi$  is a rational map of varieties. If  $G = \text{GL}(W)$ , then this means every entry coordinate is a polynomial in the entries of  $\text{GL}(W)$  divided by some power of the determinant.

**Definition 6.2.** A representation  $G \curvearrowright V$  is called *simple* if there is no non-trivial submodule  $W \subsetneq V$  such that  $G.V \subseteq W$ . It is called *semisimple* if it is the direct sum of simple modules.

**Definition 6.3.** An algebraic group over an algebraically closed field of characteristic 0 is called *reductive* if every rational representation  $\phi : G \rightarrow \text{GL}(V)$  is semisimple.

All groups we consider will be reductive and all representations will be rational. For a reductive group  $G \curvearrowright V$ , the invariant ring  $k[V]^G$  has many nice properties. The most important fact about such rings is Hilbert's celebrated Basissatz.

**Theorem 6.4** ([47, 46]). *If  $W$  is a  $G$ -module and the induced action on  $k[W]$  is completely reducible, the invariant ring  $k[V]^G$  is finitely generated.*

We note that for rational representations of reductive groups, the action on  $k[W]$  will always be completely reducible. As a consequence, for rational representations of reductive groups, the resulting invariant ring is always finitely generated. This is incredibly useful because then one needs only find a finite generating set for the invariant ring. Two orbits can be separated by orbits if and only if there is a generator that separates them. Actually, one does not necessarily need the generators of the invariant ring. A set of invariants is called a separating set if they induce the same equivalence class of orbits as the full invariant ring [57].

There are other useful properties of invariant rings of rational representations of reductive groups. Of note is the fact that the invariant ring is Cohen-Macaulay [49]. This is useful when trying to establish a degree bound on generators of the invariant ring, see for example [87, 88].

## 6.1 Centralizers of Semisimple Algebras

In this section, we outline the basic theory of centralizers of finite semisimple algebras. The use of centralizers has been fundamental in the computation of many of the invariant rings of classical actions. We first recall the definition of a semisimple algebra. Recall that if  $M$  is an  $A$ -module,  $M$  is called simple if it contains no non-zero proper  $A$ -submodule.

**Definition 6.5.** A finite dimensional algebra  $A$  is called semisimple if, when viewed as an  $A$ -module acting on itself by left (or right) multiplication, it decomposes as a direct sum of simple  $A$ -modules. Equivalently, every  $A$ -module is semisimple.

Semisimple algebras are well understood. For finite dimensional semisimple algebras over a field  $k$ , the Artin-Wedderburn Theorem implies that every such algebra is a product of matrix algebras over finite dimensional division algebras [3, 105]. More precisely, if  $A$  is such an algebra, then

$$A \cong \prod_{i=1}^n \text{Mat}_{n_i \times n_i}(D_i)$$

where each  $D_i$  is a division algebra over  $k$ .

We have been discussing actions by groups, that is, representations  $\phi : G \rightarrow \text{GL}(V)$  for some vector space  $V$ . Note that this lifts uniquely to a representation of algebras  $\phi : k[G] \rightarrow \text{End}(V)$ . Let  $\langle G \rangle_\phi$  denote the image of  $k[G]$  under the map  $\phi$ .

The algebra  $k[G]$  is called the *group algebra* of  $G$ . It is the algebra generated by the  $k$ -span of the elements  $\{v_g \mid g \in G\}$  and where multiplication is defined by  $v_g \cdot v_h := v_{gh}$ . In many classical cases, the image of a representation of  $k[G]$  is a finite dimensional semisimple algebra.

**Definition 6.6.** Given an algebra representation  $\varphi : A \rightarrow \text{End}(V)$ , we define the *centralizer* of  $A$  with respect to  $\varphi$  as

$$\text{End}_A(V) := \{a \in \text{End}(V) \mid ab = ba \ \forall b \in \varphi(A)\}.$$

Studying the algebra  $\text{End}_A(V)$  can be used to construct invariant rings for adjoint actions of various classical groups. However, it is important that these algebras be well behaved. The following famous theorem is a key ingredient to much of classical invariant theory.

**Theorem 6.7** (Double Centralizer Theorem [60]). *Over any field  $k$ , let  $A \subseteq \text{End}(V)$  be a finite dimensional semisimple algebra and  $A'$  be its centralizer. Then*

- (a)  $A'$  is a finite dimensional semisimple algebra and  $(A')' = A$ .
- (b)  $V$  has a unique decomposition  $V_1 = \bigoplus_{i=1}^n V_i$  into simple, non-isomorphic  $A \otimes A'$ -modules.
- (c) Each  $V_i = W_i \otimes_{D_i} Y_i$  where  $W_i$  is a simple  $A$ -module,  $Y_i$  a simple  $A'$ -module, and  $D_i$  a finite dimensional division algebra over  $k$ .

Consider the representation of  $\mathcal{S}_n$  on  $V^{\otimes n}$  by permutation on the factors  $V_i$ . Over a field of characteristic 0, Maschke's theorem says that the image of  $k[\mathcal{S}_n]$  in  $\text{End}(V^{\otimes n})$  is a semisimple algebra [76, 77].

There is also an action of  $\text{GL}(V)$  on  $V^{\otimes n}$  by  $g \cdot (\bigotimes_{i=1}^n v_i) = \bigotimes_{i=1}^n g v_i$ . It is a classical fact that  $\text{End}_{\mathcal{S}_n}(V^{\otimes n}) = \text{End}(V)$  and likewise  $\text{End}_{\text{GL}(V)}(V^{\otimes n})$  is the image of  $k[\mathcal{S}_n]$  under the discussed representation. The second and third parts of Theorem 6.7 then lies at the heart of the celebrated Schur-Weyl duality for the symmetric group.

We are interested in products of groups, so we wish to prove that centralizers behave well for products for the representations we consider. We

are interested in a groups of the form  $G_{\mathbf{d}} := \times_{i=1}^n G_{d_i}$  acting on the space  $V^{\otimes m}$  where  $V = \bigotimes_{i=1}^n V_i$  by a representation  $\phi = \bigotimes_{i=1}^n \phi_i$ . Viewing  $V^{\otimes m} \cong V_1^{\otimes m} \otimes \cdots \otimes V_n^{\otimes m}$ , this action is given by

$$(g_1, \dots, g_n) \cdot \left( \left( \bigotimes_{j=1}^m v_{1j} \right) \otimes \cdots \otimes \left( \bigotimes_{j=1}^m v_{nj} \right) \right) = \left( \bigotimes_{j=1}^m \phi_1(g_1) v_{1j} \right) \otimes \cdots \otimes \left( \bigotimes_{j=1}^m \phi_n(g_n) v_{nj} \right)$$

extended linearly.

**Theorem 6.8.** *Let  $G_{\mathbf{d}} = \times_{i=1}^n G_{d_i}$  act on  $V^{\otimes m} \cong V_1^{\otimes m} \otimes \cdots \otimes V_n^{\otimes m}$  over a field of characteristic 0 via a representation  $\phi$ . Assume each  $\langle G_{d_i} \rangle_{\phi_i}$  is a semisimple algebra, where  $\phi_i$  is the restricted representation  $G_{d_i} \curvearrowright (V_i)^{\otimes m}$ . If  $\text{End}_{G_{d_i}}(V_i^{\otimes m}) = A_i$  then*

- (a)  $\text{End}_{\otimes A_i}(V^{\otimes m}) = \langle G_{\mathbf{d}} \rangle_{\phi}$ .
- (b)  $\text{End}_{G_{\mathbf{d}}}(V^{\otimes m}) = \bigotimes_{i=1}^n A_i$ .

*Proof.* To prove part (a), we know that  $V^{\otimes m} \cong V_1^{\otimes m} \otimes \cdots \otimes V_n^{\otimes m}$  and we have an action of  $\times_{i=1}^n A_i$  by

$$(a_1, \dots, a_n) \cdot \left( \left( \bigotimes_{j=1}^m v_{1j} \right) \otimes \cdots \otimes \left( \bigotimes_{j=1}^m v_{nj} \right) \right) = a_1 \cdot \left( \bigotimes_{j=1}^m v_{1j} \right) \otimes \cdots \otimes a_n \cdot \left( \bigotimes_{j=1}^m v_{nj} \right)$$

extended linearly. We see that the span of the image of the action of  $\times_{i=1}^n A_i$  is equal to  $\bigotimes_{i=1}^n A_i$ . Let  $\varphi : \text{End}(V)^{\otimes m} \cong \text{End}(V^{\otimes m})$  be the isomorphism given by

$$\varphi \left( \bigotimes_{i,j} M_{ij} \right) \left( \bigotimes_{i,j} v_{ij} \right) = \bigotimes_{i,j} M_{ij} v_{ij}.$$

Since the image of  $\times_{i=1}^n A_i$  spans  $\bigotimes_{i=1}^n A_i$ , it is sufficient to see that for  $\alpha = (a_1, \dots, a_n) \in \times_{i=1}^n A_i$ ,

$$\alpha \cdot \varphi \left( \bigotimes_{i,j} M_{ij} \right) \left( \bigotimes_{i,j} v_{ij} \right) = \varphi \left( \bigotimes_{i,j} a_i M_{ij} \right) \left( \bigotimes_{i,j} v_{ij} \right) \quad \text{and}$$

$$\varphi \left( \bigotimes_{i,j} M_{ij} \right) (\alpha \cdot \bigotimes_{i,j} v_{ij}) = \varphi \left( \bigotimes_{i,j} M_{ij} \right) \left( \bigotimes_{i,j} a_i v_{ij} \right) = \varphi \left( \bigotimes_{i,j} M_{ij} a_i \right) \left( \bigotimes_{i,j} v_{ij} \right).$$

So  $\varphi \left( \bigotimes_{i,j} M_{ij} \right) \in \text{End}(V^{\otimes m})$  commutes with  $\times_{i=1}^n A_i$  precisely when we have that each  $\varphi_i \left( \bigotimes_j M_{ij} \right) \in \text{End}_{A_i}(V_i^{\otimes m})$  where  $\varphi_i : \text{End}(V_i)^{\otimes m} \cong \text{End}(V_i^{\otimes m})$  is the canonical isomorphism. Thus we have that  $\text{End}_{\otimes A_i}(V^{\otimes m})$  is generated by those tensors  $w_1 \otimes \cdots \otimes w_n$  where each  $w_i \in \langle G_{d_i} \rangle_{\phi_i}$ . So rewriting gives us that

$$w = \bigotimes_{i=1}^n \left( \sum_j \beta_{ij} x_{ij} \right) = \sum_{j_1, \dots, j_n} \beta_{1j_1} x_{1j_1} \otimes \cdots \otimes \beta_{nj_n} x_{nj_n}$$

where  $x_{ij} \in \langle G_{d_i} \rangle_{\phi_i}$ . So we see that  $w \in \langle G_{\mathbf{d}} \rangle_{\phi}$ . This proves the first part. Part (b) follows from the Double Centralizer Theorem (Theorem 6.7).  $\square$

## 6.2 The Categorical Quotient

Even if we can determine invariant rings for the actions of interest to us, we still want to investigate how well polynomials separate orbits. This leads us to study the orbit spaces using algebraic geometry. We saw in Example 6.1 that the orbit classes were parameterized by  $k \cong \text{Spec } k[V]^{k^\times}$ . In fact, for any  $G \curvearrowright k[V]$ , the orbit classes form a variety isomorphic to  $\text{Spec } k[V]^G$ . We have already seen that all invariants are constant on orbit classes. It turns out that distinct orbit classes can always be distinguished.

**Theorem 6.9** ([80]). *For two distinct orbit classes, there is an invariant that takes different values on each class.*

As a consequence, every orbit class is specified by a choice of value for each generator of the invariant ring. Thus we prove our assertion that the orbit classes are parameterized by the points of  $\text{Spec } k[V]^G$ . This variety is called the *categorical quotient* of the action  $G \curvearrowright V$  and is often denoted  $V//G$ . This comes with a canonical map  $\pi : V \rightarrow V//G$ .

The fibers of the map  $\pi$  also contain a lot of structure. We give the following structure theorem for orbit classes.

**Theorem 6.10** ([15, 80]). *Given an action of an algebraic group  $G \curvearrowright V$ , the orbit closure  $\overline{G.x}$  is the union of  $G.x$  and orbits of strictly smaller dimension. An orbit of minimal dimension is closed, thus every closure  $\overline{G.x}$  contains a closed orbit. Furthermore, this closed orbit is unique.*

So we have that every orbit class has a unique representative given by closed orbit and every closed orbit trivially lies in some orbit class. So the closed orbits are also parameterized by  $V//G$ . This motivates the definition of different types of points in  $V$  with respect to an action of  $G$ .

**Definition 6.11.** Given an action  $G \curvearrowright V$  and a point  $v \in V \setminus \{0\}$ , then  $v$  is called

- (a) an *unstable point* if  $0 \in \overline{G.v}$ ,
- (b) a *semistable point* if  $0 \notin \overline{G.v}$ ,
- (c) a *polystable point* if  $G.v$  is closed,



(d) or a *stable point* if  $G.v$  is closed and the stabilizer of  $v$  is finite.

A powerful tool in studying orbit closures lies in looking at limits of 1-parameter subgroups, also called cocharacters. Given a representation  $\varphi : G \rightarrow \mathrm{GL}(V)$ , a *cocharacter* of  $G$  is a representation  $\lambda : k^\times \rightarrow \mathrm{GL}(V)$  such that there exists a map  $\psi : k^\times \rightarrow G$  such that the following diagram commutes.

$$\begin{array}{ccc} k^\times & \xrightarrow{\psi} & G \\ & \searrow \lambda & \downarrow \varphi \\ & & \mathrm{GL}(V). \end{array}$$

Given a group homomorphism  $\psi : k^\times \rightarrow G$ , this induces a cocharacter by the representation  $\lambda = \varphi \circ \psi : k^\times \rightarrow \mathrm{GL}(V)$ . Often we simply specify the map  $\psi$  and then write  $\psi(t).v := \varphi(\psi(t))v$ .

**Theorem 6.12** (The Hilbert-Mumford Criterion [58]). *For a linearly reductive group  $G$  acting on a variety  $V$ , if  $v \in \overline{G.w} \setminus G.w$  then there exists a cocharacter  $\lambda(t) : k^\times \rightarrow G$ , such that  $\lim_{t \rightarrow 0} \lambda(t).w = v$ .*

From the representation theory of  $k^\times$ , we know that since this group is semi-simple, any rational representation of  $k^\times \curvearrowright V$  decomposes into a direct sum of 1-dimensional subspaces on which it acts by multiplication by  $t^\alpha$ , for some  $\alpha \in \mathbb{Z}$ . Thus every cocharacter is diagonalizable where the diagonal entries are of the form  $t^{\alpha_i}$ ,  $\alpha_i \in \mathbb{Z}$  (cf. [60]). If the group  $G$  is sufficiently large, the change of basis necessary to diagonalize any cocharacters is an element of the group. This is useful since one can then focus one's attention to studying limits of diagonal cocharacters.

**Definition 6.13.** The *null cone* of an action  $G \curvearrowright V$  is the set of unstable points and the origin. We denote it by  $\mathcal{N}_V$ . Equivalently,  $\mathcal{N}_V$  are those  $v \in V$  such that  $f(v) = f(0)$  for all invariant polynomials  $f$ .

If one wants to compute the generators of an invariant ring, this can be accomplished by determining the null cone. This is done by seeing which elements of  $v \in V$  have the property that there is a cocharacter that takes  $v \rightarrow 0$ . If the cocharacters can be diagonalized by  $G$ , this makes the task even easier.

**Proposition 6.14** ([46, 27]). *For  $G \curvearrowright V$ , the ideal  $I(\mathcal{N}_V)$  is generated by invariants  $f_1, \dots, f_s$  and the invariant ring is  $k[V]^G = k[f_1, \dots, f_s]$ .*

### 6.3 An Extended Example: The Adjoint Action of $\mathrm{GL}(V)$ on $\mathrm{End}(V)$

We now work out in detail the invariant ring and categorical quotient of  $\mathrm{GL}(V)$  acting on  $\mathrm{End}(V)$  by conjugation. Of course, it is well known that two matrices are in the same conjugacy class if and only if they have the same Jordan normal form. We rephrase this fact in the context of invariant theory using the methods outlined in the previous section.

We first seek to understand the null cone of this action,  $\mathcal{N}_{\mathrm{End}(V)}$ . From there we will determine generators of this ideal that are also invariant under conjugation. This will give us the generators of the invariant ring by Proposition 6.14.

Let us consider a cocharacter of  $\mathrm{GL}(V)$ ,  $\lambda(t)$  written in some basis. Apart from conjugation,  $\lambda(t)$  also acts naturally on  $V$  by left multiplication. Therefore, we know that under some change of basis,  $\lambda(t)$  is diagonal with entries  $t^{\alpha_i}$ ,  $\alpha_i \in \mathbb{Z}$ . That is to say, there is some  $g \in \mathrm{GL}(V)$  such that  $g\lambda(t)g^{-1}$  is diagonal.

Now suppose we are considering the limit  $\lim_{t \rightarrow 0} (\lambda(t)M\lambda(t)^{-1})$  for some  $M \in \mathrm{End}(V)$ , and suppose this limit exists. Then the following limit exists:

$$\lim_{t \rightarrow 0} g\lambda(t)g^{-1}gMg^{-1}g\lambda(t)^{-1}g^{-1}.$$

Now if we choose  $g$  such that  $\mu(t) = g\lambda(t)g^{-1}$  is diagonal, and we denote  $N = gMg^{-1}$ , we have that

$$\lim_{t \rightarrow 0} \mu(t)N\mu(t)^{-1}$$

exists. Furthermore,  $N$  is in the orbit of  $M$ . So we study those matrices that can be taken to the origin by a diagonal cocharacter. We know that every element of the null cone has such a matrix in its orbit.

Let us now focus our attention on the diagonal cocharacters of  $\mathrm{GL}(V)$  in some given basis. Let  $\lambda(t)$  be such a cocharacter and  $M \in \mathrm{End}(V)$ . Let  $\lambda(t)_{ii} = t^{\alpha_i}$  and  $M = \{m_{ij}\}$  is an  $n \times n$  matrix. Then  $\lambda(t)M\lambda(t)^{-1}$  (which we henceforth abbreviate as  $\lambda(t).M$ ) is equal to  $\{t^{\alpha_i - \alpha_j}m_{ij}\}$ .

We want to know under what conditions does  $\lim_{t \rightarrow 0} \lambda(t).M = 0$ . We have now moved to the action of a maximal torus inside of  $\mathrm{GL}(V)$  on  $\mathrm{End}(V)$ . This action also has invariants, which are much easier to see. They are of the form

$$C_{i_1, \dots, i_k}(M) := \prod_{j=1}^k m_{j i_j + 1 \pmod{k}}.$$

Indeed, let us look at the value of  $C_{i_1, \dots, i_k}(\lambda(t).M)$ :

$$\begin{aligned} C_{i_1, \dots, i_k}(\lambda(t).M) &= \prod_{i=1}^k t^{\alpha_{j_i} - \alpha_{j_{i+1} \pmod k}} m_{j_i j_{i+1} \pmod k} = \\ t^{\sum_{i=1}^k \alpha_{j_i} - \alpha_{j_{i+1} \pmod k}} \cdot \prod_{i=1}^k m_{j_i j_{i+1} \pmod k} &= \prod_{i=1}^k m_{j_i j_{i+1} \pmod k}. \end{aligned}$$

Thinking of  $M$  as an adjacency matrix, where the edges have weights, these polynomials correspond to the products of weights on a cycle. It is not difficult to see that these generate all of the invariants of this action.

Clearly, if  $M = 0$ , then all of the polynomials  $C_{i_1, \dots, i_k}(M) = 0$ . The null cone of this new actions is therefore all matrices  $M$  such that  $C_{i_1, \dots, i_k}(M) = 0$ . But this means that if we view  $M$  as a weighted adjacency matrix, the graph is acyclic. From there it is not hard to see that  $M$  is permutation conjugate to a strictly upper triangular matrix. In particular  $M$  is nilpotent.

In fact, every nilpotent matrix is conjugate to a strictly upper triangular matrix. Therefore, the null cone is precisely the nilpotent matrices. Furthermore, we know that a matrix  $M$  is nilpotent if and only if  $\text{Tr}(M^k)$  vanish for  $1 \leq k \leq n$ . The polynomials  $\text{Tr}(M^k)$  are the power sum symmetric polynomials in the eigenvalues:  $\lambda_1^k + \dots + \lambda_n^k$ . It is well known that these polynomials are algebraically independent.

Thus we have that  $k[\text{End}(V)]^{\text{GL}(V)} = k[\text{Tr}(M^k) \mid 1 \leq k \leq \dim(V)]$ . Setting each generator to a specific value uniquely determines a choice of spectrum. Since there are no restrictions on what may constitute the spectrum of a matrix, this is another proof that the generators are algebraically independent. Thus we have that categorical quotient is  $\text{End}(V) // \text{GL}(V) = \text{Spec } k[\text{End}(V)]^{\text{GL}(V)} \cong k^{\dim(V)}$ . Furthermore, each orbit class is parameterized by a choice of spectrum.

In this example, two matrices are distinguishable by invariants if only if they have the same spectrum. If the matrix is diagonalizable, then it is well known that the corresponding orbit is closed (cf. [14]). So diagonalizable matrices are polystable. We will prove a generalization of this fact in Chapter 8.

The Jordan decomposition tells us that every matrix can be written as the sum of a diagonalizable matrix and a nilpotent matrix. So we see that the unique closed orbit contained in the orbit closure of a matrix is the orbit of the diagonalizable matrix in this decomposition. This tells us a lot about the geometry of the orbit space and motivates the following definition.

**Definition 6.15.** Given an action  $G \curvearrowright V$ , a *Jordan decomposition* of a point  $v$  is given by  $v = v_s + v_n$  where  $v_s$  is a polystable point and  $v_n$  is an unstable point.

For a rational representation of a reductive group  $G \curvearrowright V$ , such a Jordan decomposition always exists. This is well known (cf. [66]), but we include a proof for completeness.

**Theorem 6.16.** *For a reductive group action  $\varphi : G \rightarrow \mathrm{GL}(V)$  a Jordan decomposition always exists.*

*Proof.* By Theorem 6.10,  $\overline{\varphi(G)v}$  contains a polystable point  $v_s$ , and by the Hilbert-Mumford criterion (Theorem 6.12), there exists a cocharacter  $\lambda(t) : k^\times \rightarrow G$  such that  $\lim_{t \rightarrow 0} \varphi(\lambda(t))v$  is polystable. Since  $\varphi(\lambda(t))$  is diagonalizable, there is some  $g \in \mathrm{GL}(V)$  such that  $\lim_{t \rightarrow 0} g\varphi(\lambda(t))g^{-1}gv = gv_s$  for some  $v_s \in V$ .

Now if  $g\varphi(\lambda(t))g^{-1}$  is diagonal, then  $g\varphi(\lambda(t))v$  is the vector  $gv$  with every entry multiplied by a some non-negative power of  $t$  (since the limit exists). The unstable part of  $gv$ ,  $gv_n$ , is the all zero vector except for those entries of  $gv$  that get multiplied by a positive power of  $t$ . The stable part is  $gv_s = gv - gv_n$ . Then we see that  $\lim_{t \rightarrow 0} g\varphi(\lambda(t))g^{-1}gv_s = gv_s$  and so  $\lim_{t \rightarrow 0} \varphi(\lambda(t))v_s = v_s$ . Then we let  $v_n = v - v_s$ . We quickly see that  $\lim_{t \rightarrow 0} \varphi(\lambda(t))v = v_s$  and thus  $\lim_{t \rightarrow 0} \varphi(\lambda(t))v_n = 0$ . Then  $v = v_s + v_n$  is the Jordan decomposition.  $\square$

## Chapter 7

# The Invariants of Local Conjugation

We now turn directly to the problem of studying the invariant ring of  $\mathrm{GL}_T$  acting on a tensor network  $T$ . This problem in full generality is very difficult, so we make some strong simplifying assumptions. We first restrict ourselves to tensor networks where all boxes contain morphisms in the same space:  $\mathrm{End}(V)$ , where  $V = \bigotimes_{i=1}^n V_i$  over an algebraically closed field of characteristic 0,  $k$ .

The graphs underlying such tensor networks will be regular. We also restrict the action to a subgroup of  $\mathrm{GL}_T$  for such regular tensor networks. We define a *local group*  $G_{\mathbf{d}} = \times_{i=1}^n G_{d_i}$ , where  $G_{d_i}$  is a subgroup of  $\mathrm{GL}(V_i)$ , acting on  $\mathrm{End}(V)$  by

$$\times_{i=1}^n g_i.M := \left( \bigotimes_{i=1}^n g_i \right) M \left( \bigotimes_{i=1}^n g_i^{-1} \right). \quad (7.1)$$

This naturally extends to an action on  $\mathrm{End}(V)^{\oplus m}$  by simultaneous conjugation. If there are  $m$  boxes in a regular tensor network  $T$ , we restrict our action to the group  $\mathrm{GL}_{\mathbf{d}} := \times_{i=1}^n \mathrm{GL}(V_i)$  on  $\mathrm{End}(V)^{\oplus m}$ , which is a subgroup of  $\mathrm{GL}_T$ . In this chapter, we compute the invariant ring for certain local groups, which includes  $\mathrm{GL}_{\mathbf{d}}$ . The case  $n = 1$  was solved by Procesi [89].

This problem is also important for understanding entanglement of quantum states [9, 38, 39, 45, 51, 61, 71, 72, 74]. For  $V = \bigotimes_{i=1}^n V_i$ , physicists look at trace one, positive semi-definite matrices in  $\mathrm{End}(V)$ . These operators, called *density operators*, represent a superposition of quantum states, where each is a state of  $n$  particles with the  $i^{\text{th}}$  particle having  $\dim(V_i)$  spins.

Many of the most important notions of entanglement are invariant under the action of  $U_{\mathbf{d}} := \times_{i=1}^n U(\mathbb{C}^{d_i})$  [31]. Entanglement in turn relates to quantum computation [82, 90], quantum error correction [82], and quantum simulation [70]. Many of the results we prove in this and subsequent chapters are with this application in mind.

For the study of actions of  $U_{\mathbf{d}}$ , we do not have access to many of the theorems from Chapter 6 as it is not a reductive group. However, the following two propositions tell us that studying  $GL_{\mathbf{d}}$  is sufficient.

**Proposition 7.1.** *If  $H$  is a Zariski dense subgroup of  $G$  and  $\rho$  is a rational representation of  $G$  acting on a vector space  $V$ ,  $k[V]^G = k[V]^H$ .*

*Proof.* The representation  $\rho$  is a continuous map from  $G \rightarrow GL(V)$  with respect to the Zariski topology by assumption of the rationality of the representation. For every  $v \in V$ , consider the map  $\varphi_v : G \rightarrow G.v$  given by  $g \mapsto g.v$ . This is also a continuous map and it implies that for every  $v \in V$ ,  $H.v$  is dense in  $G.v$  since the continuous image of dense sets are dense. The invariant ring is the ring of polynomials which are constant on orbit closures. Since the orbit closures of  $H$  and  $G$  coincide, their invariant rings must be the same.  $\square$

It is well known that  $U(\mathbb{C}^{d_i})$  is the maximal compact subgroup of  $GL(\mathbb{C}^{d_i})$  and as such is a Zariski dense subgroup. This implies that  $U_{\mathbf{d}}$  is Zariski dense in  $GL_{\mathbf{d}}$ , so  $\mathbb{C}[\text{End}(V)^{\oplus m}]^{U_{\mathbf{d}}} = \mathbb{C}[\text{End}(V)^{\oplus m}]^{GL_{\mathbf{d}}}$ . Furthermore, the action  $GL_{\mathbf{d}} \curvearrowright \text{End}(V)^{\oplus m}$  is not faithful and has the same orbits as the action of  $SL_{\mathbf{d}} \curvearrowright \text{End}(V)^{\oplus m}$ . Therefore, we have that  $\mathbb{C}[\text{End}(V)^{\oplus m}]^{SU_{\mathbf{d}}} = \mathbb{C}[\text{End}(V)^{\oplus m}]^{SL_{\mathbf{d}}} = \mathbb{C}[\text{End}(V)^{\oplus m}]^{GL_{\mathbf{d}}}$ .

**Proposition 7.2.** *Two Hermitian matrices are in the same  $GL_{\mathbf{d}}$  orbit if and only if they are in the same  $U_{\mathbf{d}}$  orbit.*

*Proof.* Consider the polar decomposition of  $\otimes_{i=1}^n g_i = (\otimes_{i=1}^n p_i)(\otimes_{i=1}^n u_i)$  where the  $p_i$  are invertible Hermitian matrices and the  $u_i$  are unitary. We can assume without loss of generality that all  $u_i = \text{id}$  since it does not change the  $U_{\mathbf{d}}$  orbit we are in. So note that  $P = \otimes_{i=1}^n p_i$  is a Hermitian matrix. Let  $H$  be Hermitian and suppose that  $PHP^{-1}$  is Hermitian. Then  $PHP^{-1} = (PHP^{-1})^{\dagger} = P^{-1}HP$ , implying that  $P^2HP^{-2} = H$ . This implies that either  $P$  commutes with  $H$ , and thus  $PHP^{-1}$  is in the same  $U_{\mathbf{d}}$  orbit as  $H$ , or  $P^2 = PP^{\dagger} = \text{id}$ , implying that  $P$  was unitary.  $\square$

We first determine the homogeneous invariants of  $GL_{\mathbf{d}}$ . We know from Theorem 6.4 that the invariants ring is finitely generated. We compute an

upper bound for  $\beta_{\text{GL}(V)}(\text{End}(V)^{\oplus m})$ , defined as follows. Given a representation of a group  $G$  on a vector space  $V$  over a field  $k$ ,

$$\beta_G(V) := \min\{d \mid k[V]^G \text{ is generated by polynomials of degree } \leq d\}.$$

For arbitrary semisimple groups  $G$ , general upper bounds have been given on  $\beta_G(V)$  [27, 87, 88, 48]. For a ring  $R$ , we define

$$\beta(R) := \min\{d \mid R \text{ is generated by polynomials of degree } \leq d\}.$$

## 7.1 Multilinear Invariants of Local Groups

We first describe the invariants for a product of groups  $G_{\mathbf{d}} = \times_{i=1}^n G_{d_i}$  acting by conjugation on  $\text{End}(V)$ . Let  $A_i^k$  be the centralizer of  $\langle G_{d_i} \rangle$  acting on  $V^{\otimes k}$ . Here we do not make any assumptions about our base field.

We describe a surjection of  $\bigoplus_{k=1}^{\infty} \otimes_{i=1}^n A_i^k$  onto the multilinear invariants of  $k[\text{End}(V)^{\oplus m}]^{G_{\mathbf{d}}}$  where the module  $\otimes_{i=1}^n A_i^k$  maps to the  $i^{\text{th}}$  graded piece of  $k[\text{End}(V)^{\oplus m}]^{G_{\mathbf{d}}}$  under the natural grading. We follow Kraft and Procesi's ([60]) treatment of the Fundamental Theorems, generalizing to conjugation by  $G_{\mathbf{d}}$ ; see also Leron [67].

**Definition 7.3.** We define the following multilinear polynomials on rank one tensors of  $V^{\otimes m}$  and extend linearly. For every  $\alpha_i \in A_i^m$ , there is a corresponding multilinear invariant of degree  $m$ ,  $T_{\alpha_i}^m \in k[\text{End}(V)^{\oplus m}]^{G_{d_i}}$  (we prove this in Theorem 7.4 although it is classical). We define  $T_{\alpha}^m := \prod_{i=1}^n T_{\alpha_i}^m$  which acts on rank one tensors by

$$T_{\alpha}^m \left( \bigotimes_{j=1}^n M_{j1}, \dots, \bigotimes_{j=1}^n M_{jm} \right) := \prod_{i=1}^n T_{\alpha_i}^m(M_{i1}, \dots, M_{im}).$$

**Theorem 7.4.** *The multilinear invariants of  $\text{End}(V)^{\oplus m}$  under the adjoint action of  $G_{\mathbf{d}}$  are generated by the  $T_{\alpha}^m$ .*

*Proof.* Let  $M$  be the multilinear functions from  $\text{End}(V)^{\oplus m} \cong (V \otimes V^*)^{\oplus m} \rightarrow k$ . Then we can identify  $M$  with  $(V \otimes V^*)^{*\otimes m}$  by the universal property of tensor product. But we have an isomorphism

$$\varphi : \text{End}(V)^{\otimes m} \rightarrow (V \otimes V^*)^{*\otimes m}$$

given by  $\varphi(M)(v \otimes \eta) = \eta(Mv)$  which is  $GL(V)$ -equivariant. So we get a  $G_{\mathbf{d}}$ -equivariant isomorphism  $\text{End}(V)^{\otimes m} \cong M$ . This induces an isomorphism

$$\text{End}_{G_{\mathbf{d}}}(V^{\otimes m}) \cong M^{G_{\mathbf{d}}}$$

where  $M^{\mathbf{G_d}}$  are the  $\mathbf{G_d}$ -invariant multilinear functions.

Since  $V^{\otimes m} \cong V_1^{\otimes m} \otimes \cdots \otimes V_n^{\otimes m}$ , we can write  $\varphi = \bigotimes \varphi_i$  where  $\varphi_i$  are the induced isomorphisms  $\text{End}(V_i)^{\otimes m} \rightarrow (V_i \otimes V_i^*)^{*\otimes m}$ .

Since  $\text{End}_{\mathbf{G_d}}(V^{\otimes m}) \cong M^{\mathbf{G_d}}$ , by Theorem 6.8, the image of each  $\alpha \in \times_{i=1}^n A_i^m$  under the isomorphism  $\varphi$  gives the generators of  $M^{\mathbf{G_d}}$ . In the case of  $n = 1$ , this is classical and gives the multilinear invariants  $M^{G_{d_1}}$ . This is the map that associates to every  $\alpha_i$  a multilinear invariant in  $k[\text{End}(V)^{\oplus m}]^{G_{d_i}}$ .

So consider  $\alpha = (\alpha_1, \dots, \alpha_n) \in \times_{i=1}^n A_i^m$ ; we have

$$\begin{aligned} \varphi(\alpha) \left( \bigotimes_{i,j} v_{ij} \otimes \bigotimes_{i,j} \eta_{ij} \right) &= \left( \bigotimes_{i,j} \eta_{ij} \right) \left( \bigotimes_{i,j} \alpha_i v_{ij} \right) \\ &= \prod_{i=1}^n \eta_{im}(\alpha_i v_{im}) = T_{\alpha_1}^m \cdots T_{\alpha_n}^m = T_{\alpha}^m, \end{aligned}$$

where  $\bigotimes_{j=1}^m v_{ij} \in V_i^{\otimes m}$  and  $\bigotimes_{j=1}^m \eta_{ij} \in (V^*)^{\otimes m}$ .

□

We already mentioned in Section 6.1 that over a field of characteristic 0,  $\text{End}_{\text{GL}(V)}(V^{\otimes m}) = \langle \mathcal{S}_m \rangle_{\phi}$ , where  $\phi : \mathcal{S}_m \rightarrow \text{GL}(V^{\oplus m})$  is the representation of  $\mathcal{S}_m$  acting on  $V^{\otimes m}$  by a permutation of the copies of  $V$ . Therefore, by Theorem 6.8,  $\text{End}_{\mathbf{GL_d}}(V^{\otimes m}) = \langle \mathcal{S}_m^n \rangle_{\phi}$ , where  $\phi : \mathcal{S}_m^n \rightarrow \text{GL}(V^{\otimes m})$  is the representation of  $\mathcal{S}_m^n$  acting on  $V^{\otimes m} \cong V_1^{\oplus m} \otimes \cdots \otimes V_n^{\oplus m}$  where the  $i^{\text{th}}$  factor acts by permuting the copies of  $V_i$ .

Since  $\langle \mathcal{S}_m^n \rangle_{\phi}$  is generated by the elements of  $\mathcal{S}_m^n$ , to determine the multilinear invariants of  $\text{End}(V)^{\oplus m}$  acted upon by  $\mathbf{GL_d}$ , we need only look at the image of  $\alpha(\sigma)$  for each  $\sigma \in \mathcal{S}_m^n$ . Working this out gives the following functions.

**Definition 7.5.** For  $\sigma = (\sigma_1, \dots, \sigma_n) \in \mathcal{S}_m^n$ , let  $\sigma_i = (r_1 \cdots r_k)(s_1 \cdots s_l) \cdots$  be a disjoint cycle decomposition. For such a  $\sigma \in \mathcal{S}_m^n$ , define the *trace monomials* by  $\text{Tr}_{\sigma} = T_{\sigma_1} \cdots T_{\sigma_n}$  on  $\text{End}(V)^{\oplus m}$ , where

$$T_{\sigma_i} \left( \bigotimes A_{j_1}, \dots, \bigotimes A_{j_m} \right) = \text{Tr}(A_{ir_1} \cdots A_{ir_k}) \text{Tr}(A_{is_1} \cdots A_{is_l}) \cdots \quad (7.2)$$

and extend multilinearly.

**Corollary 7.6.** *The multilinear invariants of  $\text{End}(V)^{\oplus m}$  (over a field of characteristic 0) acted upon by  $\mathbf{GL_d}$  are generated by the functions  $\text{Tr}_{\sigma}$  for  $\sigma \in \mathcal{S}_m^n$ .*



### 7.1.1 Description of the Invariant Ring

Over a field characteristic 0 (which we assume from here on out), knowing the multilinear invariants of an action gives an easy way to determine all invariants. The idea is that, by repeating some of the entries of any multilinear invariant, one can come up with a generating set for all invariants. We now make this precise.

**Definition 7.7.** A function  $f \in k[V_1 \oplus \cdots \oplus V_r]$  is *multihomogeneous* of degree  $d = (d_1, \dots, d_r)$  if  $f(\lambda_1 v_1, \dots, \lambda_r v_r) = \lambda_1^{d_1} \cdots \lambda_r^{d_r} f(v_1, \dots, v_r)$ .

The coordinate ring  $k[V_1 \oplus \cdots \oplus V_r]$  can be graded with respect to the multidegree, and the multihomogeneous components of an invariant polynomial are themselves invariant. It is clear that the multihomogeneous components of  $k[V_1 \oplus \cdots \oplus V_r]^G$  generate this ring.

**Definition 7.8.** Suppose  $f \in k[V_1^{\oplus d_1} \oplus \cdots \oplus V_r^{\oplus d_r}]$  is a multilinear polynomial. Then the restitution of  $f$ ,  $\mathcal{R}f \in k[V_1 \oplus \cdots \oplus V_r]$  is defined by

$$\mathcal{R}f(v_1, \dots, v_r) = f(\underbrace{v_1, \dots, v_1}_{d_1}, \dots, \underbrace{v_r, \dots, v_r}_{d_r}).$$

Given a multihomogeneous polynomial  $F$ , we know that it is the restitution of a multilinear polynomial  $f$ . We call  $f$  the *polarization* of  $F$ . Thus, by polarizations and restitutions, we can move freely between multihomogeneous and multilinear polynomials. For a more formal definition of polarization, see [64, 60].

**Proposition 7.9** ([60]). *Assume  $\text{char } k = 0$  and  $V_1, \dots, V_m$  are representations of a group  $G$ . Then every multihomogeneous invariant  $f \in k[V_1 \oplus \cdots \oplus V_m]^G$  of degree  $d = (d_1, \dots, d_m)$  is the restitution of a multilinear invariant  $F \in k[V_1^{\oplus d_1} \oplus \cdots \oplus V_m^{\oplus d_m}]^G$ .*

We are now able to describe the invariant ring for a product of groups acting by conjugation on  $\text{End}(V)^{\oplus m}$ . Let us fix the notation that for a graded ring  $R$ ,  $R_{d_1, \dots, d_r}$  is the graded piece of  $(d_1, \dots, d_r)$  multihomogeneous functions.

**Theorem 7.10.** *Over a field of characteristic 0,*

$$k[\text{End}(V)^{\oplus m}]_{d_1, \dots, d_r}^{G_{\mathbf{d}}} \cong \bigotimes_{i=1}^n k[\text{End}(V_i)^{\oplus m}]_{d_1, \dots, d_r}^{G_{d_i}}.$$

*Proof.* We know that every multilinear invariant of  $k[\text{End}(V)^{\oplus m}]_{d_1, \dots, d_r}^{\text{GL}_d}$  is a multilinear invariant of  $\bigotimes_{i=1}^n k[\text{End}(V_i)^{\oplus m}]_{d_1, \dots, d_r}^{G_{d_i}}$  by Theorem 7.4. Then taking restitutions gives all invariants by Proposition 7.9.  $\square$

We now return to the case of  $\text{GL}_d$ . Consider an ordered multiset  $M = \{m_i\}$  with elements from  $[m]$ , and denote the group of permutations on  $|M|$  letters by  $\mathcal{S}_{|M|}$ . Let  $\sigma = (\sigma_1, \dots, \sigma_n) \in \mathcal{S}_{|M|}^n$ . Let  $(m_{r_1} \cdots m_{r_k})(m_{s_1} \cdots m_{s_l}) \cdots$  be a disjoint cycle decomposition for  $\sigma_i$ .

**Definition 7.11.** Given a multiset  $M$  and  $\sigma \in \mathcal{S}_{|M|}^n$ , define the trace monomials on  $\text{End}(V)^{\oplus m}$  by their action on simple tensors in  $\bigotimes_{i=1}^n \text{End}(V_i)$ ,

$$T_{\sigma_i}^M \left( \bigotimes_{j=1}^n A_{j1}, \dots, \bigotimes_{j=1}^n A_{jm} \right) = \text{Tr}(A_{im_{r_1}} \cdots A_{im_{r_k}}) \text{Tr}(A_{im_{s_1}} \cdots A_{im_{s_l}}) \cdots$$

$$\text{Tr}_{\sigma}^M = \prod_{i=1}^d T_{\sigma_i}^M$$

and extending multilinearly to  $\text{End}(V)^{\oplus m}$ .

Note that Definition 7.11 differs from Definition 7.5 in that it allows for repetition. As such, we see that these are restitutions of the multilinear invariants  $\text{Tr}_{\sigma}$  in Definition 7.5.

**Corollary 7.12.** *The ring of  $\text{GL}_d$ -invariants of  $\text{End}(V)^{\oplus m}$  is generated by the  $\text{Tr}_{\sigma}^M$ .*

Note that  $\text{Tr}_{\sigma}^M \text{Tr}_{\sigma'}^{M'} = \text{Tr}_{\sigma \sqcup \sigma'}^{M \sqcup M'}$ , where  $\sigma \sqcup \sigma'$  is the induced permutation on  $M \sqcup M'$ . So every invariant is a linear combination of the generators above.

## 7.2 Tensor Network Representations of Invariants

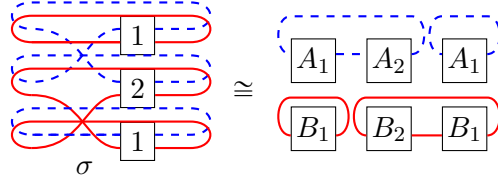
In the next section, we work out bounds for  $\beta_{\text{GL}_d}(\text{End}(V)^{\oplus m})$  for a particular case of  $\text{GL}_d$ . The approach is very combinatorial; as such it is convenient to consider invariants as represented by tensor networks.

**Observation 7.13.** Each trace monomial  $\text{Tr}_{\sigma}^M$  corresponds to a tensor network.

The polynomial  $\text{Tr}(A_1 \cdots A_k)$ , where each  $A_i$  is one of  $m$  possible  $n \times n$  matrices, has a representation as a tensor network by

$$\overbrace{[A_1] - \dots - [A_k]}.$$

A trace monomial  $\text{Tr}_\sigma^M$  acting on a simple tensor is a product of such loops. For example, let  $V = V_1 \otimes V_2$  and take  $M = \{1, 2, 1\} = \{m_1, m_2, m_3\}$  and  $\sigma = ((m_1 m_2)(m_3), (m_1)(m_2 m_3)) \in \mathcal{S}_3^2$ . Then the degree-three trace monomial  $\text{Tr}_\sigma^M(A_1 \otimes B_1, A_2 \otimes B_2)$  is equal to the tensor network:



Thus  $M$  tells us which elements of  $\text{End}(V)$  are selected and  $\sigma$  how to connect the wires. The trace monomials  $\text{Tr}_\sigma^M$  defined on simple tensors in  $\text{End}(V_1) \otimes \dots \otimes \text{End}(V_n)$  extend to all of  $\text{End}(V)^{\oplus m}$ . In particular, a matrix  $M$  in a tensor network whose wires correspond to copies of the vector space  $k^n$  decomposes as a sum of simple tensors of  $n \times n$  matrices:  $M = \sum_j (\otimes_{i=1}^n A_{ij})$ , and we are considering  $m$  such  $M$ .

For example, arbitrary  $M_1, M_2 \in \text{End}(V_1 \otimes V_2)^{\oplus m}$  are of rank at most four. So  $M_1 = \sum_{i=1}^4 A_{1i} \otimes B_{1i}$  and  $M_2 = \sum_{i=1}^4 A_{2i} \otimes B_{2i}$ , corresponding to a sum of tensor network diagrams as follows:

$$\begin{array}{c} \text{Diagram of } M_1 \text{ and } M_2 \end{array} = \sum_{i,j,k} \begin{array}{c} \text{Diagram of } A_{1i}, A_{2j}, A_{1k} \text{ and } B_{1i}, B_{2j}, B_{1k} \end{array}.$$

Multiset reordering corresponds to the action of an element of  $\mathcal{S}_{|M|}^n$  as an automorphism of the invariant ring. In the following we assume that  $M$  is in weakly increasing order, so the  $M$  in our example would become  $M = \{1, 1, 2\}$ .

### 7.3 Bounding the Degree of the Generators

In this section, we focus on the case  $V = (k^2)^{\otimes n}$ . We exploit the relationship between  $k[\text{End}(V)^{\oplus m}]^{\text{GL}_d}$  and the rings  $k[\text{End}(V_i)]^{\text{GL}(V_i)}$ . Much is known about ring of invariants of  $\text{End}(V)^{\oplus m}$  under the adjoint representation of  $\text{GL}(V)$ . We will be able to use several classical theorems that will allow us to

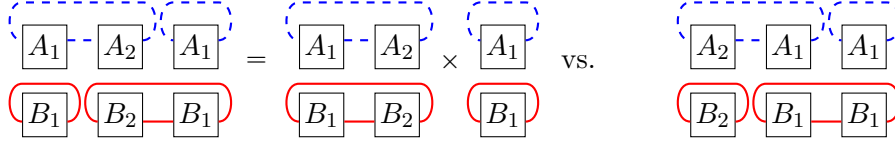


Figure 7.1: The network on the left factors as  $\text{Tr}_{(12),(23)}^{1,2,1} = \text{Tr}_{(12),(12)}^{1,2} \text{Tr}_{(),()}^1$  into generators of smaller degree, while the network  $\text{Tr}_{(12),(23)}^{2,1,1}$  on the right does not factor.

bound the degrees of generators for our related ring of invariants, beginning with the following:

**Theorem 7.14** ([60, 91]). *The ring  $k[\text{End}(V)^{\oplus m}]^{\text{GL}(V)}$  under the adjoint action is generated by*

$$\text{tr}_{i_1, \dots, i_k} := \text{Tr}(A_{i_1} \cdots A_{i_k}) \quad 1 \leq i_1, \dots, i_k \leq m$$

where  $k \leq \dim(V)^2$ . If  $n = \dim(V) \leq 3$ ,  $k \leq \binom{n+1}{2}$  suffices.

This, however, does not provide a bound on the degree. Note that the degree of  $\text{Tr}_{\sigma}^M$  as a polynomial in the matrix entries equals  $|M|$ . To see the issue, consider the tensor networks depicted in Figure 7.1. Some trace monomials such as  $\text{Tr}_{(12),(23)}^{1,2,1}$  factor into trace monomials of smaller degree,  $\text{Tr}_{(12),(23)}^{1,2,1} = \text{Tr}_{(12),(12)}^{1,2} \text{Tr}_{(),()}^1$ , while others such as  $\text{Tr}_{(12),(23)}^{2,1,1}$  do not.

We will need to bound the maximal degree of a trace monomial which does not factor. This will require a somewhat detailed combinatorial argument which will occupy the rest of this section. We begin with the following definitions.

**Definition 7.15.** The *size* of  $T_{\sigma_i}^M$  is defined to be the size of the largest cycle in the disjoint cycle decomposition of  $\sigma_i$ .

**Definition 7.16.** Given a minimal set of generators, we define the *girth* of  $k[\text{End}(V)^{\oplus m}]^{G_{\mathbf{t}}}$  as the tuple  $(w_1, \dots, w_n)$  where  $w_i$  is the maximum size of any  $T_{\sigma_i}^M$  appearing in a generator. The girth of a function  $\text{Tr}_{\sigma}^M$  is a tuple  $(s_1, \dots, s_n)$ , where  $s_i$  is the size of  $T_{\sigma_i}^M$ .

Note that the girth of the simple case  $k[\text{End}(V)]^{\text{GL}(V)}$  is simply the minimum  $k$  such that the functions  $\{\text{Tr}(A_{i_1} \cdots A_{i_k}) : 1 \leq i_1, \dots, i_k \leq m\}$  generate it. We put a partial ordering on girth as follows:  $(w_1, \dots, w_n) < (w'_1, \dots, w'_n)$  if  $\exists i$  such that  $w_i < w'_i$  and for no  $j$  do we have  $w'_j < w_j$ . The girth is bounded locally by the square of the dimension.

**Proposition 7.17.** *If  $(w_1, \dots, w_n)$  is the girth of  $k[\text{End}(V)^{\oplus m}]^{\text{GL}_d}$ , then  $w_i \leq y_i$ , where  $y_i$  is the girth of  $k[\text{End}(V_i)^{\oplus m}]^{\text{GL}(V_i)}$ . In particular for  $V = V_1 \otimes \dots \otimes V_n$ , the girth of  $k[\text{End}(V)^{\oplus m}]^{\text{GL}_d}$  is bounded by  $(t_1^2, \dots, t_n^2)$ . If  $t_i \leq 3$ , then the girth is bounded by  $((t_1+1)_2, \dots, (t_n+1)_2)$ .*

*Proof.* First note that  $T_{\sigma_i}^M$  lies in the invariant ring  $R_i = k[\text{End}(V_i)^{\oplus m}]^{\text{GL}(V_i)}$ . Thus it has size at most  $y_i$ , where  $y_i$  is the girth of  $R_i$ . Now apply Theorem 7.14.  $\square$

As we mentioned above, we are specifically interested in the case where  $V = (k^2)^{\otimes n}$ . The case where  $n = 1$  is well understood for  $k[\text{End}(k^2)^{\oplus m}]^{\text{GL}(k^2)}$  are well understood. We make use of the following theorem.

**Theorem 7.18** ([44]). *The ring  $k[\text{End}(k^2)^{\oplus m}]^{\text{GL}(k^2)}$  is minimally generated by*

$$\begin{aligned} \text{Tr}(A_i) & \quad 1 \leq i \leq m \\ \text{Tr}(A_{i_1} A_{i_2}) & \quad 1 \leq i_1, i_2 \leq m, \text{ and} \\ \text{Tr}(A_{i_1} A_{i_2} A_{i_3}) & \quad 1 \leq i_1 < i_2 < i_3 \leq m. \end{aligned}$$

So we may assume that  $\text{Tr}_{\sigma_i}^M$  is written in terms of the trace monomials in Theorem 7.18. The degree bound for generators of  $k[\text{End}(V)^{\oplus m}]^{\text{GL}_d}$  we give depends on the generic tensor rank of  $\text{End}(V)$  as an element of  $\bigotimes_{i=1}^n \text{End}(V_i)$ .

We begin the analysis by restricting to the tensors in  $\text{End}(V)$  which are rank one in  $\bigotimes_{i=1}^n \text{End}(V_i)$  in Sections 7.3.1 and 7.3.2 and then will consider linear combinations of these to obtain the general case in Section 7.3.3.

### 7.3.1 Girth at most $(2, \dots, 2)$ invariants operating on rank-one tensors

Let  $S$  be the subvariety of  $\text{End}(k^2)^{\otimes n}$  of rank one matrices, i.e. matrices of the form  $\bigotimes_{i=1}^n M_i$ ,  $M_i \in \text{End}(k^2)$ . First we bound the degree on a simpler ring, which we call  $R_{\text{trans}}$ , which is the subring of  $R = k[S^{\oplus m}]^{\text{GL}_d}$  generated by functions with girth at most  $(2, \dots, 2)$ . Note that in the case  $m = 2$ ,  $R_{\text{trans}} = R$  by Theorem 7.18

We want to show that for  $|M|$  sufficiently large, for any  $\sigma \in S_{|M|}^n$ ,  $\text{Tr}_{\sigma}^M$  factors as  $\text{Tr}_{\sigma}^M = \text{Tr}_{\sigma_a}^{M_a} \text{Tr}_{\sigma_b}^{M_b}$  for two disjoint multisets  $M_a, M_b$  with  $M = M_a \cup M_b$ .

**Definition 7.19.** Let  $M' \subseteq M$  be a sub-multiset. Let  $x_1, \dots, x_k \in M$ . We say that  $M'$  does not separate the points  $x_1, \dots, x_k$  if either  $\{x_i \mid 1 \leq i \leq k\} \subseteq M'$  or  $\{x_i \mid 1 \leq i \leq k\} \subseteq M \setminus M'$ . Otherwise, we say  $M'$  separates  $x_1, \dots, x_k$ .

**Definition 7.20.** Given a trace polynomial  $\text{Tr}_\sigma^M$ , we say that  $M' \subseteq M$  separates a monomial of  $\text{Tr}_\sigma^M$  if there is a trace monomial  $\text{Tr}(A_{im_1} \cdots A_{im_k})$ ,  $1 \leq k \leq 3$ , in  $\text{Tr}_\sigma^M$  such that  $M'$  separates  $m_1, \dots, m_k$ . Otherwise we say that  $M'$  does not separate monomials of  $\text{Tr}_\sigma^M$ .

We can now rephrase what it means for a trace polynomial to factor in a more convenient way.

**Definition 7.21.** A trace polynomial  $\text{Tr}_\sigma^M$  factors if there exists  $M' \subsetneq M$  such that  $M'$  does not separate monomials of  $\text{Tr}_\sigma^M$ .

We describe a *tableau-shape* for a multiset  $M$  that we will use to encode which elements of  $M$  we want to be inseparable. For the purposes of this subsection, at most two elements of  $M$  will be inseparable.

The tableau-shape of  $M$ ,  $\mathbb{M}$ , will be a collection of pairs,  $[-, -]$ , and singles  $[-]$ , which are unfilled, arranged in a particular pattern, see Figure 7.2 (a). Let  $M$  be drawn from  $[m]$  and let  $s = |M|$ . There will be  $m + 1$  rows. The first  $m$  rows each have  $\lfloor \frac{s}{2} \rfloor$  pairs and will be labeled from top to bottom by the elements of  $[m]$ . The last row, which we call the augmented row, contains  $s$  singles followed by  $\lfloor \frac{s}{2} \rfloor$  pairs. In the tableau-shape, pairs represent elements that cannot be separated by some  $M' \subseteq M$  for a trivial reason.

We fill  $\mathbb{M}$  by declaring the pairs  $m_1, m_2$  inseparable if  $\text{Tr}(A_{im_1} A_{im_2})$  appears in  $T_{\sigma_i}^M$ . We fill each the  $k$ th row with pairs of the form  $[m_1, m_2]$  for those inseparable pairs  $m_1, m_2 \in M$  where  $m_1 = m_2 = k$ . We call such pairs *duplicate pairs*. There are at most  $\lfloor \frac{s}{2} \rfloor$  of them. The pairs in the augmented row are of the form  $[m_1, m_2]$  and represent inseparable pairs  $m_1, m_2 \in M$  where  $m_1 \neq m_2$ . These are called *non-duplicate pairs*. The singles in the tableau contain the elements  $m_k$  appearing in  $\text{Tr}(A_{im_k})$ .

Note that there is some non-uniqueness in how to fill the augmented row. We call a filled tableau-shape simply a *tableau*. This gives a recipe for taking a trace polynomial  $T_{\sigma_i}^M$  and filling  $\mathbb{M}$  to give a tableau which we denote  $\mathbb{T}_{\sigma_i}$ . An example is given in Figure 7.2 (b).

Let  $\mathbb{T}$  be a tableau. Then choosing an  $1 \leq i \leq n$ , we can associate a function  $T_{\sigma_i}^M$ , for some  $\sigma_i$ , to  $\mathbb{T}$ . First let us disregard pairs or singles containing no elements. Secondly, if a pair has a single element, move that

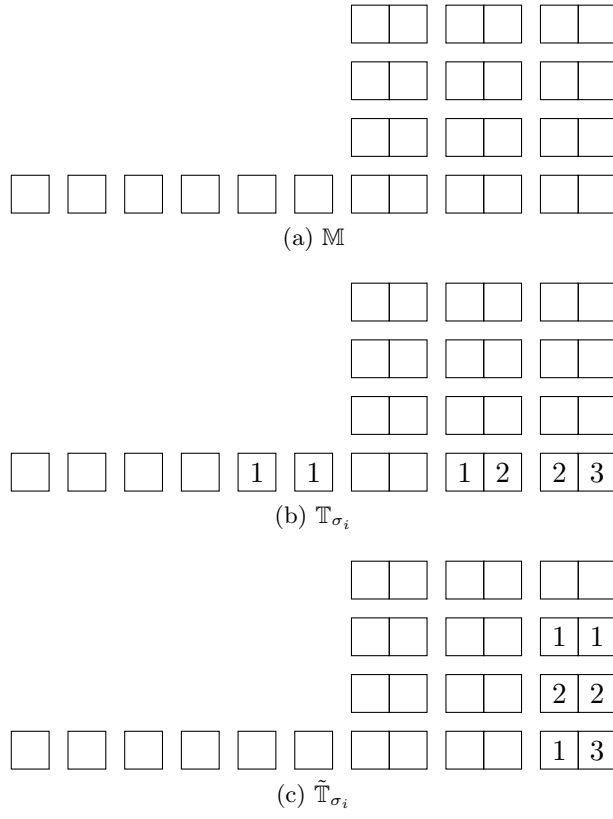


Figure 7.2: For  $M = \{1, 1, 1, 2, 2, 3\}$  and  $\sigma_i = (m_1)(m_2)(m_3m_4)(m_5m_6)$ .

element to a single in the augmented column. Then for every pair  $[m_1, m_2]$  in the tableau,  $\text{Tr}(A_{im_1}A_{im_2})$  appears in  $T_{\sigma_i}^M$ . For every single,  $[m_1]$ ,  $\text{Tr}(A_{im_1})$  appears. Let the  $f_{i,\mathbb{T}}$  denote the function that  $\mathbb{T}$  represents for choice of  $i$ . Note that  $f_{i,\mathbb{T}_{\sigma_i}} = T_{\sigma_i}^M$ .

We define an equivalence relation  $\sim$  on tableaux in the following way:  $\mathbb{T}_1 \sim \mathbb{T}_2$  if  $f_{i,\mathbb{T}_1} = f_{i,\mathbb{T}_2}$ , which will be independent of choice of  $i$ . Now let  $\mathbb{T}_1$  and  $\mathbb{T}_2$  be two different fillings of  $\mathbb{M}$ ,  $\mathbb{T}_1 \sim \mathbb{T}_2$ . Then if we allow horizontal permutations of the elements in a row and vertical permutations of elements in a column, we can permute  $\mathbb{T}_1$  into  $\mathbb{T}_2$ .

Now suppose we are considering a tableau  $\mathbb{T}_{\sigma_i}$  filled from  $\mathbb{M}$ . We make two types of adjustments. First, take any two elements  $m_1, m_2$  appearing in singles in the augmented row. We declare them inseparable. We do this until there is at most one element appearing in a single left.

Secondly, look at the non-duplicate pairs. Suppose there are two elements of  $[m]$  repeated in this row. So we have the pairs  $[m_1, m_2]$ ,  $[m_3, m_4]$  where  $m_1 = m_3$ . We replace these two pairs with  $[m_1, m_3]$ ,  $[m_2, m_4]$  and then move the pair  $[m_1, m_3]$  to the appropriate row of duplicate pairs. We repeat until all elements appearing in non-duplicate pairs are distinct. Lastly, we flush all elements are far right as possible. Note that the augmented row has at most  $2\lfloor \frac{m}{2} \rfloor + 1$  elements. Let us call this adjusted tableau  $\tilde{\mathbb{T}}_{\sigma_i}$ . An example is given in Figure 7.2 (c).

Now we consider a restricted set of permutations on our adjusted tableaux. Let  $P_{aug}$  be permutations of the elements of the augmented row and  $P_{vert}$  be permutations of elements within a column, for any column. Then our restricted permutations are  $P_{aug} \times P_{vert}$ . In fact, we can insist that the permutation in  $P_{aug}$  is always applied first, followed by the permutation from  $P_{vert}$ .

**Observation 7.22.** Consider two functions  $T_{\sigma_i}^M$  and  $T_{\sigma_j}^M$ . Since many types of permutations on  $\tilde{\mathbb{T}}_{\sigma_i}$  are trivial with respect to  $\sim$ , it is not hard to see that there is a permutation in  $P_{aug} \times P_{vert}$  that takes  $\tilde{\mathbb{T}}_{\sigma_i}$  to a tableau  $\mathbb{T}' \sim \mathbb{T}_{\sigma_j}$ , although they won't be equal in general.

**Theorem 7.23.** The  $\text{Tr}_{\sigma}^M$  with degree at most  $2(m+1)\lfloor \frac{m}{2} \rfloor + 2m+1$  generate  $R_{trans}$ .

*Proof.* Let us first consider  $\tilde{\mathbb{T}}_{\sigma_1}$ . The tableau  $\tilde{\mathbb{T}}_{\sigma_1}$  differs from a tableau  $\mathbb{T}'$  such that  $\mathbb{T}' \sim \mathbb{T}_{\sigma_i}$  by first applying a horizontal permutation in the augmented row and then vertical permutations in the columns, for any  $i$ .

Now suppose that  $|M| > 2(m+1)\lfloor \frac{m}{2} \rfloor + 2m+1$ . Then there are at least  $2\lfloor \frac{m}{2} \rfloor + 3$  filled columns in  $\tilde{\mathbb{T}}_{\sigma_1}$ . Indeed, suppose  $\tilde{\mathbb{T}}_{\sigma_1}$  had only  $2\lfloor \frac{m}{2} \rfloor + 2$



filled columns, what is the maximum  $|M|$ ? This is the case where there are  $\lfloor \frac{m}{2} \rfloor + 1$  duplicate pairs for every element of  $[m]$  as well as  $\lfloor \frac{m}{2} \rfloor$  non-duplicate pairs and one single. Thus the duplicate pairs contribute  $2m\lfloor \frac{m}{2} \rfloor + 2m$  to the size of  $|M|$  and the augmented row contributes  $2\lfloor \frac{m}{2} \rfloor + 1$ .

Now let  $M'$  be the elements filling the rightmost  $2\lfloor \frac{m}{2} \rfloor + 2$  columns, so  $M' \subsetneq M$ . Note that the restricted set of permutations we described above preserve  $M'$  as the subset of  $M$  filling the rightmost  $2\lfloor \frac{m}{2} \rfloor + 2$  columns of  $\mathbb{T}'$  and  $f_{\sigma_i, \mathbb{T}'} = T_{\sigma_i}^M$ . Furthermore,  $M'$  does not separate monomials for all  $T_{\sigma_i}^M$ . So  $\text{Tr}_{\sigma}^M$  factors.  $\square$

**Corollary 7.24.** *For  $m = 2$ , the  $\beta(k[S^{\oplus m}]) \leq 11$ .*

*Proof.* This follows from Theorem 7.23 by substituting in 2 for  $m$  and noticing that  $R_{trans} = R$  when  $m = 2$ .  $\square$

### 7.3.2 General girth, rank-one tensors

For general  $m$ ,  $R$  has girth  $(3, \dots, 3)$ . We adapt the ideas from the previous section to achieve our degree bound.

For a multiset  $M$ , we define a tableau-shape  $\mathbb{M}$  the same as before but with extra rows added above. We add  $\binom{m}{3}$  rows each with  $\lfloor \frac{s}{3} \rfloor$  triplets  $[--, --, --]$ ,  $s = |M|$ . We will think of each of these rows corresponding to a trace monomial  $\text{Tr}(A_{i_1}A_{i_2}A_{i_3})$ ,  $1 \leq i_1 < i_2 < i_3 \leq m$ .

We call a filled tableau-shape a tableau and, as before, we can associate to a tableau  $\mathbb{T}$  a trace polynomial  $T_{\sigma_i}^M$  for some  $i$ , which we denote  $f_{i, \mathbb{T}}$ , and we define the same equivalence relation  $\sim$  as before. Given a trace polynomial  $T_{\sigma_i}^M$ , we fill  $\mathbb{M}$  as before, but now placing the trace monomials  $\text{Tr}(A_{i_1}A_{i_2}A_{i_3})$  is the corresponding row. We call this tableau  $\mathbb{T}_{\sigma_i}$ . We have  $f_{i, \mathbb{T}_{\sigma_i}} = T_{\sigma_i}^M$ .

Given a tableau  $\mathbb{T}_{\sigma_i}$ , we will form another tableau  $\tilde{\mathbb{T}}_{\sigma_i}$  by first performing the two adjustments we did before. In addition, suppose there are three non-duplicate pairs in the augmented row:  $[m_1, m_2], [m_3, m_4], [m_5, m_6]$ , and we can assume that  $m_1 < m_2 < \dots < m_6$ . Then replace these three pairs with the triplets  $[m_1, m_2, m_3], [m_4, m_5, m_6]$ , which are then placed in their appropriate rows.

If there are two non-duplicate pairs and one single left afterwards, then one of the non-duplicate pairs contains two elements distinct from the element in the single. We declare this pair and single inseparable and place it in the appropriate triplet row. Otherwise, there may be two non-duplicate pairs left and no singles.

We also make adjustments on the duplicate pairs. For  $[m_1, m_1]$ ,  $[m_2, m_2]$ , and  $[m_3, m_3]$ , replace them with  $[m_1, m_2, m_3]$ ,  $[m_1, m_2, m_3]$

One may object that we separated  $m_3$  and  $m_4$ , for example, while they were clearly inseparable originally. This does not matter however since we do not require  $f_{i, \tilde{T}_{\sigma_i}} = f_{i, T_{\sigma_i}}$  and it will still be true that  $\tilde{T}_{\sigma_i}$  differs from  $T_{\sigma_i}$  by some combination of horizontal and vertical permutations. Note that the augmented row of  $\tilde{T}_{\sigma_i}$  has at most four elements. There are at most two non-empty rows of duplicate pairs.

**Observation 7.25.** Suppose we have two functions  $T_{\sigma_i}$  and  $T_{\sigma_j}$ . Once again, many of these permutations on  $\tilde{T}_{\sigma_i}$  are trivial with respect to  $\sim$ . Once again, there is a permutation  $P_{aug} \times P_{vert}$  that transforms  $\tilde{T}_{\sigma_i}$  into a tableau  $\mathbb{T}'$ ,  $\mathbb{T}' \sim T_{\sigma_j}$ . We can insist, as before, that the element of  $P_{aug}$  is applied first.

**Theorem 7.26.** For  $m \geq 3$ ,  $\beta(k[S^{\oplus m}]) \leq 6\binom{m}{3} + 16$ .

*Proof.* Let us first consider  $\tilde{T}_{\sigma_1}$ .  $\tilde{T}_{\sigma_1}$  differs from a tableau  $\mathbb{T}'$  such that  $\mathbb{T}' \sim T_{\sigma_i}$  by first applying a horizontal permutation in the augmented row and then vertical permutations in the columns, for any  $i$ .

Not suppose that  $|M| > 6\binom{m}{3} + 15$ . Then there are at least seven filled columns in  $\tilde{T}_{\sigma_1}$ . Indeed, suppose  $\tilde{T}_{\sigma_1}$  had only six filled columns, what is the maximum  $|M|$ ? The augmented row accounts for four elements. Every column (excluding elements in the augmented row) accounts for  $\binom{m}{3} + 2$  elements. Thus there are at most  $6\binom{m}{3} + 16$  elements in the tableau.

Now let  $M'$  be the elements filling the rightmost six columns, so  $M' \subsetneq M$ . Note that the restricted set of permutations we described above preserve  $M'$  as the subset of  $M$  filling the rightmost six columns of  $\mathbb{T}'$  and  $f_{\sigma_i, \mathbb{T}'} = T_{\sigma_i}^M$ . Furthermore,  $M'$  does not separate monomials for all  $T_{\sigma_i}^M$ . So  $\text{Tr}_{\sigma}^M$  factors.  $\square$

**Example 7.27.** Let  $V$  be the subvariety of  $\text{End}(\mathbb{C}^2)$  of rank one matrices. Let  $V^{\oplus 2}$  be acted on by  $G = \text{GL}(k^2) \times \text{GL}(k^2)$ . The ring  $k[V^{\oplus 2}]^G$  has 23 generators. For a pair of  $4 \times 4$  matrices  $(A, B) = (A_1 \otimes A_2, B_1 \otimes B_2)$ , the generators are:

$$\begin{array}{ll}
\text{Tr}(A_1 B_1) \text{Tr}(A_2 B_2), & \text{Tr}(A_1) \text{Tr}(A_1 B_1) \text{Tr}(B_2) \text{Tr}(A_2^2), \\
\text{Tr}(A_1) \text{Tr}(B_1) \text{Tr}(A_2 B_2), & \text{Tr}(A_1^2) \text{Tr}(B_1) \text{Tr}(A_2 B_2) \text{Tr}(A_2) \\
\text{Tr}(A_1 B_1) \text{Tr}(A_2) \text{Tr}(B_2), & \text{Tr}(B_1) \text{Tr}(A_1 B_1) \text{Tr}(A_2) \text{Tr}(B_2^2), \\
\text{Tr}(A_1^2) \text{Tr}(B_1^2) \text{Tr}(A_2 B_2)^2, & \text{Tr}(A_1) \text{Tr}(B_1^2) \text{Tr}(B_2) \text{Tr}(A_2 B_2), \\
\text{Tr}(A_1 B_1)^2 \text{Tr}(A_2^2) \text{Tr}(B_2^2), & \text{Tr}(A_1 B_1)^2 \text{Tr}(A_2^2) \text{Tr}(B_2)^2, \\
\text{Tr}(A_1)^2 \text{Tr}(B_1^2) \text{Tr}(A_2 B_2)^2, & \text{Tr}(A_1^2) \text{Tr}(A_2)^2, \\
\text{Tr}(A_1^2) \text{Tr}(B_1)^2 \text{Tr}(A_2 B_2)^2, & \text{Tr}(B_1^2) \text{Tr}(B_2)^2, \\
\text{Tr}(A_1 B_1)^2 \text{Tr}(A_2)^2 \text{Tr}(B_2^2), & \text{Tr}(B_1)^2 \text{Tr}(B_2^2), \\
\text{Tr}(A_1) \text{Tr}(A_2), & \text{Tr}(A_1)^2 \text{Tr}(A_2^2), \\
\text{Tr}(A_1^2) \text{Tr}(A_2^2), & \text{Tr}(B_1) \text{Tr}(B_2), \\
\text{Tr}(B_1^2) \text{Tr}(B_2^2), & \\
\text{Tr}(A_1 B_1) \text{Tr}(A_1) \text{Tr}(B_1) \text{Tr}(A_2^2) \text{Tr}(B_2)^2, & \\
\text{Tr}(A_1^2) \text{Tr}(B_1^2) \text{Tr}(A_2 B_2) \text{Tr}(A_2) \text{Tr}(B_2). &
\end{array}$$

We then extend these functions multilinearly to give  $23 \text{Tr}_\sigma^M$ . By Corollary 7.24, those  $\text{Tr}_\sigma^M$  of degree at most 11 generate  $k[V^{\oplus 2}]^G$ . We simply enumerated all  $\text{Tr}_\sigma^M$  up to degree 11 and removed those that were a product of functions of smaller degree. Notice that the highest degree in this example is 4.

### 7.3.3 General case, arbitrary tensors

**Theorem 7.28.** *Let  $r$  be the generic rank of  $\text{End}(V)$  as a  $4 \times \cdots \times 4 = 4^n$  tensor. Then  $\beta_{G_t}(k[\text{End}(V)^{\oplus m}]) \leq 6 \binom{rm}{3} + 16$ .*

*Proof.* This proof follows precisely the same logic as Theorem 7.26 with the difference that the tableaux can contain up to  $rm$  different matrices.  $\square$

## Chapter 8

# A Complete Set of Invariants for Local Unitary Equivalence

In this chapter we concern ourselves with the problem of finding a *complete set of invariants* for density operators. By this we mean a set of  $U_{\mathbf{d}}$ -invariant functions  $f_1, \dots, f_s$  such that two density operators  $\Psi_1$  and  $\Psi_2$  are in the same  $U_{\mathbf{d}}$ -orbit if and only if  $f_i(\Psi_1) = f_i(\Psi_2)$  for all  $i$ . As already mentioned, this problem is important for understanding quantum entanglement.

If we restrict the functions to be polynomials, Propositions 7.1 and 7.2 tell us that we can focus our attention instead on the ring  $\mathbb{C}[\text{End}(V)]^{\text{GL}_{\mathbf{d}}}$ . However, we may run into the problem that two density operators are in distinct  $\text{GL}_{\mathbf{d}}$ -orbits but cannot be distinguished by invariant polynomials. We show in Section 8.1 that  $\text{GL}_{\mathbf{d}}$ -orbits of density operators can always be separated by invariant polynomials.

Throughout this chapter, whenever possible, our theorems hold for the invariant ring  $k[\text{End}(V)]^{\text{GL}_{\mathbf{d}}}$ , where  $k$  is an algebraically closed field of characteristic zero. Otherwise,  $k = \mathbb{C}$ . We wish to find a finite (and preferably small) generating set of invariants. We know that this ring is generated by the functions  $\text{Tr}_{\sigma}^M$  and we can compute an upper bound for  $\beta_{\text{GL}_{\mathbf{d}}}(\text{End}(V))$  by studying the ring  $k[\text{End}(V)]^{\text{GL}_{\mathbf{d}}}$  much more thoroughly.

**Proposition 8.1** ([46]). *For a reductive action  $G \curvearrowright V$ , there exists a set of homogeneous algebraically independent polynomials  $p_1, \dots, p_s$  such that  $k[V]^G$  is a finitely generated module over  $k[p_1, \dots, p_s]$ .*

This is a special case of Noether's Normalization Lemma which gen-

eralizes this statement to all commutative Noetherian  $L$ -algebras, with  $L$  any field. The polynomials  $p_1, \dots, p_s$  are called a *homogeneous system of parameters*, or an *hsop* for short.

In the context of invariant theory, these polynomials are also sometimes called *primary invariants*. For reductive groups, since  $k[V]^G$  is Cohen-Macaulay,  $k[V]^G$  is a finite *free* module over  $P := k[p_1, \dots, p_s]$ . That is, there exist *secondary invariants*  $h_1, \dots, h_r$  such that  $k[V]^G \cong \bigoplus_{i=1}^r Ph_i$ . This fact will be useful in computing a bound for  $\beta_{\mathrm{GL}_{\mathbf{d}}}(\mathrm{End}(V))$ .

## 8.1 Closed Orbits

We first give an a sufficient condition for  $M \in \mathrm{End}(V)$  to have a closed  $\mathrm{GL}_{\mathbf{d}}$ -orbit, where  $V$  is a Hilbert space throughout this section. We show that, in particular, normal matrices over  $\mathbb{C}$  satisfy the given properties. Since density operators are Hermitian, they are immediately normal.

**Definition 8.2.** Given  $M \in \mathrm{End}(V)$ , we define a decomposition  $V = W \oplus W^\perp$  to be *separable* if there exists a cocharacter of  $\mathrm{GL}_{\mathbf{d}}$ ,  $\lambda(t)$  such that  $\forall w \in W$ ,  $\lim_{t \rightarrow 0} \lambda(t)w = 0$ , and  $\forall w \in W^\perp$ ,  $w \neq 0$ ,  $\lim_{t \rightarrow 0} \lambda(t)w \neq 0$ . We call  $\lambda(t)$  the *separating subgroup* of the decomposition.

Given an arbitrary cocharacter of  $\mathrm{GL}_{\mathbf{d}}$ , it is not clear that there is necessarily a separable decomposition that one can associate to it. The following lemma allows us to replace a cocharacter by one that does have a separable decomposition associated to it that does not affect limits.

**Lemma 8.3.** *Let  $\lambda(t)$  be a cocharacter of  $\mathrm{GL}_{\mathbf{d}}$ . Then there exists another cocharacter  $\mu(t)$  such that the following hold:*

- (a)  $\lim_{t \rightarrow 0} \lambda(t)M\lambda(t)^{-1} = \lim_{t \rightarrow 0} \mu(t)M\mu(t)^{-1}$  for all  $M \in \mathrm{End}(V)$  such that the limit exists.
- (b)  $\mu(0) := \lim_{t \rightarrow 0} \mu(t)$  exists and is a matrix.
- (c) Unless  $\lambda(t) = t^\alpha \mathrm{id}$ , then  $\mu(0)$  has two nontrivial eigenspaces with eigenvalues 0, 1.

*Proof.* We can diagonalize  $\lambda(t)$  by some element  $g \in \mathrm{GL}_{\mathbf{d}}$ . Thus it suffices to prove the above statements for diagonal cocharacters. If  $\lambda(t)$  is a diagonal cocharacter, the diagonal entries are of the form  $t^{\alpha_i}$ ,  $\alpha_i \in \mathbb{Z}$ , as previously mention in Section 6.2. Let  $\alpha_m$  be the most negative exponent. Then let  $\mu(t) = t^{-\alpha_m} \lambda(t)$ . We see that for any  $M \in \mathrm{End}(V)$ ,  $\lambda(t)M\lambda(t) =$

$\mu(t)M\mu(t)^{-1}$ . Therefore  $\lim_{t \rightarrow 0} \lambda(t)M\lambda(t)^{-1} = \lim_{t \rightarrow 0} \mu(t)M\mu(t)^{-1}$  whenever the limit exists.

Furthermore, we see that  $\mu(t)$  has diagonal entries all non-negative powers of  $t$ . Therefore,  $\lim_{t \rightarrow 0} \mu(t)$  exists and is in fact equal to  $\mu(0)$ . Furthermore, unless  $\mu(t) = t^\alpha \text{id}$ , which occurs precisely when  $\lambda(t) = t^\beta \text{id}$ ,  $\mu(0)$  will have both zeros and ones on the diagonal. Thus it will have to non-trivial eigenspaces with eigenvalues 0, 1.  $\square$

We now show how to construct separable decompositions as it is not clear that they necessarily exist. We must use cocharacters of the form as in Lemma 8.3.

**Lemma 8.4.** *Given a cocharacter as in Lemma 8.3, except for  $\lambda(t) = t^\alpha \text{id}$ , we can associate it to a separable decomposition for which it is the separating subgroup.*

*Proof.* Let  $\mu(t)$  be a cocharacter as in Lemma 8.3. Then we know that  $\mu(0) := \lim_{t \rightarrow 0} \mu(t)$  exists and is a matrix. Then  $\mu(0)$  has two eigenspaces, one attached to eigenvalue 1 and the other to eigenvalue 0. Let  $W$  be the null space of  $\mu(0)$ . Then consider the decomposition  $V = W \oplus W^\perp$ . Then  $\forall w \in W$ ,  $\lim_{t \rightarrow 0} \mu(t)W = \mu(0)W = 0$ , and  $\forall w \in W^\perp$  then  $\lim_{t \rightarrow 0} \mu(t)w = \mu(0)w$ , which projects  $W^\perp$  onto the eigenspace attached to the eigenvalue 1. This means that the only  $v \in W^\perp$  such that  $\mu(0)v = 0$  is  $v = 0$ . So this is a separable decomposition for which  $\mu(t)$  is the separating subgroup.  $\square$

Let us analyze which decompositions are separable. Let us first analyze the case that  $\lambda(t) = \bigotimes_{i=1}^n \lambda_i(t)$  is as in Lemma 8.3 and is diagonal. Then  $\lambda_i(t)$  is diagonal and can be taken to have diagonal entries with all non-negative powers of  $t$ . Thus we can decompose  $V_i = W_i \oplus W_i^\perp$  where  $\lim_{t \rightarrow 0} \lambda(t)w = 0$  for all  $w \in W_i$  and  $\lambda(t)w = w$  for all  $w \in W_i^\perp$ . Then  $V_1 \otimes \cdots \otimes V_{i-1} \otimes W_i \otimes \cdots \otimes V_n$  gets sent to zero by  $\lambda(t)$ . It is easy to see that every separable decomposition for a diagonal cocharacter is of the form

$$(V_1 \otimes \cdots \otimes W_i \otimes \cdots \otimes V_n) \oplus (V_1 \otimes \cdots \otimes W_i^\perp \otimes \cdots \otimes V_n).$$

From here, it is easy to see that every separable decomposition is of the same form by taking the  $\text{GL}_d$ -orbits of diagonal cocharacters.

Given a matrix  $M \in \text{End}(V)$ , we are interested in separable decompositions  $W \oplus W^\perp$  such that  $M(W) \subseteq W$ . Let  $P_W$  and  $P_{W^\perp}$  be the projection operators onto each of the two subspaces. Then define  $M|_W := P_W(M)$  and  $M|_{W^\perp} := P_{W^\perp}(M)$ .

**Lemma 8.5.** *Given a separable decomposition  $V = W \oplus W^\perp$  with projections  $P_W, P_{W^\perp}$ , there exists a separating subgroup  $\lambda(t)$  that is a cocharacter of  $\mathrm{GL}_{\mathbf{d}}$  such that*

$$\lambda'(t) = \begin{matrix} & W & W^\perp \\ \begin{matrix} W \\ W^\perp \end{matrix} & \begin{pmatrix} tI & 0 \\ 0 & I \end{pmatrix} \end{matrix}$$

*Proof.* We saw above that every separable decomposition is of the form

$$(V_1 \otimes \cdots \otimes W_i \otimes \cdots \otimes V_n) \oplus (V_1 \otimes \cdots \otimes W_i^\perp \otimes \cdots \otimes V_n)$$

and it is easy to see that  $\lambda(t) = \bigotimes_{i=1}^n \lambda_i(t)$  where  $\lambda_j(t) = \mathrm{id}$  for  $j \neq i$  and

$$\lambda_i(t) = \begin{matrix} & W_i & W_i^\perp \\ \begin{matrix} W_i \\ W_i^\perp \end{matrix} & \begin{pmatrix} tI & 0 \\ 0 & I \end{pmatrix} \end{matrix}$$

satisfies the above conditions.  $\square$

**Proposition 8.6.** *For every separable decomposition  $V = W \oplus W^\perp$  such that  $M(W) \subseteq W$ ,  $M|_W \oplus M|_{W^\perp}$  is in the orbit closure of  $M$ .*

*Proof.* We can write  $M$  as

$$M = \begin{matrix} & W & W^\perp \\ \begin{matrix} W \\ W^\perp \end{matrix} & \begin{pmatrix} A & B \\ 0 & C \end{pmatrix} \end{matrix}$$

Let  $\lambda(t)$  be a separating subgroup of the decomposition  $V = W \oplus W^\perp$  as in Lemma 8.5. Let  $tI = P_W(\lambda(t))$ , then we get a cocharacter,

$$\begin{matrix} & W & W^\perp \\ \begin{matrix} W \\ W^\perp \end{matrix} & \begin{pmatrix} tI & 0 \\ 0 & I \end{pmatrix} \end{matrix} \begin{matrix} & W & W^\perp \\ \begin{matrix} W \\ W^\perp \end{matrix} & \begin{pmatrix} A & B \\ 0 & C \end{pmatrix} \end{matrix} \begin{matrix} & W & W^\perp \\ \begin{matrix} W \\ W^\perp \end{matrix} & \begin{pmatrix} t^{-1}I & 0 \\ 0 & I \end{pmatrix} \end{matrix} = \begin{matrix} & W & W^\perp \\ \begin{matrix} W \\ W^\perp \end{matrix} & \begin{pmatrix} A & tB \\ 0 & C \end{pmatrix} \end{matrix}$$

which we see takes  $M \rightarrow M|_W \oplus M|_{W^\perp}$  as  $t \rightarrow 0$ .  $\square$

**Theorem 8.7.** *A matrix  $M$  has a closed  $\mathrm{GL}_{\mathbf{d}}$ -orbit if for every separable decomposition  $V = W \oplus W^\perp$  such that  $M(W) \subseteq W$ , then  $M(W^\perp) \subseteq W^\perp$ .*

*Proof.* Suppose that  $M$  does not have a closed orbit, so it can be written as  $M = M_s + M_n$  where  $M_s$  has a closed orbit and  $M_n$  is in the null cone. Then by Theorem 6.12, there is a cocharacter  $\lambda(t)$  taking  $M \rightarrow M_s$ . We can

assume that  $\lambda(t)$  satisfies the properties of Lemma 8.3. Letting  $W$  be the kernel of  $\lambda(0)$ , we see that  $V = W \oplus W^\perp$  is a separable decomposition.

Let  $w \in W$ . We note that  $\lambda(t)Mw = \lambda(t)M\lambda(t)^{-1}\lambda(t)w$ . We know that  $\lambda(t)M\lambda(t)^{-1}$  is a matrix in which only non-negative powers of  $t$  appears. Furthermore, every entry of  $\lambda(t)w$  is scaled by some positive power of  $t$ . Therefore every element of  $\lambda(t)Mw$  is scaled by a positive power of  $t$ , so  $\lim_{t \rightarrow 0} \lambda(t)Mw = 0$ . Therefore  $M(W) \subseteq W$ .

Notice that a similar argument shows that  $M_s(W) \subseteq W$  and therefore we can write

$$M_s = \begin{matrix} & W & W^\perp \\ \begin{matrix} W \\ W^\perp \end{matrix} & \begin{pmatrix} A & B \\ 0 & C \end{pmatrix} \end{matrix}$$

However, by Proposition 8.6, we can assume that  $B = 0$ . That is to say,  $M_s(W^\perp) \subseteq M_s(W^\perp)$ .

If  $u \in W^\perp$ , then  $\lim_{t \rightarrow 0} \lambda(t)u$  lies in the eigenspace of  $\lambda(0)$  attached to the eigenvalue of 1 (it may not be the case that this eigenspace is orthogonal to the kernel of  $\lambda(0)$ ). However, we note that  $\lambda(t)M_n\lambda(t)^{-1}$  has every entry scaled by a positive power of  $t$ , and thus  $\lambda(t)M\lambda(t)^{-1}\lambda(t)u$  has all entries scaled by some positive power of  $t$  and thus  $\lim_{t \rightarrow 0} \lambda(t)M_nu = 0$ . This implies that  $M_nu$  is in  $W$  and therefore, and since  $M_s(u) \in W^\perp$ ,  $W^\perp$  is not an invariant subspace  $\square$

We can show that matrices that respect orthogonal decompositions have closed orbits. The prime example are normal matrices as these are precisely the matrices with an orthogonal basis by the spectral theorem.

**Theorem 8.8.** *For  $\text{GL}_{\mathbf{d}} \hookrightarrow \text{End}(V)^{\oplus m}$ ,  $V$  a finite dimensional Hilbert space, then those tuples of matrices, each with an orthogonal eigenbasis, have closed orbits.*

*Proof.* It suffices to show that for  $\text{GL}_{\mathbf{d}} \hookrightarrow \text{End}(V)$ , matrices with an orthogonal eigenbasis have closed orbits. Then the result follows from the fact that, if such a tuple did not have a closed orbit, then projecting onto one of the copies of  $\text{End}(V)$  would induce a non-trivial limit point, implying that the matrix in that coordinated did not have a closed orbit.

Let  $M$  have an orthogonal eigenbasis. Then let  $V = W \oplus W^\perp$  be a separable decomposition such that  $M(W) \subseteq W$ . It must be that  $W$  is a direct sum of eigenspaces of  $M$  (here, by eigenspace, we mean any subspace which  $M$  acts on by scaling). Since the eigenspaces of  $M$  are orthogonal (in the sense that given two vectors in two different eigenspaces, they are orthogonal), we immediately have that  $W^\perp$  is a direct sum of eigenspaces.



Thus  $W^\perp$  is an invariant subspace of  $M$ . Then applying Theorem 8.7, we get that  $M$  has a closed orbit.  $\square$

**Corollary 8.9.** *The  $\mathrm{GL}_{\mathbf{d}}$ -orbits of tuples of density matrices are closed, so can be separated by polynomial invariants. Moreover, two Hermitian matrices are in the same  $\mathrm{GL}_{\mathbf{d}}$ -orbit if and only if they are in the same  $\mathrm{U}_{\mathbf{d}}$ -orbit.*

*Proof.* We know from Proposition 7.2 that two density operators are in the same  $\mathrm{GL}_{\mathbf{d}}$ -orbit if and only if they are in the same  $\mathrm{U}_{\mathbf{d}}$ -orbit. We know from Theorem 8.8 that tuples of density operators have closed orbits. We know from Theorem 6.9 that two closed orbits can be distinguished by invariants if and only if they are distinct.  $\square$

For the adjoint action of  $\mathrm{GL}(V) \curvearrowright \mathrm{End}(V)$ , the normal matrices are precisely all of the closed orbits. However, this is not necessarily all of the closed  $\mathrm{GL}_{\mathbf{d}}$ -orbits. We know that the null cone  $\mathcal{N}_{\mathrm{GL}_{\mathbf{d}}}$  of  $\mathrm{GL}_{\mathbf{d}} \curvearrowright \mathrm{End}(V)$  is contained in the null cone  $\mathcal{N}_{\mathrm{GL}(V)}$  of  $\mathrm{GL}(V) \curvearrowright \mathrm{End}(V)$ . Suppose they were equal. Then by Proposition 6.14, we would have  $k[\mathrm{End}(V)]^{\mathrm{GL}_{\mathbf{d}}} = k[\mathrm{End}(V)]^{\mathrm{GL}(V)}$ . However, we know this is not true for  $\mathrm{GL}_{\mathbf{d}} \neq \mathrm{GL}(V)$  by Corollary 7.12. Therefore,  $\mathcal{N}_{\mathrm{GL}_{\mathbf{d}}} \subsetneq \mathcal{N}_{\mathrm{GL}(V)}$ . Then take an element  $N \in \mathcal{N}_{\mathrm{GL}(V)} \setminus \mathcal{N}_{\mathrm{GL}_{\mathbf{d}}}$ . Since  $N$  is not in the null cone of  $\mathrm{GL}_{\mathbf{d}}$ , its orbit closure contains a closed orbit that is different from the origin and thus there is a closed orbit in  $\mathcal{N}_{\mathrm{GL}(V)} \setminus \mathcal{N}_{\mathrm{GL}_{\mathbf{d}}}$  by Theorem 6.10. Let  $M$  be in that closed orbit. Since it is nilpotent, it is not normal.

**Corollary 8.10.** *The functions  $\mathrm{Tr}_{\sigma}^M$  form a complete set of invariants for tuples of density operators under the action of  $\mathrm{U}_{\mathbf{d}}$ .*

*Proof.* This follows from Corollary 8.9 and Corollary 7.12.  $\square$

So we know that two tuples of density operators are not in the same  $\mathrm{U}_{\mathbf{d}}$  orbit if and only if there is some  $\mathrm{Tr}_{\sigma}^M$  on which they take different values. We know from Theorem 6.4, that there exists a finite set of functions  $\mathrm{Tr}_{\sigma}^M$  that forms a complete system of invariants. However, this theorem does not tell us what such a finite set may be.

There are some general bounds known for reductive groups. We mention one such here, giving a complete, finite set of invariants. For a reductive group  $G$ , it is in particular an affine variety defined by polynomials  $h_1, \dots, h_\ell$  in the ring  $k[X_1, \dots, X_\ell]$ . Let  $H := \max\{\mathrm{degree}(h_1), \dots, \mathrm{degree}(h_\ell)\}$ . For a rational representation  $\rho : G \rightarrow \mathrm{GL}(V)$ , for  $g \in G$  we have that  $\rho(g)$  is equal to the matrix  $\{a_{ij}(g)\}$  for maps  $a_{ij}$  in the coordinate ring of  $G$ . Let  $A := \max\{\mathrm{degree}(a_{ij})\}$ .

**Theorem 8.11** ([26]). *For a reductive group  $G$  and a rational representation  $\rho : G \rightarrow \mathrm{GL}(V)$ , if  $\rho$  has finite kernel then*

$$\beta(k[V]^G) \leq \max\{2, \frac{3}{8} \dim(V) H^{t-m} A^m\}$$

where  $m := \dim(G)$ .

We note that while the representation of  $\mathrm{GL}_{\mathbf{d}} \hookrightarrow \mathrm{End}(V)^{\oplus m}$  does not have a finite kernel, it is the same action as  $\mathrm{SL}_{\mathbf{d}} \hookrightarrow \mathrm{End}(V)^{\oplus m}$  which does have a finite kernel. We note that  $H = \dim(V)$  as  $\mathrm{SL}_{\mathbf{d}}$  is defined by quadratic polynomials (for the Segre variety) and the equation of setting the determinant equal to 1. Furthermore, it is easy to see that for conjugation  $A = 2$ . Lastly,  $\dim(\mathrm{SL}_{\mathbf{d}}) = \sum_{i=1}^n (\dim(V_i)^2 - 2) + 1$  since is the affine cone over the image of the Segre embedding of groups of dimension  $\dim(V_1)^2 - 1, \dots, \dim(V_n)^2 - 1$ .

**Corollary 8.12.** *The group  $k[\mathrm{End}(V)^{\oplus m}]^{\mathrm{GL}_{\mathbf{d}}}$  is generated by the polynomials  $\mathrm{Tr}_{\sigma}^M$  of degree at most  $\dim(V)^{(\dim(V)^2 - D)} 2^D$  where*

$$D = \sum_{i=1}^n (\dim(V_i)^2 - 2) + 1.$$

## 8.2 Description of the Null Cone

This and the following section are dedicated to results to help find smaller generating sets of  $k[\mathrm{End}(V)]^{\mathrm{GL}_{\mathbf{d}}}$ , although we do not find such sets in this dissertation. One strategy is to use Proposition 6.14, so we first investigate the null cone  $\mathcal{N}_{\mathrm{GL}_{\mathbf{d}}}$ . By Theorem 6.12, we need to determine those matrices  $M \in \mathrm{End}(V)$  that can be taken to 0 by a cocharacter. Note that if  $\lambda(t) = (\lambda_1(t), \dots, \lambda_n(t))$  is a cocharacter of  $\mathrm{GL}_{\mathbf{d}}$ , we can choose  $g = (g_1, \dots, g_n) \in \mathrm{GL}_{\mathbf{d}}$  so that

$$g\lambda(t)g^{-1} = \bigotimes_{i=1}^n g_i\lambda_i(t)g_i^{-1},$$

where each  $g_i\lambda_i(t)g_i^{-1}$  is a diagonal matrix. We then follow a similar approach to determining  $\mathcal{N}_{\mathrm{GL}_{\mathbf{d}}}$  as we did in Example 6.3.

**Definition 8.13.** A point  $M$  is *viable* if it can be taken to a point in the unique closed orbit contained in its orbit closure by a diagonal cocharacter.

Given a viable matrix  $M$  with Jordan decomposition  $M = M_s + M_n$ , we would like to be able to analyze the matrices  $M_s$  and  $M_n$  more closely. Unfortunately, as Jordan decompositions are not unique, we make a choice of Jordan decomposition.

**Definition 8.14.** We define a Jordan decomposition (as in Definition 6.15) of a viable matrix  $M = M_s + M_n$  to be a *maximal Jordan decomposition* if for every diagonal cocharacter  $\lambda(t)$ ,  $\lim_{t \rightarrow 0} \lambda(t)M_s\lambda(t)^{-1} = M_s$  or is undefined. Then  $M_s$  is the stable part of  $M$  and  $M_n$  is the unstable part of  $M$ .

The terminology of stable and unstable parts is justified for viable matrices because of the uniqueness of the decomposition supplied by Corollary 8.19. This does not, however, lift to a unique choice of decomposition for every matrix.

We quickly recall the definition of a quiver.

**Definition 8.15.** A *quiver* is a directed graph, denoted by a tuple  $Q = (Q_0, Q_1, h, t)$ , where  $Q_0$  are the vertices,  $Q_1$  the edges (or arrows), and maps  $h, t : Q_1 \rightarrow Q_0$ , denoting the head and tail of an arrow, respectively. A quiver can thus be viewed as a category where  $Q_0$  are the objects,  $Q_1$  the morphisms,  $h$  and  $t$  the domain and codomain maps.

**Definition 8.16.** Viewing a quiver  $Q$  as a category, a representation of a quiver is a functor from  $Q \rightarrow \text{Vect}_k$ .

We describe a way to associate a representation of a quiver to a viable matrix. Suppose we are given an  $d_1 \cdots d_n \times d_1 \cdots d_n$  matrix  $M = \{m_{ij}\} \in \text{End}(V)$ . Define  $r(Q_M)$  as the quiver representation on the complete digraph with nodes  $\{1, \dots, d_1 \cdots d_n\}$ , dimension vector  $\alpha = (1, \dots, 1) \in \mathbb{N}^{d_1 \cdots d_n}$ , and maps  $i \rightarrow j$  given by multiplication by  $m_{ij}$ .

The diagonal subgroup inside of  $\text{GL}_{\mathbf{d}}$ , naturally has more invariants than the action by the subgroup of diagonal matrices in  $\text{GL}(V)$  which are well understood [66]. Here we have that the group acting on  $\text{End}(V)$  is the tensor product of diagonal groups on each  $V_i$ . We need to find which tuples of arrows have the property that the product of the scalars on each arrow is invariant under the diagonal  $\text{GL}_{\mathbf{d}}$  action. We call the product of the scalars on the arrows of a subgraph  $H$  of the complete digraph on  $d_1 \cdots d_n$  vertices the *weight* of the subgraph, denoted  $wt(H)$ .

Let  $T = \{t_{ij}\}$  be a diagonal matrix in  $\text{GL}_{\mathbf{d}}$  acting on  $M = \{m_{ij}\}$ . Then  $TMT^{-1} = \{\mu_{ij}\}$  where  $\mu_{ij} = t_{ii}m_{ij}t_{jj}^{-1}$ . The diagonal matrices of  $\text{GL}_{\mathbf{d}}$  are cut out from the diagonal matrices of  $\text{GL}(V)$  by the equations of the affine cone over the Segre embedding of  $\mathbb{P}_{\mathbb{C}}^{d_1-1} \times \cdots \times \mathbb{P}_{\mathbb{C}}^{d_n-1}$ . This is

just the observation that  $\mathrm{GL}_{\mathbf{d}}$  is the affine cone over the Segre embedding  $\times_{i=1}^n \mathrm{GL}(V_i) \rightarrow \bigotimes_{i=1}^n \mathrm{GL}(V_i)$  (for an exposition on the Segre variety, see for example [64]).

If  $t_{ii}t_{jj} - t_{kk}t_{\ell\ell}$  is such an equation, then  $m_{ik}m_{j\ell}, m_{i\ell}m_{jk}, m_{ki}m_{j\ell}$ , and  $m_{\ell i}m_{kj}$  are invariant under the action of any such diagonal, local  $T$ . These are all the new invariants caused by restriction from diagonal  $\mathrm{GL}(V)$  to diagonal  $\mathrm{GL}_{\mathbf{d}}$ , since any others would give new relations satisfied by points on the Segre variety. Together with the cyclic invariants discussed in Section 6.3, these generate the invariants of the diagonal  $\mathrm{GL}_{\mathbf{d}}$  action.

**Definition 8.17.** Denote the set of subgraphs  $H$  of  $Q_M$  whose weights  $wt(H)$  are diagonal- $\mathrm{GL}_{\mathbf{d}}$ -invariant by  $C(M)$ .

The diagonal  $\mathrm{GL}_{\mathbf{d}}$  orbit  $O$  corresponding to a representation  $r(Q_M)$  is  $\mathrm{GL}_{\mathbf{d}}.M$ . We now need to say when the representations  $r(Q_M)$  correspond to closed orbits  $\mathrm{GL}_{\mathbf{d}}.M$ .

Suppose that we now have a quiver representation  $r(Q_M)$ . We want to find the  $\mathrm{GL}_{\mathbf{d}}$ -stable part of  $M$ . Beginning with  $r(Q_M)$ , define another representation  $r(Q_M)_s$  by setting the map on an arrow to zero if and only if every subgraph  $c \in C$  which includes the arrow has weight zero.

**Theorem 8.18.** *Given the representation  $r(Q_M)$ , the matrix associated to  $r(Q_M)_s$  is the stable part of  $M$ .*

*Proof.* Let  $r(Q_M)_n = r(Q_M) - r(Q_M)_s$ . We see that the weight of every subgraph in  $C(M)$  is 0 and thus  $r(Q_M)_n$  lies in the null cone of the diagonal  $\mathrm{GL}_{\mathbf{d}}$  action and thus the  $\mathrm{GL}_{\mathbf{d}}$  null cone. We just need to show that  $r(Q_M)_s$  has a closed orbit. Every arrow with a nonzero map lies on a subgraph in  $C(M)$  with nonzero weight. Let  $T = \{t_{ij}\}$  be a diagonal matrix in  $\mathrm{GL}_{\mathbf{d}}$  acting on  $M = \{m_{ij}\}$ . Since  $TMT^{-1} = \{\mu_{ij}\}$  where  $\mu_{ij} = t_{ii}m_{ij}t_{jj}^{-1}$ , taking the limit  $t \rightarrow 0$  in any diagonal cocharacter takes a map on an arrow to zero, infinity, or leaves it unchanged. We see that no arrow with a nonzero map can be taken to the zero map by a cocharacter, otherwise a nonzero invariant will then take the value zero in the limit. Therefore the limit of every cocharacter is either undefined or it leaves the representation fixed. By the Hilbert-Mumford criterion, this implies that it has a closed orbit.  $\square$

**Corollary 8.19.** *The unique maximal  $\mathrm{GL}_{\mathbf{d}}$  Jordan decomposition of a viable  $M$  is  $M = M_s + M_n$ , where  $M_s$  and  $M_n$  are the matrices associated to the representations  $r(Q_M)_s$  and  $r(Q_M)_n$ , respectively.*

Theorem 8.18 implies that if  $M$  is a matrix which is both viable and unstable, then the weight of every subgraph in  $C(M)$  is zero. We can now

give a description of the null cone of  $\mathrm{GL}_{\mathbf{d}}$  as the orbits of viable unstable matrices. The following theorem gives a more concrete description of such matrices.

**Definition 8.20.** A *subsystem* of  $V = \bigotimes_{i=1}^n V_i$  is a vector space  $A = \bigotimes_{i \in I \subseteq [n]} V_i$ .

**Theorem 8.21.** *The null cone of  $\mathrm{GL}_{\mathbf{d}} \curvearrowright \mathrm{End}(V)$  are the orbits of matrices  $M$ , such that for every subsystem  $A = \bigotimes_{i \in I \subseteq [n]} V_i$ ,  $\mathrm{Tr}_A(M)$  is permutation conjugate to a strictly upper triangular matrix.*

*Proof.* For  $I \subseteq [n]$  let  $A$  and  $B$  be a bipartition. We may assume the order of the tensor product is such that  $A$  is the tensor product of the first  $|I|$  vector spaces.

Suppose  $E_{ij}$  is a basis vector in  $V$  and  $E_{ij} = E_{i_1 j_1}^1 \otimes \cdots \otimes E_{i_{|I|} j_{|I|}}^{|I|} \otimes E_{i_{|I|+1} j_{|I|+1}}^{|I|+1} \otimes \cdots \otimes E_{i_n j_n}^n$ . Denote the tensor factors  $E_{ij}^A := E_{i_1 j_1}^1 \otimes \cdots \otimes E_{i_{|I|} j_{|I|}}^{|I|}$  and  $E_{ij}^B := E_{i_{|I|+1} j_{|I|+1}}^{|I|+1} \otimes \cdots \otimes E_{i_n j_n}^n$ .

Let  $\lambda(t) = \bigotimes_{i=1}^n \lambda_i(t)$  be a diagonal cocharacter, and define  $\lambda(t)^A = \bigotimes_{i \in I} \lambda_i(t)$ . Then  $E_{ij}^A$  is one of the idempotent basis vectors in  $A$  if and only if  $\lambda^A(t) \cdot E_{ij}^A = E_{ij}^A$  for every choice of  $\lambda$ . Two matrices  $\lambda(t) E_{ij} \lambda(t)^{-1}$  and  $\lambda(t) E_{i'j'} \lambda(t)^{-1}$  have the same power of  $t$  in their unique nonzero entry for every choice of  $\lambda$  if and only if there exists a subsystem  $A$  such that  $E_{ij}^A$  and  $E_{i'j'}^A$  are both idempotent and  $E_{ij}^B = E_{i'j'}^B$ .

So we see that a subgraph  $c$  is in  $C(M)$  if and only if it induces a cycle in the associated quiver of  $\mathrm{Tr}_A(M)$  for some subsystem  $A$ . So for  $M$  to be a viable matrix in the null cone, it must be that for every subsystem  $A$ ,  $\mathrm{Tr}_A(M)$  must have an associated quiver such that, for every cycle, its weight is zero. This implies that the associated quiver is acyclic as a directed graph, proving the claim.  $\square$

**Corollary 8.22.** *The null cone of  $\mathrm{GL}_{\mathbf{d}} \curvearrowright \mathrm{End}(V)^{\oplus m}$  are those tuples  $(M_1, \dots, M_m)$  such for every subsystem  $A$ ,  $(\mathrm{Tr}_A(M_1), \dots, \mathrm{Tr}_A(M_m))$  is in the null cone of  $\mathrm{End}(B)^{\oplus m}$  acted upon by  $\mathrm{GL}(B)$ , where  $B$  is the complementary subsystem of  $A$ .*

*Proof.* Each  $M_i$  must be in the null cone, so it is conjugate to a matrix  $N_i$  such that for every subsystem  $A$ ,  $\mathrm{Tr}_A(N_i)$  is permutation conjugate to a strictly upper triangular matrix. Furthermore, it must be the case that all  $\mathrm{Tr}_A(M_i)$  are simultaneously upper triangularizable for every subsystem  $A$ .  $\square$

If one can find a set of homogeneous polynomials that set-theoretically define the null cone, one can get a degree bound on the ring of invariants. Let  $\sigma$  be the largest degree of a set of polynomials defining the null cone. Then there exists an hsop where every polynomial is homogeneous of degree  $\text{lcm}\{1, \dots, \sigma\}$  (cf. [27]).

**Theorem 8.23** ([59, 87]). *For a reductive group  $G \curvearrowright V$ , and  $\delta$  is the largest degree of a hsop, then  $\beta(V) \leq \dim(V)\delta$ . In particular, if  $\sigma$  is the largest degree of a set of polynomials defining the null cone, then  $\beta(V) \leq \dim(V)\text{lcm}\{1, \dots, \sigma\}$ .*

### 8.3 The Relations Among the Invariants

For the classical action of  $\text{GL}(V) \curvearrowright \text{End}(V)^{\oplus m}$ , the relations among invariants of  $k[\text{End}(V)^{\oplus m}]^{\text{GL}(V)}$  are all a consequence of the Cayley-Hamilton Theorem [89]. Every invariant of degree  $d$  is the restitution of a multilinear invariant of  $\text{End}(V)^{\oplus d}$ . We denote the set of degree  $d$  multilinear invariants by  $(\text{End}(V)^{\otimes d})^{\text{GL}(V)}$ . Recall that there is a homomorphism  $\varphi^{\text{GL}(V)} : k[\mathcal{S}_d] \cong (\text{End}(V)^{\otimes d})^{\text{GL}(V)}$ ; it is the special case of the map defined in Equation 7.5 where  $n = 1$ . The relations among multihomogeneous invariants come from relations among multilinear invariants and vice versa.

**Theorem 8.24** ([89, 91]). *Given the map*

$$\varphi^{\text{GL}(V)} : \bigoplus_{m=1}^{\infty} k[\mathcal{S}_m] \rightarrow \bigoplus_{m=1}^{\infty} (\text{End}(V)^{\otimes m})^{\text{GL}(V)},$$

*the kernel is generated by the completely skew-symmetric Young symmetrizers acting on Young diagrams on at least  $\dim(V) + 1$  boxes. That is, the kernel is generated by  $\sum_{\sigma \in \mathcal{S}_m} \text{sgn}(\sigma)\sigma$  for  $m > \dim(V)$ . The image of these Young symmetrizers are the Cayley-Hamilton relations.*

Note that the kernel of map  $\varphi^{\text{GL}(V)}$  is graded by degree as well. We let  $\varphi_m^{\text{GL}(V)} : k[\mathcal{S}_m] \rightarrow (\text{End}(V)^{\otimes m})^{\text{GL}(V)}$  be the induced map on the  $m^{\text{th}}$  graded piece. Considering  $\text{GL}_{\mathbf{d}} \curvearrowright \bigotimes_{i=1}^n \text{End}(V_i)$ , we now look at the homomorphism of graded rings

$$\phi : \bigoplus_{m=1}^{\infty} \bigotimes_{i=1}^n k[\mathcal{S}_m] \rightarrow \bigoplus_{m=1}^{\infty} \bigotimes_{i=1}^n (\text{End}(V)^{\otimes m})^{\text{GL}(V_i)}. \quad (8.1)$$

We note that for a particular graded piece, the induced map  $\phi_m : \bigotimes_{i=1}^n k[\mathcal{S}_m] \rightarrow \bigotimes_{i=1}^n (\text{End}(V_i)^{\otimes m})^{\text{GL}(V_i)}$  is equal to  $\bigotimes_{i=1}^n \varphi_m^{\text{GL}(V_i)}$ .

Given linear maps  $\psi_1, \dots, \psi_n, \psi_i : V_i \rightarrow W_i$ , it is a well known fact from linear algebra that the kernel of  $\psi_1 \otimes \dots \otimes \psi_n$  is

$$\left( \ker(\psi_1) \otimes V_2 \otimes \dots \otimes V_n \right) \oplus \left( V_1 \otimes \ker(\psi_2) \otimes \dots \otimes V_n \right) \oplus \dots \quad (8.2)$$

However, consider the subgroup  $\mathcal{S}_m \subset \mathcal{S}_m^n$  given by the inclusion  $\iota : \sigma \mapsto (\sigma, \dots, \sigma)$ . For  $m > \dim(V)$ , the map  $\phi$  takes the element  $\sum_{\sigma \in \mathcal{S}_m} \text{sgn}(\sigma) \iota(\sigma)$  to 0. These give the relations among the classical invariants of  $\text{GL}(V) \curvearrowright \text{End}(V)^{\oplus m}$ .

**Definition 8.25.** A subgroup  $H \subseteq \mathcal{S}_m^n$ , which is isomorphic to  $\mathcal{S}_m^k$ , is said to have *standard action* on  $V^{\otimes m}$  if there is decomposition  $V = \bigotimes_{i=1}^k V_i$  such that the induced action of  $H$  on  $V^{\otimes m} \cong V_1^{\otimes m} \otimes \dots \otimes V_k^{\otimes m}$  is

$$\sigma = (\sigma_1, \dots, \sigma_k). \left( \bigotimes_{i,j} v_{ij} \right) = \bigotimes_{i,j} v_{i\sigma_i(j)},$$

where  $\bigotimes_{j=1}^m v_{ij} \in V_i^{\otimes m}$ . We say that  $H$  has a *semi-standard action* if it has a standard action or there is a decomposition  $V = \bigotimes_{i=1}^{k+1} V_i$  such that  $H$  acts by a standard action on  $\bigotimes_{i=1}^k V_i$  and trivially on  $V_{k+1}$ .

**Observation 8.26.** Every subgroup  $H \subseteq \mathcal{S}_m^n$  with a semi-standard action is one where, for each of the natural projection maps  $p_i$ ,  $1 \leq i \leq n$ ,  $p_i(H)$  is either equal to  $\mathcal{S}_m$  or  $\{0\}$ .

For every  $H \subseteq \mathcal{S}_m^n$  with a semi-standard action,  $H \cong \mathcal{S}_m^{k-1}$  or  $\mathcal{S}_m^k$  (depending if the action is standard or not), the map  $\phi_m$  restricts to a map

$$\bigotimes_{i=1}^n k[\mathcal{S}_m^k] \rightarrow \bigotimes_{i=1}^k (\text{End}(V_i)^{\otimes m})^{\text{GL}(V_i)}$$

where  $V = \bigotimes_{i=1}^k V_i$  is the decomposition associated to the action of  $H$ . Furthermore, this map is exactly of the form given in Equation 7.5. Therefore, we have that this restricted map is equal to a product

$$\phi_m^H := \psi_1^H \otimes \dots \otimes \psi_k^H. \quad (8.3)$$

We see that each map  $\psi_i^H$  is a map  $k[\mathcal{S}_m] \rightarrow (\text{End}(V_i)^{\text{GL}(V_i)})$  of the form as in Equation 7.5, with the exception of  $\psi_k^H$  if the action is properly semi-standard. In that case, it is a map of the same kind but precomposed with the map  $k[\mathcal{S}_m] \rightarrow k[\mathcal{S}_m]$  induced by the group homomorphism  $\mathcal{S}_m \rightarrow \{0\}$ . Once again, we know that the kernel looks like the decomposition in Equation 8.3.

**Theorem 8.27.** *Given  $V = \bigotimes_{i=1}^n V_i$ , every relation among the invariants of  $k[\text{End}(V)^{\oplus m}]^{\text{GL}_d}$  is in the kernel of  $\psi_i^H$  for some  $H \subseteq \mathcal{S}_m^n$  with a semi-standard action.*

*Proof.* We know that the kernel is graded by degree, so we restrict ourselves to the  $m^{\text{th}}$  graded piece of the ring  $k[\text{End}(V)]^{\text{GL}_d}$ . We know that every relation comes from a relation on  $(\text{End}(V)^{\otimes m})^{\text{GL}_d} = \bigotimes_{i=1}^n (\text{End}(V_i)^{\otimes m})^{\text{GL}(V_i)}$ . Given a relation, it is of the form  $\bigotimes_{i=1}^r \zeta_i$  where some of the  $\zeta_i = 0$ . We can also assume that each such  $\zeta_i = 0$  cannot be written as a product  $\bigotimes_j \zeta_{ij}$  where some of the  $\zeta_{ij}$  are 0. If it could, we would readjust the original factorization  $\bigotimes_{i=1}^r \zeta_i$ .

Let  $\zeta_{\ell_1}, \dots, \zeta_{\ell_k} = 0$  be the factors that are relations. We know that  $\zeta_{\ell_j}$  lies in some subsystem  $A_j = \bigotimes_{i \in I_j \subseteq [n]} V_i$ . Furthermore, the relation  $\zeta_{\ell_j}$  comes from the image of a the Young symmetrizer  $\sum_{\sigma \in \mathcal{S}_m} \text{sgn}(\sigma) \sigma$  given how the map  $\phi_m$  is defined. Then consider the decomposition  $V \cong \bigotimes_{i=1}^k A_i \otimes W$  where  $W = \bigotimes_{i \notin \bigcup I_j} V_i$ .

We see that there is a corresponding subgroup  $H \subseteq \mathcal{S}_m^n$  that is isomorphic to  $\mathcal{S}_m^k$  with an inclusion  $\iota$ : given  $\sigma = (\sigma_1, \dots, \sigma_k) \in H$ ,  $p_i(\sigma) = \sigma_j$  for  $i \in I_j$  and  $p_i(\sigma) = \text{id}$  for  $i \notin \bigcup I_j$ . Then we see that  $\phi_m$  restricts to a map equal  $\psi_1^H \otimes \dots \otimes \psi_k^H \otimes \psi_{k+1}^H$  where  $\psi_j^H$  is the map  $k[\mathcal{S}_m] \rightarrow (k[\text{End}(A_j)]^{\otimes m})^{\text{GL}(A_j)}$  and  $\psi_{k+1}^H$  is the map  $k[\mathcal{S}_m] \rightarrow (k[\text{End}(W)]^{\otimes m})^W$ , where every basis vector of  $k[\mathcal{S}_m]$  is first mapped to the basis vector associated with the identity and then composed with map  $\varphi^{\text{GL}(W)}$ . Then the kernel of  $\psi_1^H \otimes \dots \otimes \psi_k^H \otimes \psi_{k+1}^H$  is of the form in Equation 8.3 and this proves the result.  $\square$

Given that the degree bound in Corollary 8.12 is so large as to be computationally infeasible for moderate examples, it is desirable to better understand the relations among the invariants in the hope of lowering the bound.

One may also wonder if, by restricting to tuples of density operators, some relations are introduced among the  $\text{Tr}_\sigma^M$  that may decrease the number of polynomials that need to be checked for equivalence. We denote the set of  $m$ -tuples of density operators inside of  $\text{End}(V)^{\oplus m}$  by  $\mathcal{D}^{\oplus m}$ .

**Proposition 8.28** ([81]). *Let  $G$  act on a subvariety  $X \subseteq V$ . If  $G$  is reductive, and its ideal,  $I \subseteq k[V]$ , is a  $G$ -stable ideal, then  $k[V]^G / (I \cap k[V]^G) \cong (k[V]/I)^G$ .*

**Lemma 8.29.** *The Zariski closure of  $\mathcal{D}^{\oplus m}$  is the set  $T = \{(A_1, \dots, A_m) \mid \text{Tr}(A_i) = 1\}$ .*

*Proof.* First we work with the case  $m = 1$ . The result will follow from the fact that the product of Zariski dense sets is Zariski dense. Note that



$\text{Conv}(\mathcal{P}) = \mathcal{D} \subset \text{aff}(\mathcal{P})$  (the affine hull of  $\mathcal{P}$ ). Now suppose that  $\overline{\text{Conv}(\mathcal{P})} \subsetneq \overline{\text{aff}(\mathcal{P})}$ . Then  $\text{Conv}(\mathcal{P})$  must sit in a hypersurface of  $\text{aff}(\mathcal{P})$  and thus have dimension less than that of  $\text{aff}(\mathcal{P})$ . But  $\text{Conv}(\mathcal{P})$  contains a full dimensional simplex in  $\text{aff}(\mathcal{P})$ , so it cannot lie in a hypersurface. Thus  $\overline{\text{Conv}(\mathcal{P})} = \overline{\text{aff}(\mathcal{P})}$ . Now note that  $\text{aff}(\mathcal{P}) = \mathcal{H} \cap T$ , where  $\mathcal{H}$  is the set of Hermitian matrices. However,  $\mathcal{H}$  is Zariski dense in  $\text{End}(V)$  ([28]). Thus  $\overline{\text{aff}(\mathcal{P})} = T$ , noting that  $T$  is Zariski closed.  $\square$

**Theorem 8.30.**  $\mathbb{C}[\mathcal{D}^{\oplus m}]^{\text{GL}_d} \cong \mathbb{C}[\text{End}(V)^{\oplus m}]^{\text{GL}_d}/I$ , where  $I \subset \mathbb{C}[\text{End}(V)^{\oplus m}]^{\text{GL}_d}$  is generated by the polynomials  $\text{Tr}(M_i) - 1$ .

*Proof.* The invariant ring of  $\mathcal{D}^{\oplus m}$  is equal to the invariant ring of its Zariski closure, which is defined by the ideal  $I\mathbb{C}[\text{End}(V)^{\oplus m}]$  by Lemma 8.29. By Proposition 8.28,  $\mathbb{C}[\mathcal{D}^{\oplus m}]^{\text{GL}_d} \cong \mathbb{C}[\text{End}(V)^{\oplus m}]^{\text{GL}_d}/I$ .  $\square$

## Chapter 9

# Applications to the Study of Quantum Entanglement

In this chapter, we outline how the results in Chapters 7 and 8 can potentially be used to distinguish whether or not two quantum states have the same entanglement in the laboratory setting. We also present an example where we determine, using invariants, that two density operators are in the same local unitary orbit.

Given two quantum states with density operators  $\Psi_1$  and  $\Psi_2$ , to determine if they have the same entanglement, then by Corollary 8.10, one only needs to see if  $\text{Tr}_\sigma^M(\Psi_1) = \text{Tr}_\sigma^M(\Psi_2)$  for all  $\text{Tr}_\sigma^M$  up to the degree bound given in Corollary 8.12.

However, given two quantum states in the laboratory, determining the density operators  $\Psi_1$  and  $\Psi_2$  is not necessarily feasible. This shortcoming also appears when trying the approach of finding normal forms for each  $U_{\mathbf{d}}$  orbit. Necessary and sufficient conditions for  $U_{\mathbf{d}}$ -equivalence using a normal form for each  $U_{\mathbf{d}}$ -orbit were worked out in several papers [61, 108, 107, 69, 68].

Nevertheless, computing the values of invariant polynomials for a density operator may not require such knowledge. For example, consider the Rényi Entropies, defined previously in Equation 3.3. Given a *bipartition*  $A:B$  of  $V$ , where  $A$  and  $B$  are complementary subsystems, and a density operator  $\rho$ , we then note the following equality.

$$\text{Tr}(\text{Tr}_A(\rho)^q) = \exp((1-q)H_q^{AB}(\rho))$$

which is a polynomial for  $q$  a natural number. The Rényi entropies [92, 6, 9, 10, 34] are a well-studied measurement of entanglement. Positive in-

tegral ( $q \in \mathbb{Z}_{\geq 1}$ ) Rényi entropies can be measured experimentally without computing the density operators explicitly [20, 1, 24, 95, 86]. This makes them especially attractive as invariants used to separate  $U_{\mathbf{d}}$ -orbits of density operators. However, they are not necessarily a complete set of invariants.

This suggests that it may be possible to compute  $\text{Tr}_{\sigma}^M(\Psi_1)$  without computing  $\Psi_1$ . This would mean that the invariant polynomials can be expressed as a series of measurements that can be carried out on a quantum state in the laboratory. However, whether or not this is true is still unresolved.

## 9.1 Example

We now consider the case of two qubit quantum states, that is to say, states with density operators in  $\text{End}((\mathbb{C}^2)^{\otimes 2})$ . Suppose we are given the following two density operators:

$$\Psi_1 = -\frac{1}{8} \begin{pmatrix} -6 & -4+2i & 18 & 12-6i \\ -4-2i & 2 & 12+6i & -6 \\ 18 & 12-6i & -6 & -4+2i \\ 12+6i & -6 & -4-2i & 2 \end{pmatrix}$$

$$\Psi_2 = -\frac{1}{8} \begin{pmatrix} -8 & 0 & 24i & 0 \\ 0 & 4 & 0 & -12i \\ -24i & 0 & -8 & 0 \\ 0 & 12i & 0 & 4 \end{pmatrix}$$

Then we note that both of these matrices are in the image of the Segre embedding  $\text{End}(\mathbb{C}^2) \times \text{End}(\mathbb{C}^2)$  inside of  $\text{End}(\mathbb{C}^4)$  and so can be factored in the following way:

$$\Psi_1 = -\frac{1}{8} \begin{pmatrix} -1 & 3 \\ 3 & -1 \end{pmatrix} \otimes \begin{pmatrix} 6 & 4-2i \\ 4+2i & -2 \end{pmatrix},$$

$$\Psi_2 = -\frac{1}{8} \begin{pmatrix} -2 & 6i \\ -6i & -2 \end{pmatrix} \otimes \begin{pmatrix} 4 & 0 \\ 0 & -2 \end{pmatrix}.$$

We calculated a complete generating set for  $k[\text{End}(\mathbb{C}^4)^2]^{\text{GL}_{2,2}}$ ,  $\text{GL}_{2,2} := \text{GL}(\mathbb{C}^2) \times \text{GL}(\mathbb{C}^2)$  in Example 7.27. Here we only need those invariants that are also invariants of  $k[\text{End}(\mathbb{C}^4)]^{\text{GL}_{2,2}}$ . We simply need to check whether these two density operators agree on the resulting 4 invariants. If we write  $\Psi_1 = -\frac{1}{8}A_1 \otimes B_1$  and  $\Psi_2 = -\frac{1}{8}A_2 \otimes B_2$ , we first note that

$$\begin{aligned}
\text{Tr}(A_1) &= -2 & \text{Tr}(A_1^2) &= 20 \\
\text{Tr}(B_1) &= 4 & \text{Tr}(B_1^2) &= 80 \\
\text{Tr}(A_2) &= -4 & \text{Tr}(A_2^2) &= 80 \\
\text{Tr}(B_2) &= 2 & \text{Tr}(B_2^2) &= 20
\end{aligned}$$

Then computing the values of the 4 invariants, we get

$$\begin{aligned}
-\frac{1}{8}\text{Tr}(A_1)\text{Tr}(B_1) &= 1 = -\frac{1}{8}\text{Tr}(A_2)\text{Tr}(B_2) \\
\frac{1}{64}\text{Tr}(A_1^2)\text{Tr}(B_1^2) &= 25 = \frac{1}{64}\text{Tr}(A_2^2)\text{Tr}(B_2^2) \\
\frac{1}{64}\text{Tr}(A_1)^2\text{Tr}(B_1^2) &= 5 = \frac{1}{64}\text{Tr}(A_2)^2\text{Tr}(B_2^2) \\
\frac{1}{64}\text{Tr}(A_1^2)\text{Tr}(B_1)^2 &= 5 = \frac{1}{64}\text{Tr}(A_2^2)\text{Tr}(B_2)^2
\end{aligned}$$

Then according to Corollary 8.10, we know that these two matrices must be in the same local unitary orbit. Indeed, taking the matrix

$$g = \begin{pmatrix} -1 & 0 \\ 0 & -i \end{pmatrix} \otimes \begin{pmatrix} \frac{2-i}{\sqrt{6}} & \frac{-2+i}{\sqrt{30}} \\ \frac{1}{\sqrt{6}} & \frac{5}{\sqrt{30}} \end{pmatrix}$$

we get that  $g\Psi_2g^{-1} = \Psi_1$ . Thus these two density operators are in the same  $U_{2,2}$  orbit. We were able to reduce the number of invariants that needed to be tested down to four by restricting our example to matrices in the image of the Segre embedding of  $\text{End}(\mathbb{C}^2)^2$  inside of  $\text{End}(\mathbb{C}^4)$ .

Note that these invariants are the polynomial versions of Rényi Entropies, as noted in Equation 3.3. In general, for those density operators in the image of the Segre embedding of  $\times_{i=1}^n \text{End}(V_i)$  inside of  $\text{End}(\otimes_{i=1}^n V_i)$ , the  $U_d$  orbits can be separated by the Rényi Entropies. Thus they can be separated in the laboratory setting. However, the states represented by such density operators correspond to the so-called *separable states*, which are not entangled.

In general, many more invariants need to be computed to determine if two matrices are in the same  $U_d$ -orbit. However, these invariants give a concrete computational method for solving this problem.

# Bibliography

- [1] D. Abanin and E. Demler. Measuring entanglement entropy of a generic many-body system with a quantum switch. *Physical review letters*, 109(2):020504, 2012.
- [2] E. Allman and J. Rhodes. Phylogenetic ideals and varieties for the general markov model. *Advances in Applied Mathematics*, 40(2):127–148, 2008.
- [3] E. Artin. Zur theorie der hyperkomplexen zahlen. In *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, volume 5, pages 251–260. Springer, 1927.
- [4] F. Bacchus, S. Dalmao, and T. Pitassi. Algorithms and complexity results for # SAT and Bayesian inference. In *Foundations of Computer Science, 2003. Proceedings. 44th Annual IEEE Symposium on*, pages 340–351. IEEE, 2003.
- [5] J. Baez. Spin network states in gauge theory. *arXiv preprint gr-qc/9411007*, 1994.
- [6] J. Baez. Renyi entropy and free energy. *arXiv preprint arXiv:1102.2098*, 2011.
- [7] D. Bates and L. Oeding. Toward a salmon conjecture. *Experimental Mathematics*, 20(3):358–370, 2011.
- [8] V. Bergholm and J. Biamonte. Categorical quantum circuits. *Journal of Physics A: Math. Theor.*, 44:245304, 2011.
- [9] J. Biamonte, V. Bergholm, and M. Lanzagorta. Tensor network methods for invariant theory. *To appear in J. Phys. A Math Gen.*, 2013.
- [10] J. Biamonte, J. Morton, and J. Turner. Tensor network contractions for # SAT. *arXiv preprint arXiv:1405.7375*, 2014.

- [11] J. Biamonte, S. Clark, and D. Jaksch. Categorical tensor network states. *AIP Advances*, 1(4):042172–042172, 2011.
- [12] J. Bonin, A. De Mier, and M. Noy. Lattice path matroids: enumerative aspects and Tutte polynomials. *Journal of Combinatorial Theory, Series A*, 104(1):63–94, 2003.
- [13] J. Bonin and O. Gimenez. Multi-path matroids. *Combinatorics, Probability and Computing*, 16(02):193–217, 2007.
- [14] M. Brion. Representations of quivers. 2008.
- [15] M. Brion. Introduction to actions of algebraic groups. *Les cours du CIRM*, 1(1):1–22, 2010.
- [16] A. Bulatov and M. Grohe. The complexity of partition functions. *Theoretical Computer Science*, 348(2):148–186, 2005.
- [17] J.Y. Cai, X. Chen, and P. Lu. Non-negatively weighted  $\#$  CSP: An effective complexity dichotomy. In *Computational Complexity (CCC), 2011 IEEE 26th Annual Conference on*, pages 45–54. IEEE, 2011.
- [18] J.Y. Cai, P. Lu, and M. Xia. Holographic algorithms with matchgates capture precisely tractable planar- $\#$  CSP. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*, pages 427–436. IEEE, 2010.
- [19] 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.
- [20] J. Cardy. Measuring entanglement using quantum quenches. *Phys. Rev. Lett.*, 106:150404, Apr 2011.
- [21] C. Chamon and E.R. Mucciolo. Rényi entropies as a measure of the complexity of counting problems. *Journal of Statistical Mechanics: Theory and Experiment*, 4:8, April 2013.
- [22] F.R.K. Chung and R.P. Langlands. A combinatorial Laplacian with vertex weights. *J. Combin. Theory Ser. A*, 75(2):316–327, 1996.
- [23] P. Cvitanovic. *Group theory*. Nordita, 1984.

- [24] A.J. Daley, H. Pichler, J. Schachenmayer, and P. Zoller. Measuring entanglement growth in quench dynamics of bosons in an optical lattice. *Physical review letters*, 109(2):020505, 2012.
- [25] C. Damm, M. Holzer, and P. McKenzie. The complexity of tensor calculus. *Computational Complexity*, 11(1):54—89, 2003.
- [26] H. Derksen. Polynomial bounds for rings of invariants. *Proceedings of the American Mathematical Society*, 129(4):955–963, 2001.
- [27] H. Derksen and H. Kraft. Constructive invariant theory, 1997.
- [28] M. Domokos. Hermitian matrices with a bounded number of eigenvalues. *arXiv preprint arXiv:1301.7215*, 2013.
- [29] J. Draisma and J. Kuttler. On the ideals of equivariant tree models. *Mathematische Annalen*, 344(3):619–644, 2009.
- [30] O. Dubois. On the r,s-SAT satisfiability problem and a conjecture of Tovey. *Discrete Applied Mathematics*, 26(1):51–60, 1990.
- [31] W. Dür, G. Vidal, and J.I. Cirac. Three qubits can be entangled in two inequivalent ways. *Phys. Rev. A*, 62:062314, Nov 2000.
- [32] M. Dyer and D. Richerby. An effective dichotomy for the counting constraint satisfaction problem. *arXiv preprint arXiv:1003.3879*, 2010.
- [33] J. Eisert. Entanglement and tensor network states. *ArXiv e-prints*, August 2013.
- [34] J. Eisert, M. Cramer, and M.B. Plenio. Colloquium: Area laws for the entanglement entropy. *Reviews of Modern Physics*, 82(1):277, 2010.
- [35] G. Evenbly and G. Vidal. Tensor network states and geometry. *Journal of Statistical Physics*, 145:891–918, November 2011.
- [36] E. Fischer, J.A. Makowsky, and E.V. Ravve. Counting truth assignments of formulas of bounded tree-width or clique-width. *Discrete Applied Mathematics*, 156(4):511–529, 2008.
- [37] S. Friedland and E. Gross. A proof of the set-theoretic version of the salmon conjecture. *Journal of Algebra*, 356(1):374–379, 2012.
- [38] G. Gour and N.R. Wallach. Classification of multipartite entanglement of all finite dimensionality. *Physical review letters*, 111(6):060502, 2013.

- [39] M. Grassl, M. Rötteler, and T. Beth. Computing local invariants of quantum-bit systems. *Physical Review A*, 58(3):1833, 1998.
- [40] R.B. Griffiths. Channel kets, entangled states, and the location of quantum information. *PRA*, 71(4):042337, April 2005.
- [41] R.B. Griffiths, S. Wu, L. Yu, and S.M. Cohen. Atemporal diagrams for quantum circuits. *PRA*, 73(5):052309, May 2006.
- [42] D. Gross and J. Eisert. Novel Schemes for Measurement-Based Quantum Computation. *Physical Review Letters*, 98(22):220503, June 2007.
- [43] R. Halin. S-functions for graphs. *Journal of Geometry*, 8(1-2):171–186, 1976.
- [44] E.C. Tan, H. Aslaksen, and C. Zhu. Generators and relations of invariants of  $2 \times 2$  matrices. *Communications in Algebra*, 22:1821–1832, 1994.
- [45] M.W. Hero and J.F. Willenbring. Stable hilbert series as related to the measurement of quantum entanglement. *Discrete Mathematics*, 309(23):6508–6514, 2009.
- [46] D. Hilbert. Über die vollen Invariantensysteme. *Math. Ann.*, 42:313–373, 1893.
- [47] D. Hilbert. Über die theorie der algebraischen formen. *Mathematische Annalen*, 36(4):473–534, 1890.
- [48] K. Hiss. *Constructive invariant theory for reductive algebraic groups*. PhD thesis, Thesis Brandeis University, Waltham, 1996.
- [49] M. Hochster and J.L. Roberts. Rings of invariants of reductive groups acting on regular rings are Cohen-Macaulay. *Advances in mathematics*, 13(2):115–175, 1974.
- [50] R.A. Horn and C.R. Johnson. *Matrix analysis*. Cambridge University Press, 1990.
- [51] M. Johansson, M. Ericsson, K. Singh, E. Sjöqvist, and M.S. Williamson. Topological phases and multiqubit entanglement. *Physical Review A*, 85(3):032112, 2012.



- [52] T.H. Johnson, J.D. Biamonte, S.R. Clark, and D. Jaksch. Solving search problems by strongly simulating quantum circuits. *Scientific Reports*, 3, February 2013.
- [53] T.H. Johnson, S.R. Clark, and D. Jaksch. Dynamical simulations of classical stochastic systems using matrix product states. *Phys. Rev. E*, 82(3):036702, September 2010.
- [54] A. Joyal and R. Street. The geometry of tensor calculus II. *Unpublished draft, available from Ross Street’s website*.
- [55] A. Joyal and R. Street. The geometry of tensor calculus. I. *Advances in Mathematics*, 88(1):55–112, 1991.
- [56] A. Joyal and R. Street. Braided tensor categories. *Advances in Mathematics*, 102(1):20–78, 1993.
- [57] G. Kemper. Separating invariants. *Journal of Symbolic Computation*, 44(9):1212–1222, 2009.
- [58] G. Kempf. Instability in invariant theory. *Annals of Mathematics*, pages 299–316, 1978.
- [59] F. Knop. Der kanonische modul eines invariantenrings. *Journal of Algebra*, 127(1):40–54, 1989.
- [60] H. Kraft and C. Procesi. Classical invariant theory, a primer. *Lecture Notes, Version*, 2000.
- [61] B. Kraus. Local unitary equivalence of multipartite pure states. *Physical review letters*, 104(2):020504, 2010.
- [62] Y. Lafont. Towards an algebraic theory of Boolean circuits. *Journal of Pure and Applied Algebra*, 184(2):257–310, 2003.
- [63] J.M. Landsberg, J. Morton, and S. Norine. Holographic algorithms without matchgates. *Linear Algebra and its Applications*, 438(15), 2013.
- [64] J.M. Landsberg. *Tensors:: Geometry and Applications*, volume 128. AMS Bookstore, 2012.
- [65] S.L. Lauritzen and D.J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems.

*Journal of the Royal Statistical Society. Series B (Methodological)*, pages 157–224, 1988.

- [66] L. Le Bruyn and C. Procesi. Semisimple representations of quivers. *Transactions of the American Mathematical Society*, pages 585–598, 1990.
- [67] U. Leron. Trace identities and polynomial identities of  $n \times n$  matrices. *J. Algebra*, 42:369–377, 1976.
- [68] J.L. Li and C.F. Qiao. Classification of arbitrary multipartite entangled states under local unitary equivalence. *Journal of Physics A: Mathematical and Theoretical*, 46(7):075301, 2013.
- [69] B. Liu, J.L. Li, X. Li, and C.F. Qiao. Local unitary classification of arbitrary dimensional multipartite pure states. *Physical review letters*, 108(5):050501, 2012.
- [70] S. Lloyd et al. Universal quantum simulators. *Science-New York*, pages 1073–1077, 1996.
- [71] J.-G. Luque and J.-Y. Thibon. Polynomial invariants of four qubits. *Phys. Rev. A (3)*, 67(4):042303, 5, 2003.
- [72] J.-G. Luque and J.-Y. Thibon. Algebraic invariants of five qubits. *Journal of physics A: mathematical and general*, 39(2):371, 2006.
- [73] S. Mac Lane. *Categories for the working mathematician*. Springer verlag, 1998.
- [74] T. Maciążek, M. Oszmaniec, and A. Sawicki. How many invariant polynomials are needed to decide local unitary equivalence of qubit states? *arXiv preprint arXiv:1305.3894*, 2013.
- [75] I.L. Markov and Y. Shi. Simulating quantum computation by contracting tensor networks. *SIAM Journal on Computing*, 38(3):963–981, 2008.
- [76] H. Maschke. Ueber den arithmetischen charakter der coefficienten der substitutionen endlicher linearer substitutionsgruppen. *Mathematische Annalen*, 50(4):492–498, 1898.
- [77] H. Maschke. Beweis des satzes, dass diejenigen endlichen linearen substitutionsgruppen, in welchen einige durchgehends verschwindende

Koeffizienten auftreten, intransitiv sind. *Mathematische Annalen*, 52(2):363–368, 1899.

- [78] J. Morton. Pfaffian circuits. *arXiv preprint arXiv:1101.0129*, 2010.
- [79] J. Morton and J. Turner. Generalized counting constraint satisfaction problems with determinantal circuits. *Linear Algebra and its Applications*, 466:357–381, 2015.
- [80] D. Mumford, J. Fogarty, and F. Kirwan. *Geometric invariant theory*, volume 34 of *Ergebnisse der Mathematik und ihrer Grenzgebiete (2) [Results in Mathematics and Related Areas (2)]*. Springer-Verlag, Berlin, third edition, 1994.
- [81] M. Nagata. Invariants of group in an affine ring. *Kyoto Journal of Mathematics*, 3(3):369–378, 1963.
- [82] M.A. Nielsen and I.L. Chuang. *Quantum computation and quantum information*. Cambridge University Press, Cambridge, 2000.
- [83] N. Nishimura, P. Ragde, and S. Szeider. Solving # SAT using vertex covers. *Acta Informatica*, 44(7-8):509–523, 2007.
- [84] R. Penrose. Applications of negative dimensional tensors. *Combinatorial mathematics and its applications*, pages 221–244, 1971.
- [85] R. Penrose. *The road to reality: a complete guide to the laws of the universe*. Knopf, New York, NY, 2005.
- [86] H. Pichler, L. Bonnes, A.J. Daley, A.M. Läuchli, and P. Zoller. Thermal versus entanglement entropy: a measurement protocol for fermionic atoms with a quantum gas microscope. *New Journal of Physics*, 15(6):063003, 2013.
- [87] V.L. Popov. Constructive invariant-theory. *Astérisque*, (87-8):303–334, 1981.
- [88] V.L. Popov. The constructive theory of invariants. *Izvestiya: Mathematics*, 19(2):359–376, 1982.
- [89] C. Procesi. The invariant theory of  $n \times n$  matrices. *Advances in Mathematics*, 19(3):306–381, 1976.
- [90] R. Raussendorf and H.J. Briegel. A one-way quantum computer. *Physical Review Letters*, 86(22):5188, 2001.

- [91] J.P. Razmyslov. Trace identities of full matrix algebras over a field of characteristic zero. *Mathematics of the USSR-Izvestiya*, 8(4):727, 1974.
- [92] A. Rényi. On measures of entropy and information. In *Fourth Berkeley Symposium on Mathematical Statistics and Probability*, pages 547–561, 1961.
- [93] C. Rovelli and L. Smolin. Spin networks and quantum gravity. *Physical Review D*, 52(10):5743, 1995.
- [94] M. Samer and S. Szeider. A fixed-parameter algorithm for #SAT with parameter incidence treewidth. *arXiv preprint cs/0610174*, 2006.
- [95] J. Schachenmayer, B.P. Lanyon, C.F. Roos, and A.J. Daley. Entanglement growth in quench dynamics with variable range interactions. *Physical Review X*, 3(3):031015, 2013.
- [96] P. Selinger. A survey of graphical languages for monoidal categories. *New Structures for Physics*, pages 275–337, 2009.
- [97] C.A. Tovey. A simplified NP-complete satisfiability problem. *Discrete Applied Mathematics*, 8(1):85–89, 1984.
- [98] L. Valiant. Quantum circuits that can be simulated classically in polynomial time. *SIAM J. Comput.*, 31(4):1229–1254, 2002.
- [99] L. Valiant. Holographic algorithms (extended abstract). *Proceedings of the 45th annual Symposium on Foundations of Computer Science*, pages 306–315, 2004.
- [100] L. Valiant. The complexity of computing the permanent. *Theoret. Comput. Sci.*, 8(2):189–201, 1979.
- [101] L. Valiant. The complexity of enumeration and reliability problems. *SIAM Journal on Computing*, 8(3):410–421, 1979.
- [102] L. Valiant and V.V. Vazirani. NP is as easy as detecting unique solutions. *Theoretical Computer Science*, 47:85–93, 1986.
- [103] L. Valiant. Accidental algorithms. In *Proc. 47th Annual IEEE Symposium on Foundations of Computer Science*, pages 509–517. Citeseer, 2006.

- [104] F. Verstraete, V. Murg, and J.I. Cirac. Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems. *Advances in Physics*, 57:143–224, 2008.
- [105] J.H. Maclagan Wedderburn. On hypercomplex numbers. *Proceedings of the London Mathematical Society*, 2(1):77–118, 1908.
- [106] M. Xia, P. Zhang, and W. Zhao. Computational complexity of counting problems on 3-regular planar graphs. *Theoretical Computer Science*, 384(1):111–125, 2007.
- [107] T.-G. Zhang, M.-J. Zhao, M. Li, S.-M. Fei, and X. Li-Jost. Criterion of local unitary equivalence for multipartite states. *Phys. Rev. A*, 88:042304, Oct 2013.
- [108] T.-G. Zhang, M.-J. Zhao, X. Li-Jost, and S.-M. Fei. Local unitary invariants for multipartite states. *International Journal of Theoretical Physics*, 52(9):3020–3025, 2013.

## **Vita**

### **Jacob Turner**

The author was born on April 12, 1989 in Hazard, Kentucky. In 2006, he graduated valedictorian from Perry County Central High School in Hazard, Kentucky. In May 2010, the author graduated magna cum laude with honors from Western Kentucky University with a B.A. in mathematics. The author then pursued his graduate studies in invariant theory at the Pennsylvania State University. He completed his thesis under the advisement of Dr. Jason Morton in the Spring semester of 2015 and was awarded his PhD in Mathematics in May 2015.