Identifying Perfect Nonlocal Games

by

Adam Bene Watts

Submitted to the Department of Physics
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2021

Author .....
Department of Physics
August 6, 2021

Certified by .....
Aram W. Harrow
Associate Professor of Physics
Thesis Supervisor

Accepted by .....
Deepto Chakrabarty
Chairman, Department Committee on Graduate Theses

Identifying Perfect Nonlocal Games

by

Adam Bene Watts

Submitted to the Department of Physics
on August 6, 2021, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

This thesis is about nonlocal games. These "games" are really interactive tests in which a verifier checks the correlations that can be produced by non-communicating players. We study the class of commuting operator correlations: correlations which can by produced by players who make commuting measurements on some shared entangled state. This thesis contains following results:

- A general algebraic characterization of games with a "perfect" commuting operator strategy, i.e. games with a winning correlation that can be produced exactly by commuting operator measurements. This characterization is built on a key result in non-commutative algebraic geometry known as a (non-commutative) Nullstellensatz.
- A sufficient condition for a class of nonlocal games called XOR games to have a perfect commuting operator strategy. This condition can be checked in polynomial time, and can be understood either as non-existence of a combinatorial object called a PREF (the noPREF condition) or as non existence of a solution to an instance of the subgroup membership problem in a specially constructed group.
- A family of simple one-qubit-per-player strategies we call MERP strategies, which we show are optimal for any XOR game which has a perfect commuting operator strategy by the noPREF condition.
- Proofs that the noPREF condition is both necessary and sufficient for symmetric XOR games and 3 player XOR games.
- Explicit constructions of several families of XOR games with interesting properties.
- An analysis of randomly generated XOR games using the noPREF condition and the first moment method.

Thesis Supervisor: Aram W. Harrow
Title: Associate Professor of Physics

Acknowledgments

Any set of acknowledgements I write must begin with thanks to my parents, Veronica Bene and Paul Watts, and my sister Simona. My childhood home has always been a place of peace and beauty, and a constant source of stability in my life. It is, without a doubt, the wisdom of my parents and the care of my sister that has kept it that way. Beyond that, I need to thank my parents for nurturing my curiosity (some of my earliest childhood memories are learning basic mathematics on the blackboard my father made), and my sister for inspiring me with her own accomplishments.

Of course, my family is bigger than just my sister and parents. I must also thank my Aunt and Uncle Hanne and Martin Guiffrida who, among other things, are responsible for my first introduction to computers and the word of technology; my Aunt Marion Voysey for providing a much-needed organizational influence in our family, and providing me with most of the photos of myself that I've ever used professionally; my half brother Ian Watts for being both an older brother and a role model; and my cousin Galen Voysey for being one of my longest and truest friends.

Speaking of friends, I've been blessed with many. While it's impossible to list them all, Tyler Lawtey, Tristan and Liam Gaw, Levi Jackson, Sylvannah Densmore, Lina Wahlstrom, Michele Tonutti, Katie Linder, Maia Courtenay, and Danie Martin all deserve special mention as childhood friends without whose influence I wouldn't be the person I am today. More recently, my roommates Jane Panagaden and Galen Voysey (who gets his mention here) and later Chris McNally and David Berardo have all been constant friendship and support in my life.

Academically I need to thank my early physics teachers Mr. Tanner (whose first name I'm embarrassed to admit I've either forgotten or never knew) and Mark Wheen, along with the teachers at the International Summer School for Young Physicists (ISSYP) at Perimiter Institute for nurturing a chilhood interest in physics. Research-wise I've benefited from a long list of mentors, all of who have provided me with invaluable guidance: Doina Precup, Prakash Panangaden, Bruce Reed, Patrick Hayden, Andrew Childs, David Gosset, Sergey Norin, Peter Selinger, Aram Harrow, Robin Kothari, John Wright, Nicole Yunger Halpern,

and Bill Helton. Equal thanks are owed to my peers, both from the Honours Physics program at McGill and more recently my fellow graduate students at MIT, from whom I learned as much as I learned from any of my teachers. Again it is impossible to list them all, but Xuan Sun, Jane Panangaden, Luise Dziobek-Garrett, David Berardo, Chris McNally, Gurtej Kanwar, Anand Natarajan, Mehdi Soleimanifar, Linghang Kong, and Luke Schaeffer all deserve special mention.

The results in this thesis are closely based on several papers (including one still in preparation) and related to several others. The authors on those papers (many of whom have been mentioned already, but who I mention again now in their coauthor capacity) deserve special mention. Thanks to Gurtej Kanwar, Anand Natarajan, Aram Harrow, Robin Kothari, Luke Schaeffer, Avishay Tal, Nicole Yunger Halpern, Bill Helton, and Igor Klep. My thesis committee members Aram Harrow, Peter Shor, and David Kaiser also all deserve thanks for their patience and support as I threw together a pretty last minute thesis.

Penultimately, and somewhat unconventionally, I feel I owe thanks to the staff and owners at several coffee shops, where probably the majority of the work contained in this thesis was done. Café Plume in Montreal (now Paquebot), Curio Coffee in Boston, Java Jive in Salmon Arm (now Ecotreats), and Kaffeeklatsch in Calgary rank among my most frequented. I'll also take this opportunity to recommend these cafes to anyone who happens to be reading this thesis and looking for a office away from the office.

Finally, just as any acknowledgements section must begin with mention of my sister and parents, it must end with mention of my partner, Claudine Lebosquain. For the past nine years she has brought joy, excitement, drama, and love to my life in equal measure, and I wouldn't have it any other way.

Financial Acknowledgements: While performing the research in this thesis, I was financially supported by by NSF grant CCF-1729369.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Background

This thesis is about quantum correlations. The study of quantum correlations began with John Bell's pioneering paper [4], which showed that non-communicating observers measuring a quantum state could produce correlations which would be impossible to reproduce classically. This result implied the existence of a "Bell Test" – a test with purely classical input and output which could certify the presence of quantum behavior in a system. More abstractly, it showed that quantum systems can behave in ways which are completely impossible for classical systems to reproduce.

Recent results have shown that the set of correlations which can be produced by measurements like the one's described by Bell are incredibly rich. In particular, some correlations in the set require measurements of arbitrarily large, or even infinite dimensional systems to be produced. Exactly which correlations can be produced by measurements of infinite dimensional systems depends on assumptions about the underlying mathematical structure of the Hilbert space in which the quantum state lives. And it is in general undecidable whether a given correlation can be produced, or even approximated, by measurements of some quantum state.

This thesis develops techniques for deciding membership in the set of quantum correlations. More concretely, it is concerned with questions of the form:

> "Does there exist any quantum state and set of measurements that non-communicating observers can make which can produce a specified

correlation?"

Because of the aforementioned undecidability results, any techniques for answering this question will fail on some correlations. Yet there do exist large classes of correlations for which this question can be efficiently answered. This thesis focuses on these classes of correlations. By developing mathematical tools to identify these correlations, as well as the states and measurements that produce them, we can further our understanding of the operational power of quantum mechanics. In particular, computational techniques for finding quantum correlations and measurement strategies have the potential to both uncover sophisticated measurement strategies that have not yet been discovered, and to tell us when a class of measurement techniques that has already been developed is in a sense optimal. These measurement strategies may then find use throughout quantum information, in areas far removed from quantum games [5, 53, 67].

In the remainder of introduction we first give some background to the field of quantum correlations and nonlocal games, as well as a quick overview of some important mathematical concepts. Then, in Section 1.4 we give an overview of the results in this thesis.

1.1 Nonlocal Games and Quantum Correlations

We begin by describing an experiment involving a verifier (or referee) and two "players", canonically called Alice and Bob. During the experiment the verifier selects two questions i and j from a list of possible questions, then sends question i to Alice and j to Bob. After receiving their questions, *but without knowing the question sent to the other player,* Alice and Bob choose responses a and b and send those responses back to the verifier. This process is illustrated diagrammatically below:

An important note is that in this experiment we have assumed the randomness used by the verifier in selecting questions i and j is is independent of any randomness shared by Alice and Bob, so absolutely no information is leaked to Alice or Bob about the question the other player is sent. This assumption, known as the measurement-independence assumption, requires considerable effort to enforce experimentally [38, 52, 18, 41, 40, 29, 24] and is a necessary part of a formal Bell test [28, 23]. In this (purely theoretical) thesis, we will assume

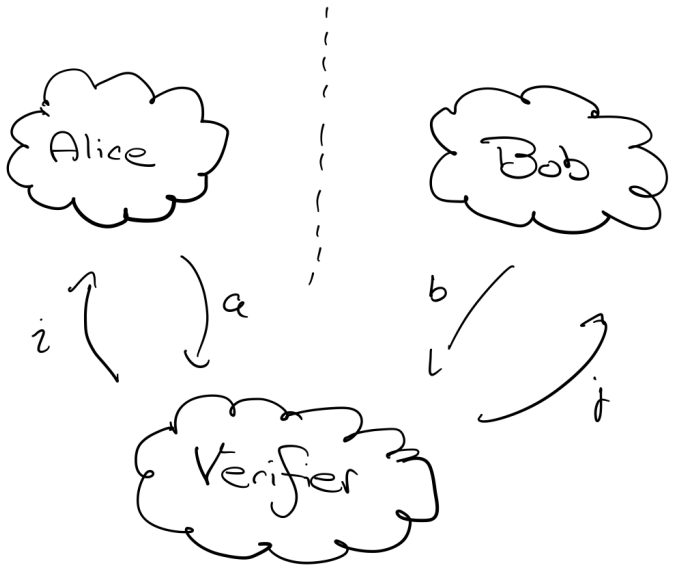

Figure 1-1: An experiment in which the verifier tests the correlations that can be produced by Alice and Bob.

it freely.

We can describe the likelihood of a certain outcome when performing this experiment via the conditional probability $p(a, b|i, j)$, which represents "the probability the verifier receives response a from Alice and b from Bob given that they sent question i to Alice and j to Bob". We let $\mathcal{I}_A, \mathcal{I}_B$ denote be the set of all the questions (inputs) that can be sent to Alice and Bob respectively, and $\mathcal{O}_A$, $\mathcal{O}_B$ be the sets containing all their possible responses. A correlation is a set of conditional probabilities

$$\{p(a, b|i, j) \mid a \in \mathcal{O}_A,\ b \in \mathcal{O}_B,\ i \in \mathcal{I}_A,\ j \in \mathcal{I}_B\} \tag{1.1.1}$$

which together specify the probability of Alice and Bob giving any possible response to any possible set of questions asked. Note that we can assume without loss of generality that the question and responses sent to and from Alice and Bob are integers, so sets $\mathcal{I}$ and $\mathcal{O}$ are specified completely by their size. Technically, whenever talking about a correlations we should specify the question size n and response size m, though we will often omit these details. Whenever a correlation is referred to without n and m specified it should be assumed

that both are finite constants.

We can view correlations as tuples in $\mathbb{R}^{|\mathcal{I}_A||\mathcal{I}_B||\mathcal{O}_A||\mathcal{O}_B|}$ and then define addition and scalar multiplication of correlations in the natural (entry-wise) way. This observation allows us to discuss notions such as closure and convexity of various sets of correlations.

Closely related to correlations are the strategies used by Alice and Bob to map the questions they receive from the verifier to the responses they send back. In our standard view of correlations, we think of Alice and Bob as knowing ahead of time a correlation which they are supposed to produce, and then trying to come up with a strategy for producing that correlation during the experiment with the verifier. The key observation motivating the study of quantum correlations is that that the correlations Alice and Bob can produce depend on the resources they are given. In the next section we discuss some of these possible resources, the descriptions of the associated strategies, and some preliminary results about the correlations they can produce.

1.1.1 Correlation Sets

In this subsection we introduce various sets of correlations, following the notation in [60].

Classical Correlations

We first consider the case where Alice and Bob only have access to classical resources. The simplest strategy they can pursue is a deterministic one, with Alice and Bob deciding to give a fixed response a_i, b_j to each possible question i and j. Such a strategy can be completely specified by listing the variables a_i, b_j for every possible question i and j, and produces correlations of the form

$$p(a, b|i, j) = \delta_{a,a_i}\delta_{b,b_j} \tag{1.1.2}$$

where δ is the standard Kronecker delta.

Classical Alice and Bob can also use randomness to produce correlations. In this setting we imagine Alice and Bob looking at some random classical event – for example a die roll – and then choosing their response to a question based on that. This randomness can be

private, meaning Alice and Bob each roll a die individually once the experiment has started, or shared, meaning Alice and Bob roll a die before the game has started and store the result somewhere, then only look at it once the experiment is underway. In either case, the strategy Alice and Bob use to produce correlations can be specified by a probability distribution $r(\lambda)$ over some classical randomness, along with a variables $p_a^{i,\lambda}$, $q_b^{j,\lambda}$ giving the probability of Alice (resp. Bob) giving response a (resp. b) to question i (resp. j) when classical randomness takes value λ. These variables must form a probability distribution, so $p_a^{i,\lambda}, q_b^{j,\lambda} \geq 0$ and

$$\sum_{a \in \mathcal{O}_A} p_a^{i,\lambda} = \sum_{b \in \mathcal{O}_B} q_b^{j,\lambda} = 1 \tag{1.1.3}$$

and for all λ, i, j. The correlations produced by these classical strategies satisfy

$$p(a,b|i,j) = \int_\lambda r(\lambda) p_a^{i,\lambda} q_b^{j,\lambda}. \tag{1.1.4}$$

where we made use of the measurement independence assumption in assuming λ was independent of i and j. We let C_c denote the set of all classical correlations, and $C_c(n,m)$ denote all classical correlations with fixed question and answer set sizes $|\mathcal{I}_A| = |\mathcal{I}_B| = n$ and $|\mathcal{O}_A| = |\mathcal{O}_B| = m$.

One way of understanding C_c is by noting that any correlation produced by random classical strategies can be thought of as a mixture of correlations produced by deterministic classical strategies. Put differently, C_c can be viewed as the convex hull of the deterministic classical correlations defined in Equation (1.1.2) (recall that we defined addition and scalar multiplication of correlations by viewing them as tuples in $|\mathcal{I}_A||\mathcal{I}_B||\mathcal{O}_A||\mathcal{O}_B|$). From this observation it follows that the set of correlations C_c is both closed and convex.

Tensor Product Correlations

Now we consider the case where Alice and Bob have access to quantum resources. In particular, they mare share an entangled state, and measure it before deciding on a response to send to the verifier. We can describe such a strategy by specifying the shared state ρ and positive operator-valued measure (POVM) operators $\{P_a^i\}$ and $\{Q_b^j\}$ describing the

measurements made by Alice and Bob after receiving questions i and j, respectively. To enforce the condition that Alice and Bob cannot communicate during the game we demand that the state ρ live in a separable Hilbert space $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$ with Alice's measurement operators living entirely in $\mathcal{H}_A$ and Bob's in $\mathcal{H}_B$. (That is, for any $i \in \mathcal{I}_A$ and $a \in \mathcal{O}_A$ we have $P_a^i = \left(\tilde{P}_a^i\right)_A \otimes I_B$, and similarly for Bob). The correlations produced by these strategies are given by

$$p(a, b|i, j) = \mathrm{tr}\left[P_a^i Q_b^j \rho\right]. \tag{1.1.5}$$

We let C_q denote the set of all correlations that can be realized by strategies where Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$ are both finite dimensional. The POVM formalism allows for mixtures of strategies, so it follows that C_q is convex. Furthermore, a Naimark dilation (or "church of the large Hilbert space" argument) shows that any correlation in C_q can be realized by a strategy where measurement operators $\{P_a^i\}$ and $\{Q_b^j\}$ are projective, and $\rho = |\psi\rangle\langle\psi|$ is a pure state. In this case, correlations take the form

$$p(a, b|i, j) = \langle\psi|P_a^i Q_b^j|\psi\rangle = \left\|P_a^i Q_b^j |\psi\rangle\right\|^2 \tag{1.1.6}$$

We can also consider the correlations which can be produced by strategies with $\mathcal{H}_A$ and $\mathcal{H}_B$ infinite. We denote this set C_{qs} (for quantum spatial – referring to the "spatial" tensor product). As with C_q, mixing strategies shows that C_{qs} is convex, and a Naimark dilation argument gives that correlations in C_{qs} can be achieved with pure states ρ and projective measurements.

It was shown in [60] that neither C_q nor C_{qs} are closed. The set C_{qa} (standing for "quantum approximable") denotes the closure of C_q. By a result in [57] C_{qa} contains C_{qs}, and so C_{qa} is also the closure of C_{qs}. This set is closed and convex, and can be viewed as the set of all correlations approximable by tensor product strategies.

Commuting Operator Correlations

So far when discussing quantum strategies we enforced the condition that Alice and Bob can't communicate by demanding the Hilbert space they share be factorizable, with each player only measuring their half of the space. But there is another constraint we could have chosen to enforce the no communication condition. If we simply demand that each of Alice's measurement operators commute with each of Bob's, so

$$P_a^i Q_b^j - Q_b^j P_a^i = 0 \tag{1.1.7}$$

for all $i, j, a, b \in \mathcal{I}_A, \mathcal{I}_B, \mathcal{O}_A, \mathcal{O}_B$ we would also known that no information is exchanged between Alice and Bob during the measurement process. Any correlation which can be produced by operators which commute in this way and act on a finite dimensional Hilbert space can also be produced by measurement operators acting on a tensor product Hilbert space [57]. However commuting operators acting on a infinite dimensional Hilbert space can produce correlations which can't be reproduced by operators acting on any tensor product Hilbert space. Even stronger, these correlations aren't even the closure of the set of correlations that can be produced by tensor-product strategies [36]. We denote the set of correlations producible by commuting operator strategies C_{qc} (for "quantum commuting"). This set is both closed and convex, and strictly larger than C_{qa}.

Separations Between Correlation Sets

Table 1.1 summarizes the correlations introduced in the previous sections.

C_c	Classical correlations.
C_q	Quantum correlations, produced by tensor product (equivalently commuting) measurements on states living in a finite dimensional Hilbert space.
C_{qs}	Quantum seperable correlations, produced by tensor product measurements made on states living in a potentially infinite dimensional Hilbert space.
C_{qa}	Quantum approximable correlations, defined to be the closure of C_q (or equivalently the closure of C_{qs} [57])
C_{qc}	Quantum commuting correlations, produced by commuting measurements made on a potentially infinite dimensional Hilbert space.

Table 1.1: Classical and Quantum Correlation Sets

These correlations form an increasing chain of correlation sets, which can be summarized by the equation

$$C_c \overset{[12,\,4]}{\subset} C_q \overset{[16]}{\subset} C_{qs} \overset{[60]}{\subset} C_{qa} \overset{[36]}{\subset} C_{qc}. \tag{1.1.8}$$

While it will not be the focus of this thesis, it should be pointed out that the proofs that each of these inclusions is proper represented a major breakthrough in our understanding of quantum correlations. In particular, the final inclusion $C_{qa} \subset C_{qc}$ was proven to be strict very recently [36]. The proof of this fact is equivalent to a disproof of Connes' embedding conjecture, a longstanding conjecture in mathematics.

1.1.2 Nonlocal Games

The separations between correlation sets discussed in the previous section motivate a type of test which the verifier can use to check the resources shared by Alice and Bob. The verifier can ask Alice and Bob to produce a certain correlation, then send questions and record the Alice and Bob's responses. If the responses are consistent with a certain correlation, the verifier concludes that Alice and Bob share the resources needed to produce it. For example, by asking Alice and Bob to produce a correlation which is in C_q but not in C_c, the verifier can check that Alice and Bob share an entangled state. One can imagine tests built on the other strict inclusions which test for infinite dimensional states, or the presence non-separable Hilbert spaces allowing for commuting operator style measurements.

But there are some theoretical obstacles to such a test. Firstly, the verifier cannot know exactly the correlation that would be produced by the player's strategy, but can only approximate it based on statistics collected from the player's response. This type of approximation can never distinguish between a set of correlations and it's closure, so no test of this form can distinguish between correlations in C_q, C_{qs}, and C_{qa} (for example). Furthermore, collecting these statistics requires many repeated rounds of interaction with the players. So far our analysis has assumed Alice and Bob are "memoryless," picking a strategy before the test begins and then following that same strategy every time they are asked a question. If Alice and Bob keep a list of the list of questions asked and their responses to each

round of the game and vary their future responses depending on that, they can potentially spoof a correlation which they could not have produced with their given resources [1].

Value of a Nonlocal Game

A nonlocal game is a way of defining a test that avoids these difficulties. A nonlocal game $\mathcal{G}$ with question sets $\mathcal{I}_A, \mathcal{I}_B$ and response sets $\mathcal{O}_A, \mathcal{O}_B$ is defined by a probability distribution π on $\mathcal{I}_A \otimes \mathcal{I}_B$ and a "scoring" function $V : \mathcal{I}_A \otimes \mathcal{I}_B \otimes \mathcal{O}_A \otimes \mathcal{O}_B \to [0, 1]$. Given a nonlocal game $\mathcal{G}$ and a set of correlations C we define the value the correlations C achieve on the nonlocal game as

$$\sup_{\{p\} \in C} \left(\sum_{i,j,a,b} V(a, b, i, j)\pi(i, j)p(a, b|i, j) \right) \tag{1.1.9}$$

with the sum taken over all $i, j \in \mathcal{I}_A, \mathcal{I}_B$ and $a, b \in \mathcal{O}_A, \mathcal{O}_B$. Important values are the classical value of a game

$$\omega(\mathcal{G}) = \sup_{\{p\} \in C_c} \left(\sum_{i,j,a,b} V(a, b, i, j)\pi(i, j)p(a, b|i, j) \right) \tag{1.1.10}$$

the tensor product value

$$\omega^*_{tp}(\mathcal{G}) = \sup_{\{p\} \in C_q} \left(\sum_{i,j,a,b} V(a, b, i, j)\pi(i, j)p(a, b|i, j) \right) \tag{1.1.11}$$

$$= \sup_{\{p\} \in C_{qs}} \left(\sum_{i,j,a,b} V(a, b, i, j)\pi(i, j)p(a, b|i, j) \right) \tag{1.1.12}$$

$$= \sup_{\{p\} \in C_{qa}} \left(\sum_{i,j,a,b} V(a, b, i, j)\pi(i, j)p(a, b|i, j) \right) \tag{1.1.13}$$

(note the supremum makes the value achieved by these three sets of correlations equivalent, since $C_{qa} \supset C_{qs}$ is the closure of C_q) and the commuting operator value

$$\omega^*_{co}(\mathcal{G}) = \sup_{\{p\} \in C_{qc}} \left(\sum_{i,j,a,b} V(a, b, i, j)\pi(i, j)p(a, b|i, j) \right). \tag{1.1.14}$$

Nonlocal Games as Tests of Resources

If the value achieved on a nonlocal game differs between two correlation sets this game can be used to test the resources shared by Alice and Bob. As a concrete example, consider a game $\mathcal{G}$ with $\omega(G) < \omega^*_{tp}(\mathcal{G})$ (the Bell test can be phrased as such a game). To test the resources shared by Alice and Bob the verifier plays many rounds of this game with Alice and Bob, choosing questions according to the distribution π and scoring the player's responses according to the value function V. Afterwards, the verifier computes the average score achieved by the players over all the rounds of the game. If this score exceeds $\omega(\mathcal{G})$ the verifier concludes that players can produce correlations outside the correlation set C_c.[1] Because $\omega^*_{tp}(\mathcal{G}) > \omega(\mathcal{G})$ there must be some strategy that players with access to quantum resources can follow which will allow them to achieve an average score greater than $\omega(\mathcal{G})$ on the game. Thus, the nonlocal game $\mathcal{G}$ allows players with quantum resources to convince the verifier of this fact.[2]

It is worth discussing why the nonlocal games formalism avoids the problems with the correlation test discussed in the previous section. First, note the supremum in the definition guarantees that if $\omega^*_{tp}(\mathcal{G}) > \omega(\mathcal{G})$ then they are separated by some finite amount ϵ, and this separation can detected with a finite number of tests. More subtly, because the nonlocal game scoring function

$$\sum_{i,j,a,b} V(a,b,i,j)\pi(i,j)p(a,b|i,j) \tag{1.1.15}$$

is linear in the conditional probabilities $p(a,b|i,j)$, the expected score achieved by the players over all the rounds is just the sum of the expected scores achieved by their strategies on each round, and the players having a memory of previous rounds doesn't change the overall maximum expected score they can achieve on the game.

[1] Technically, the verifier concludes this with some probability, but that probability goes to 1 provided the average value achieved by the players stays a constant distance above $\omega(\mathcal{G})$ as the number of rounds goes to infinity.

[2] Technically, they only convince the verifier they have access to some super-classical resource, capable of producing a larger set of correlations than C_c *or* that they have managed to exploit correlations such that $\pi(i,j|\lambda) \neq \pi(i,j)$.

The Game Functional

When working with with nonlocal games it is often helpful to rewrite the expected score the players achieve when playing the game as a function of their strategy. We can then express the value of the game as a supremum over strategies.

In the classical case, linearity of the scoring function tells us that the optimal classical strategy for the game will be a deterministic strategy. We can write the expected score achieved by these strategies as

$$\sum_{i,j,a,b} V(a,b,i,j)\pi(i,j)\delta_{a,a_i}\delta_{b,b_j} \tag{1.1.16}$$

and the classical value of the game as

$$\omega(\mathcal{G}) = \max_{\{a_i\},\{b_j\}} V(a,b,i,j)\pi(i,j)\delta_{a,a_i}\delta_{b,b_j} \tag{1.1.17}$$

with the maximum taken over all possible assignments of deterministic responses $\{a_i\}, \{b_j\}$ in $\mathcal{O}_A^{|\mathcal{I}_A|}, \mathcal{O}_B^{|\mathcal{I}_B|}$, respectively.

In the quantum case we can restrict to pure states and projective measurements (by the Naimark dilation) and write the expected score achieved by players following a certain strategy as

$$\sum_{i,j,a,b} V(a,b,i,j)\pi(i,j)\,\langle\psi|P_a^i Q_b^j|\psi\rangle = \langle\psi|\Phi(\mathcal{G})|\psi\rangle\,, \tag{1.1.18}$$

where we have introduced the $\Phi_{\mathcal{G}}$ shorthand to encompass the operator part of the scoring function of the game. We will sometimes refer to $\Phi(\mathcal{G})$ as the game polynomial.

We can then obtain the tensor product and commuting operator values of a game by taking the supremum of $\langle\psi|\Phi_{\mathcal{G}}|\psi\rangle$ over strategies corresponding to tensor-product and commuting operator strategies, respectively.

For some games we will use operators other than $\delta_{a,a_i}, P_a^i, Q_b^j$ to allow us to write the classical and quantum scoring functions of the game in a more convenient form. We will see one example of this when discussing XOR games in Section 1.1.2.

Perfect Games

An important subclass of games are games for which the scoring function V evaluates to either 0 or 1. For these games we can divide sets of questions and responses into valid of "winning" responses for which $V(a,b,i,j)=1$ and invalid or "losing" responses with $V(a,b,i,j)=0$.

A strategy for one of these games is called perfect if it achieves an expected score of 1, that is, if it produces only winning responses. Games with perfect quantum strategies but no perfect classical strategies (so $\omega(\mathcal{G}) < \omega^*_{tp}(\mathcal{G}) = 1$) are called pseudotelepathy games.

XOR Games

One family of nonlocal games that will be central to this thesis are XOR games. These are games with response sets $\mathcal{O}_A = \mathcal{O}_B = \{0,1\}$ and question sets of arbitrary size (we normally take $\mathcal{I}_A = \mathcal{I}_B = \{1,2,...,n\}$, with n an arbitrary integer). Furthermore, the scoring function for these games takes value either zero or one and only depends on the parity of the players' responses so

$$V(0,0,i,j) = V(1,1,i,j) \text{ and} \tag{1.1.19}$$

$$V(0,1,i,j) = V(1,0,i,j) \tag{1.1.20}$$

for any $i,j \in \{1,2,...,n\}$. The distribution π over questions is usually taken to be uniform over some set of allowed questions. Then an XOR game can be specified by a set of allowed questions and a "winning" parity for each pair of questions (i,j) that can be sent to the players. For reasons that will become clear shortly, we specify this list of questions and parities via a system of equations

$$\hat{X}_{i_1} + \hat{Y}_{j_1} = s_1 \tag{1.1.21}$$

$$\hat{X}_{i_2} + \hat{Y}_{j_2} = s_2 \tag{1.1.22}$$

$$\vdots$$

$$\hat{X}_{i_m} + \hat{Y}_{j_m} = s_m \tag{1.1.23}$$

with each equation $X_{i_t} + Y_{j_t} = s_t$ specifying that a pair of questions (i_t, j_t) is sent to the players with winning parity response $s_t \in \{0, 1\}$.

Classical strategies for answering an XOR game can be described by variables $\hat{X}_i, \hat{Y}_j$ specifying Alice/Bob's response to question i/j respectively. By identifying these variables with the $\hat{X}, \hat{Y}$ variables in the system of equations above, we see the classical value of an XOR game is equal to the maximum fraction of satisfiable clauses in the associated system of equations, with addition taken mod 2 (i.e. the system of equations viewed as a system of equations over $\mathbb{Z}_2$). It follows that it is easy to decide if an XOR game has a perfect classical strategy (via Gaussian elimination in $\mathbb{Z}_2$). Classically, we can think of an XOR game as testing the satisfiability of the associated system of equations.

It is convenient to describe quantum strategies for XOR games in terms of observables

$$X_i = 1 - 2P_1^i \text{ and } Y_j = 1 - 2Q_1^j. \tag{1.1.24}$$

Note that

$$P_0^i = 1 - P_1^i = \frac{1}{2}\left(1 - X_i\right), \tag{1.1.25}$$

with similar equations holding for Y_j and Q_0^j, so specifying the matrices X_i, Y_j for $i, j \in \{1, ..., n\}$ completely specifies the projectors associated with the game. By construction, the matrices X_i and Y_j square to the identity. They live in separate factors of a Hilbert space if they correspond to a tensor product strategy, and commute if they correspond to a commuting operator strategy. In terms of these matrices the game polynomial for an XOR game can be written

$$\Phi(\mathcal{G}) = \frac{1}{2}\left(1 + \frac{1}{m}\sum_{t=1}^{m} X_{i_t} Y_{j_t} (-1)^{s_t}\right) \tag{1.1.26}$$

where the i_t, j_t, s_t come from the system of equations corresponding to the game outlined in Equations (1.1.21) to (1.1.23).

The tensor product and commuting operator values of XOR games coincide, and can be computed efficiently [63]. Crucially, this value can sometimes be higher than the classical

value of an XOR game, meaning that there are XOR games which allow for simple tests of quantum resources. The Bell test can be phrased as such a game.

1.1.3 Multipartite Correlations

Thus far our discussion has only involved experiments involving one verifier and two players. But all the concepts discussed, including correlations sets, nonlocal games, perfect games, and XOR games, generalize naturally to a multi-player setting. Here tensor-product correlations involve a Hilbert space that factors into separate pieces for each player, and commuting operator correlations involve the constraint that all operators corresponding two two different players' measurements must commute.

The primary complication in the multiplayer setting is notational. When dealing with three or more players we abandon the P, Q notation for projectors and instead define $P(\alpha)^i_a$ to be the projector corresponding to player α giving response a to question i. Similarly, we define

$$X_i^{(\alpha)} = 1 - 2P(\alpha)^i_1 \tag{1.1.27}$$

when discussing multiplayer XOR games, though we will sometimes use X_i, Y_i, Z_i in the special case of three player XOR games.

1.2 Bounds on the Set of Correlations

Two closely related tasks are deciding whether a correlation belongs to the set of tensor product or commuting operator correlations and computing the tensor product or commuting operator value of a nonlocal game. A naive approach to either task requires searching over possibly infinite dimensional strategies, and so it is somewhat surprising that either question can be answered. Indeed, as mentioned in the introduction, both tasks are in general undecidable.

But there do exist some general techniques for lower bounding the set of tensor-product correlations and upper bounding the set of commuting operator-correlations. We discuss

those here.

1.2.1 Brute Force Lower Bound on Tensor Product Correlations

The first approach to discuss is a slight refinement of the naive "brute force search" discussed in the introduction to this section. Consider fixing a dimension d and searching over and ϵ-net covering all matrices of dimension $\leq d$ for strategies that approximate a given correlation to within error ϵ. The set of correlations producible by these strategies is contained in the set of tensor product correlations, that is it is an inner approximation to the set of tensor product correlations. Furthermore, this inner approximation converges to C_{qa} as $d \to \infty$ and $\epsilon \to 0$. Recall from Section 1.1.1 that $C_{qs} \subset C_{qa}$, or equivalently that any infinite dimensional tensor product correlation can be approximated by a finite dimensional ones. Then there exists a algorithm which provides a converging inner approximation to C_{qa}, or equivalently an algorithm which gives a lower bound converging to $\omega^*_{tp}(\mathcal{G})$ for any nonlocal game $\mathcal{G}$.

1.2.2 ncSoS Upper Bound on Commuting Operator Correlations

It is also possible to describe a set of correlations containing the commuting operator correlations, i.e. obtain an outer approximation to the set of commuting operator correlations via bounds coming from the non-commutative sum of squares (ncSoS) hierarchy. These bounds were developed independently in [46] and [32, 19]. There are several different ways to understand the ncSOS bounds, including as result of a noncommutative positivstellensatz, or (relatedly) as obtained from a proof system based on non-commutative sums of squares. We do not discuss those views here, and instead give a direct, somewhat "low-level" view of the ncSOS approach which parallels the discussion in [46].

For notational convenience we discuss the ncSoS approach in the case of bipartite (two-player) correlations with question set $\mathcal{I}_A = \mathcal{I}_B = \{1, 2\}$ and response set $\mathcal{O}_A = \mathcal{O}_B = \{0, 1\}$. The key initial observation is that, for any projectors P^i_a, Q^j_b and state $|\psi\rangle$ we must have

$$\sum_p \langle \psi | p^* p | \psi \rangle \geq 0 \tag{1.2.1}$$

for any sum over complex polynomials p formed from the projectors P_a^i, Q_b^j (so, for example, we could have a polynomial $p = P_0^1 + (2+i)Q_0^1 - 5P_0^1P_0^2Q_1^1$), and with $*$ denoting the conjugate transpose. To encode this constraint algorithmically, we first define the vector $v^{(d)}$ of matrices associated with some commuting operator strategy to be the vector consisting of all monomials of degree $\leq d$ formed from projectors P and Q, so, for example

$$v^{(2)} = (1, P_0^1, P_1^1, Q_0^1, Q_1^1, (P_0^1)^2, P_0^1P_1^1, P_0^1Q_0^1, ..., (Q_1^1)^2). \tag{1.2.2}$$

Then the moment matrix of degree $2d$ associated with the strategy (sometimes called the Hankel matrix) $M^{(2d)}$ is defined entrywise by setting

$$M_{ij}^{(2d)} = \langle\psi|\left(v^{(d)}\right)_i^* \left(v^{(d)}\right)_j|\psi\rangle\,. \tag{1.2.3}$$

Note that the correlations produced by a strategy (or the value a strategy achieves on a nonlocal game) can be read off from the matrix $M^{(2d)}$ for any $d \geq 1$. We sometimes refer to these correlations as "coming from" the moment matrix $M^{(2d)}$. The constraint of Eq. (1.2.1) applied to all polynomials p of degree $\leq d$ is equivalent to the demand that $M^{(2d)}$ be positive semi-definite.

If the projectors P and Q come from a commuting operator strategy the entries of the moment matrix $M^{(2d)}$ satisfy additional constraints, for example

$$\langle\psi|P_a^iQ_b^j|\psi\rangle = \langle\psi|Q_b^jP_a^i|\psi\rangle \tag{1.2.4}$$

and

$$\langle\psi|P_0^i|\psi\rangle + \langle\psi|P_1^i|\psi\rangle = \langle\psi|1|\psi\rangle = 1. \tag{1.2.5}$$

We do not list all those constraints here, but instead refer the reader to [46] for a complete list.

Now we do not restrict ourselves to moment matrices coming from commuting operator strategies, but instead call any psd matrix of the correct dimensions satisfying and satisfying

constraints like the ones listed in Equations (1.2.4) and (1.2.5) a moment matrix. Thus the set of all degree $2d$ moment matrices includes all matrices associated with commuting operator strategies, but can be larger. The degree $2d$ (or level d) ncSoS approximation to the commuting operator correlations is the set of correlations coming from any degree $2d$ moment matrix. The degree $2d$ ncSoS upper bound on the commuting operator value of a game is the supremum value achievable by correlations coming from degree $2d$ moment matrices.

As an example of a ncSoS type bound, we consider the CHSH game [12] – an XOR game encoding the Bell test with clauses

$$\hat{X}_0 + \hat{Y}_0 = 0 \tag{1.2.6}$$

$$\hat{X}_0 + \hat{Y}_1 = 0 \tag{1.2.7}$$

$$\hat{X}_1 + \hat{Y}_0 = 0 \tag{1.2.8}$$

$$\hat{X}_1 + \hat{Y}_1 = 0, \tag{1.2.9}$$

hence game polynomial

$$\Phi_{CHSH} = \frac{1}{2}\left(1 + \frac{1}{4}(X_0Y_0 + X_0Y_1 + X_1Y_0 - X_1Y_1)\right). \tag{1.2.10}$$

Then (working with observables X and Y as opposed to projectors and recalling that X and Y observables commute with each other and square to the identity) we see

$$\left(\frac{1}{\sqrt{2}}(X_0 + X_1) - Y_0\right)^2 + \left(\frac{1}{\sqrt{2}}(X_0 - X_1) - Y_1\right)^2 \tag{1.2.11}$$

$$= \frac{1}{2}(X_0 + X_1)^2 + Y_0^2 - \sqrt{2}(X_0Y_0 + X_1Y_0) \tag{1.2.12}$$

$$+ \frac{1}{2}(X_0 - X_1)^2 + Y_1^2 - \sqrt{2}(X_0Y_1 - X_1Y_1) \tag{1.2.13}$$

$$= 4 - \sqrt{2}(X_0Y_0 + X_1Y_0 + X_0Y_1 - X_1Y_1) \tag{1.2.14}$$

$$= 4(1 + \sqrt{2} - 2\sqrt{2}\Phi_{CHSH}). \tag{1.2.15}$$

Thus for any commuting observables X and Y and state ψ we have

$$\langle\psi|1+\sqrt{2}-2\sqrt{2}\Phi_{CHSH}|\psi\rangle \geq 0 \implies \langle\psi|\Phi_{CHSH}|\psi\rangle \leq \frac{1}{2}\left(1+\frac{1}{\sqrt{2}}\right) \tag{1.2.16}$$

hence the commuting operator value of the CHSH game is at most $\frac{1}{2}\left(1+\frac{1}{\sqrt{2}}\right)$ (which ends up being optimal). Because this bound involved the squares of polynomials of degree at most 1, it can be produced algorithmically by the ncSoS algorithm run to level 1 (or equivalently degree 2). In fact, a result of Tsierlson shows that a tight upper bound for the commuting operator value of all 2 player XOR games can be obtained by level 1 ncSoS [63]. (In the same result Tsierlson also showed that tensor product and commuting operator values of 2 player XOR games always coincide).

As d is increased the level d ncSoS approximation to the set of commuting operator correlations gets more restrictive, since we need to extend matrices to larger matrices while keeping them PSD and satisfying the commuting operator strategy constraints. Put differently, any correlation coming from a degree $2d$ moment matrix also comes from a degree $2d-2$ moment matrix, since a truncation of a psd matrix remains psd. The key result of [46] is that ncSoS approximation to the set of commuting operator correlations converges to the set of commuting operator correlations as $d \to \infty$. Equivalently, for any game $\mathcal{G}$ the degree d ncSoS upper bound on $\omega^*_{co}(\mathcal{G})$ converges to the true commuting operator value of the game from above as $d \to \infty$.

1.2.3 Computing the Value of a Nonlocal Game

Brute force search gives a converging series of inner approximations to the set of tensor product correlations, while the ncSoS hierarchy gives a converging series of outer approximations to the set of commuting operator correlations. Consequently, for any nonlocal game $\mathcal{G}$, there exists a series of computable lower bounds which converge to $\omega^*_{tp}(\mathcal{G})$ from below, and a series of computable upper bounds which converge to $\omega^*_{co}(\mathcal{G})$ from above. That means that if the commuting operator and tensor product values of a game ever coincide, there exists an algorithm which can approximate the value of that game to arbitrary precision.

Equally interesting is the contrapositive of this statement: if there exists a nonlocal game

whose tensor product or commuting operator value cannot be approximated to arbitrary precision (i.e. a game for which it is undecidable whether $\omega^*_{tp} > C$ or $\omega^*_{tp} < C - \epsilon$ for constant ϵ) then the tensor product and commuting operator values of the game must differ by a constant amount. This is exactly the type of argument used in [36] to prove the separation between C_{qa} and C_{qc}.

1.3 Mathematical Tools

This section contains a "quick and dirty" introduction to the main mathematical tools that will be used in this thesis. A reader interested in a more formal introduction is encouraged to look at one of the many excellent textbooks on the subject.

1.3.1 Groups, Algebras, and Group Algebras

Groups

A group consists of a set of elements S together with a binary (product) operator $\cdot$ which satisfy the following rules:

1. **Closure:** For all $a, b \in S$, $a \cdot b = c \in S$.
2. **Associativity:** For all $a, b, c \in S$, $(a \cdot b) \cdot c = a \cdot (b \cdot c)$.
3. **Identity Element:** There exists an element $1 \in S$ satisfying $1 \cdot a = a$ for all $a \in S$.
4. **Inverses:** For all $a \in S$ there exists an element $a^{-1} \in S$ satisfying $a \cdot a^{-1} = 1$.

from here on we will omit the $\cdot$ notation when working with groups and just write $a \cdot b = ab$.

Group Presentations

We will frequently describe groups using the language of group presentations. To understand this language we first define the free group generated by a set of generators $G = \{g_1, ..., g_m\}$ to be the group consisting of all words formed from products of elements $g_1, ..., g_m$ and

their inverses $g_1^{-1}, ..., g_m^{-1}$, with the product of two words being their composition and the understanding that any generator g_i and it's inverse cancel to the identity, so

$$g_i g_i^{-1} = 1 \text{ and hence} \tag{1.3.1}$$

$$w_1 g_i g_i^{-1} w_2 = w_1(1)w_2 = w_1 w_2 \text{ for all } g_i \in G \tag{1.3.2}$$

where w_1 and w_2 are arbitrary words in the group.

Next, we define a set of relations R to be a set of rewrite rules on the free group. These rewrite rules can be presented as equalities of the form

$$w_i = w_j \tag{1.3.3}$$

with each w_i, w_j a word in the free group or more compactly as a set of words $R = \{r_1, r_2, ...r_k\}$ with the understanding that each $r_i \in R$ corresponds to an equality of the form

$$r_i = 1. \tag{1.3.4}$$

Finally, we define the group presented by a set of generators G and relations R to be the group consisting of all words formed by the generators G with equality between any two words that can be transformed into each other using the rewrite rules R (and the basic free group rewrite rule $g_i g_i^{-1} = 1$).[3] It should be pointed out that the rewrite rules in R can be used to both increase and decrease the length of a word, so determining equality between two words in a group presentation can be a difficult task (and is, in general, undecidable [44, 48]).

Algebras

An associative algebra (hereafter simply called an algebra) consists of a vector space together with a bilinear product. Slightly more explicitly, an algebra $\mathcal{A}$ over a field K consists of a vector space over K along with a mapping $\cdot$ satisfying

[3] A more formal definition for those already familiar with the terminology: The group with generators G and relations R is isomorphic to the quotient of the free group generated by G by the normal subgroup generated by R.

1. $a \cdot (b + c) = a \cdot b + a \cdot c$

2. $(a + b) \cdot c = a \cdot c + b \cdot c$

3. $k_1 a \cdot k_2 b = k_1 k_2 (a \cdot b)$

for all $a, b, c \in \mathcal{A}$ and $k_1, k_2 \in K$. Everywhere in this thesis we will take $K = \mathbb{C}$, so all algebras considered are algebras over the complex numbers. We call these "complex" algebras.

A complex $*$-algebra is an algebra together with an involution operator $*$ which respects the bilinear product and acts like the normal adjoint on complex numbers, so $a^* \in \mathcal{A}$ for all $a \in \mathcal{A}$ with

1. $(a^*)^* = a$

2. $(a \cdot b)^* = b^* \cdot a^*$

3. $(a + b)^* = a^* + b^*$ and

4. $(\alpha a) = \alpha^* a^*$

for all $a, b \in \mathcal{A}$ and $\alpha \in \mathbb{C}$. Finally, a C^*-algebra is a $*$ algebra together with a norm $\|\|$ such that the algebra is complete in the metric induced by the norm and

$$\|a^* a\| = \|a^*\| \|a\| \tag{1.3.5}$$

for all a in the algebra.

Group Algebras

Given a group G, the group algebra $\mathbb{C}[G]$ is a C^*-algebra with elements of the form $\sum_g \beta_g g$ with $g \in G, \beta_g \in \mathbb{C}$, multiplication inherited from the group multiplication so $g_1 \cdot g_2 = (g_1 g_2)$ and $*$ operation defined by $g^* = g^{-1}$. Informally we can think of $\mathbb{C}[G]$ as being "the algebra formed from polynomials of elements in G".

To define a norm on $\mathbb{C}[G]$ (to make it a C^*-algebra) we note that we can view elements of the algebra as acting by left multiplication on $L^2(G)$, i.e. square summable complex valued functions of G. Then the norm is just the standard operator norm.

Subgroups, Subalgebras, Ideals and Left Ideals

Given a group G a subgroup H of G is a subset of elements from G which contain the identity element, inverses, and is closed under the group binary operation. Then H itself forms a group, with the same binary operator $\cdot$ as defined for G. Given a set S of elements from G the sugbroup generated by S, denoted $\langle S \rangle$ (or $\langle S \rangle_G$ when the group G may be unclear), is the smallest subgroup of G containing the set S. We can think of $\langle S \rangle$ as the group formed by mulitplication of elements of S and their inverses using the binary operator defined on G.

Given an algebra $\mathcal{A}$, a subalgebra $\mathcal{C}$ is defined to be a vector subspace of $\mathcal{A}$ which is closed under multiplication of vectors (but not inverses). Because $\mathcal{C}$ does not need to contain inverses it may not contain a multiplicative identity. An subalgebra that does is called a unital subalgebra. A subalgebra of a $*$-algebra closed under the $*$ operation is called a $*$-subalgebra. Finally, the subalgebra (resp $*$-subalgebra) generated by a set S of elements coming from $\mathcal{A}$ is defined to be the smallest subalgebra (resp $*$-subalgebra) of $\mathcal{A}$ containing S.

Similarly, given an algebra $\mathcal{A}$ an ideal $\mathcal{I}$ is a subset of elements in $\mathcal{A}$ which is closed under addition and "absorbs multiplication", so for any $b \in \mathcal{I}$ we have $ab \in \mathcal{I}$ and $ba \in \mathcal{I}$ where $a \in \mathcal{A}$ is arbitrary. The ideal generated by a set S of elements coming from $\mathcal{A}$ is the smallest ideal of $\mathcal{A}$ containing S. A left ideal (or right ideal) $L\mathcal{I}$ of $\mathcal{A}$ is defined similarly, except it only adsorbs multiplication from the left (or right).

1.3.2 Representations

Generally speaking, a representation is a map between mathematical objects which preserves the structure of the objects in the pre-image. In this thesis we will be largely focus on representations π mapping groups or algebras into the algebra of bounded operators on a Hilbert space $\mathcal{H}$, which we denote $\mathcal{B}(\mathcal{H})$. In the groups case this means π is a mapping from G to operators which satisfies $\pi(g_1)\pi(g_2) = \pi(g_1 g_2)$ for any $g_1, g_2 \in G$. In the algebras case π is a $*$-representation, meaning it satisfies the multiplicative condition above along with the condition $\pi(a+b) = \pi(a) + \pi(b)$, $\pi(\alpha a) = \alpha\pi(a)$ and $\pi(a^*) = \pi(a)^*$ for any $a, b \in \mathcal{A}$, $\alpha \in \mathbb{C}$.

This representations language lets us view quantum operators as arising from representations of groups or algebras. We can then impose structure on the operators by asking for

representations of groups or algebras with the desired structure. This view is elaborated on greatly in Chapter 2.

1.4 Results in this Thesis

In Chapter 2 we construct a general algebraic framework, based on a result in noncommutative algebraic geometry known as a Nullstellensatz (NullSS) which we will use to analyze XOR games. In Chapter 3 we discuss a condition called the noPREF condition which guarantees existence of a perfect commuting operator strategy for an XOR game, along with a tensor product strategy for XOR games we call MERP. We then show MERP strategies are optimal for games meeting the noPREF condition and use this result to analyze a class of games called symmetric XOR games. In Chapter 4 we continue our analysis of XOR games and show that MERP strategies are optimal for any 3 player XOR game. This result hinges on an involved algebraic proof built around instances of the subgroup membership problem. In Chapter 5 we use techniques from the previous chapters to construct families of XOR games with desirable properties, and to analyze randomly generated XOR games. Finally, Chapter 6 concludes the thesis with a discussion of some open problems motivated by the results of the thesis.

Chapter 2

Algebraic Framework

In this chapter we introduce a general mathematical framework for studying the commuting-operator value of nonlocal games. We introduce the concept of a *Universal Game Algebra*, which is an algebra with generators satisfying the same relations as the projectors corresponding to a commuting-operator strategy, then show commuting operator strategies can be obtained from representations of this universal game algebra. We then connect the question of whether a nonlocal game has a perfect commuting operator strategy to a result in noncommutative algebraic geometry known as a Nullstellensatz. This connection gives an "algebraic" characterization of games with perfect commuting operator strategies in terms of ideals and sums of squares of elements in the universal game algebra. Finally, we show we can further simplify this algebraic characterization for a large class of games which includes XOR games. This simplification reduces the question of whether or not an XOR game has a perfect commuting operator strategy to the subgroup membership question – a well studied question in algebraic combinatorics.

The notation in this chapter is adapted to describe arbitrary many-player games and so is somewhat more involved than the notation in the introduction. This notation is introduced in Section 2.2.

2.1 Introduction

The foundations of classical Algebraic Geometry and Real Algebraic Geometry are NullSSs. Over the last two decades the basic analogous NullSS for NC variables have emerged.

This chapter concerns nonlocal quantum strategies for games, recalls NullSS which are helpful, extends these, and applies them to a very broad collection of games. In the process it brings together results spread over different literatures, hence rather than being terse our style is fairly expository.

2.2 Nonlocal Game Definitions

This section gives an overview of all the terminology used to discuss nonlocal games in this paper. Section 2.2.1 gives an overview of the technical definitions which are key to this paper. Then, in Section 2.2.2 we introduce an algebraic framework which we will use to describe nonlocal games and their commuting operator strategies. In Section 2.2.3 we describe some well-know families of games using the language introduced in previous sections. Finally, in Section 2.2.4 we describe the condition that a nonlocal game has perfect commuting operator strategy in terms of some of the notation introduced in previous sections.

2.2.1 Technical Definitions

Commuting Operator Strategies

We start with a definition of a commuting operator strategy for nonlocal games. The setting is a Hilbert space $\mathcal{H}$, possibly infinite dimensional. An important class of operators are all $\rho \in \mathcal{B}(\mathcal{H})$ which are positive semidefinite with trace 1. These are called density operators.

Definition 2.2.1. *A commuting operator strategy S for a k-player, n-question, m-response nonlocal game is defined by $(\rho, \mathcal{P}(1), \mathcal{P}(2), ..., \mathcal{P}(k))$ where ρ is a density operator and each $\mathcal{P}(\alpha)$ in $\{\mathcal{P}(1), ...\mathcal{P}(k)\}$ is a set of projectors acting on the same Hilbert space*

$$\mathcal{P}(\alpha) = \{P(\alpha)_a^i : i \in [n], a \in [m]\} \tag{2.2.1}$$

which satisfy

$$[P(\alpha)_a^i, P(\beta)_b^k] = 0 \; \forall \; \alpha \neq \beta \tag{2.2.2}$$

and

$$\sum_{a \in [m]} P(\alpha)_a^i = 1 \; \forall \; \alpha \in [k], i \in [n]. \tag{2.2.3}$$

Note that as a consequence of Equation (2.2.3) we have $P(\alpha)_a^i P(\alpha)_b^i = 0$ and hence $[P(\alpha)_a^i, P(\alpha)_b^i] = 0$ for any α, i and $a \neq b$.

Note. We will try and stick to the convention of having α, β, γ variables label players, i, j, k variables label questions, and a, b, c variables label responses. Using k for the number of players, n for the number of questions and m for the number of responses will also be standard.

Games and their Commuting Operator Value

A k-player, n-quesiton, m-response nonlocal game $\mathcal{G} = (V, \mu)$ is specified by a scoring function

$$V : [n]^k \times [m]^k \to [0, 1] \tag{2.2.4}$$

and a distribution μ on $[n]^k$. The score a strategy S obtains on a game $\mathcal{G} = (V, \mu)$ is given by

$$v(\mathcal{G}, S) = \sum_{\vec{i} \in [n]^k} \sum_{\vec{a} \in [m]^k} \mu\left(\vec{i}\right) V\left(\vec{i}, \vec{a}\right) \operatorname{Tr}\left[\prod_{\alpha \in [k]} P(\alpha)_{\vec{a}(\alpha)}^{\vec{i}(\alpha)} \rho\right] \tag{2.2.5}$$

The commuting operator value $\omega_{co}^*(\mathcal{G})$ of a game is defined to be the supremum value achieved over commuting operator strategies so

$$\omega_{co}^*(\mathcal{G}) = \sup_{S \in \mathcal{S}_{co}} v(\mathcal{G}, S). \tag{2.2.6}$$

where we have defined $\mathcal{S}_{co}(k, n, m)$ to be the set of all k-player, n-question, m-response commuting operator strategies. Often k, n, m are implied from context, and we just

write $\mathcal{S}_{co}$.

2.2.2 The Algebraic Picture

In this paper we think strategies $S \in \mathcal{S}_{co}$ as arising from from representations of an algebra we call the universal game algebra. We define that algebra next.

Universal Game Algebra

Here we define the *Universal Game Algebra* $\mathcal{UA}$ in terms of various generators and relations. These relations reflect the algebraic properties of projectors or related algebraic objects.

Projection Generators Define $\mathcal{UA}$ to be the *-algebra with generators $e(\alpha)^i_a$ which satisfy relations

$$[e(\alpha)^i_a, e(\beta)^j_b] = 0 \ \forall \ i, j, a, b, \alpha \neq \beta, \text{ and} \tag{2.2.7}$$

$$\left(e(\alpha)^i_a\right)^2 = \left(e(\alpha)^i_a\right)^* = e(\alpha)^i_a \tag{2.2.8}$$

$$\sum_a e(\alpha)^i_a = 1 \ \forall \ \alpha, i \tag{2.2.9}$$

with $i, j \in [n]; a, b \in [m]$, and $\alpha \in [k]$. (Technically we should define a different Universal Algebra for every different value n, m and k, so $\mathcal{UA} = \mathcal{UA}(n, m, k)$. We frequently omit this detail when n, m and k are clear from context.)

There are two common change of variables we will use when describing the algebra $\mathcal{UA}$.

Signature Matrix Generators The first change of variables is to generators satisfying the algebraic properties of signature matrices, defined by

$$x(\alpha)^i_a := 2e(\alpha)^i_a - 1. \tag{2.2.10}$$

These variables satisfy relations

$$x(\alpha)^i_a x(\beta)^j_b = x(\beta)^j_b x(\alpha)^i_a \ \forall\ i,j,a,b,\alpha \neq \beta, \text{ and} \tag{2.2.11}$$

$$\sum_a x(\alpha)^i_a = -(m-2) \tag{2.2.12}$$

$$\left(x(\alpha)^i_a\right)^* = x(\alpha)^i_a \tag{2.2.13}$$

$$\left(x(\alpha)^i_a\right)^2 = 1. \tag{2.2.14}$$

It is straightforward to check that the set of relations above gives a defining set of relations for the algebra $\mathcal{UA}$ written in terms of the $x(\alpha)^i_a$.

Cyclic Unitary Generators The second change of variables is to cyclic unitary generators, defined by

$$c^{(\alpha)}_i := \sum_a \exp\left(\frac{2\pi a}{m}\right) e(\alpha)^i_a. \tag{2.2.15}$$

These observables satisfy relations

$$c^{(\alpha)}_i c^{(\beta)}_j = c^{(\beta)}_j c^{(\alpha)}_i \ \forall\ i,j,\alpha \neq \beta, \text{ and} \tag{2.2.16}$$

$$\left(c^{(\alpha)}_i\right)^* = \left(c^{(\alpha)}_i\right)^{-1} \tag{2.2.17}$$

$$\left(c^{(\alpha)}_i\right)^m = 1. \tag{2.2.18}$$

Straightforward calculation using the inverse transformation

$$e(\alpha)^i_a = \frac{1}{m}\sum_b \left(\exp\left(\frac{-2\pi a}{m}\right) c^{(\alpha)}_i\right)^b \tag{2.2.19}$$

shows that Equations (2.2.16) to (2.2.18) also form a defining set of relations for $\mathcal{UA}$. This shows that $\mathcal{UA}$ can be viewed as the group algebra generated by the k fold direct product of the cyclic group of order m, or $\mathcal{UA} \cong \mathbb{C}\left[(\mathbb{Z}_m)^k\right]$.

A notable special case occurs when the answer set of the game contains only 2 responses.

In this case the cyclic observable and signature matrix change of variables are the same, since

$$c_i^{(\alpha)} = e(\alpha)_0^i - e(\alpha)_1^i = 2e(\alpha)_0^i - 1 = x(\alpha)_0^i \tag{2.2.20}$$

and

$$x(\alpha)_0^i = -x(\alpha)_1^i. \tag{2.2.21}$$

In this case we use the simplified notation $x_i^{(\alpha)} = c_i^{(\alpha)} = x(\alpha)_0^i$.

Strategies as Representations of the Universal Game Algebra

Now we make an observation that is key to understanding the rest of this chapter: any commuting operator strategy S can be described by a representation π of $\mathcal{UA}$ into bounded operators acting on a Hilbert space $\mathcal{H}$ and a density operator $\rho \in \mathcal{B}(\mathcal{H})$. Moreover, convexity arguments tell us that whenever a game has a perfect commuting operator strategy it has a perfect commuting operator strategy where ρ is rank 1, i.e. a projector onto a state $\psi \in \mathcal{H}$. This means that we can study perfect commuting operator strategies by studying representations of $\mathcal{UA}$ into bounded operators on some (possibly infinite dimensional) Hilbert space $\mathcal{H}$ along with the action of those representations on a state $\psi \in \mathcal{H}$. We make use of this observation in the subsequent sections.

An Algebraic Definition of the Commuting Operator Value of a Game

For any game $\mathcal{G}$ define the game polynomial $\Phi_{\mathcal{G}} \in \mathcal{UA}$ by

$$\Phi_{\mathcal{G}} = \sum_{\vec{i} \in [n]^k} \sum_{\vec{a} \in [m]^k} \mu\left(\vec{i}\right) V\left(\vec{i}, \vec{a}\right) \prod_{\alpha \in [k]} e(\alpha)_{\vec{j}(\alpha)}^{\vec{i}(\alpha)} \tag{2.2.22}$$

This game polynomial is an algebraic encoding of an average over all the winning responses for a game $\mathcal{G}$, weighted by the probability of the corresponding question being sent to the

players. Then, recalling the view of strategies as representations introduced in Section 2.2.2

$$\omega_{co}^*(\mathcal{G}) = \sup_{\pi,\rho} \mathrm{tr}[\pi(\Phi_{\mathcal{G}})\rho] \tag{2.2.23}$$

where the supremum is taken over all representations π of $\mathcal{UA}$ into bounded operators on a Hilbert space $\mathcal{H}$, and density operators $\rho \in \mathcal{B}(\mathcal{H})$. In particular, a game has a perfect commuting operator strategy iff there exists a representation π of $\mathcal{UA}$ into $\mathcal{H}$ and a density operator $\rho \in \mathcal{B}(\mathcal{H})$ with $\mathrm{tr}[\pi(\Phi_{\mathcal{G}})\rho] = 1$. By convexity, this condition is equivalent to the existence of a representation π of $\mathcal{UA}$ into $\mathcal{H}$ and a state $|\psi\rangle \in \mathcal{H}$ with $\pi(\Phi_{\mathcal{G}})\,|\psi\rangle = |\psi\rangle$.

In this paper we will restrict ourselves to the case where the image of V is $\{0,1\}$ and the distribution μ is uniform over a set of allowed questions. In this case a game can be specified by a universe of possible questions $\mathcal{Q}$ and sets $\mathcal{Y}(\vec{i})$ listing valid or "winning" response vectors to each question vector $\vec{i} \in \mathcal{Q}$. Using this notation we have

$$\Phi_{\mathcal{G}} = \frac{1}{|\mathcal{Q}|} \sum_{\vec{i}\in\mathcal{Q}} \sum_{\vec{a}\in\mathcal{Y}(\vec{i})} \prod_{\alpha\in[k]} e(\alpha)_{\vec{a}(\alpha)}^{\vec{i}(\alpha)}. \tag{2.2.24}$$

We also define the set of invalid responses $\mathcal{N}(\vec{i})$ to be the compliment to the set $\mathcal{Y}(\vec{i})$. These sets can also be used to specify a game.

2.2.3 Examples of games

XOR Games

XOR games are games with $m = 2$ responses which we interpret as a 0 or a 1. The valid responses $\mathcal{Y}(\vec{i})$ to each question vector $\vec{i}$ are all responses which sum to either 0 or sum to 1 mod 2.

The game polynomial of an XOR game takes the form

$$\Phi_{\mathcal{G}} = \frac{1}{2} + \frac{1}{2T} \sum_{t=1}^{T} (-1)^{s_t} \prod_{\alpha\in[k]} x_{\vec{i}_t(\alpha)}^{(\alpha)} \tag{2.2.25}$$

with $T > 0$ some integer, the vector $\vec{i}_t \in [n]^k$ and the integer $s_t \in \{0,1\}$ are arbitrary. We

refer to each monomial

$$(-1)^{s_t} \prod_{\alpha \in [k]} x^{(\alpha)}_{\vec{i}_t(\alpha)} \tag{2.2.26}$$

as a clause, so the game polynomial above corresponds to a T-clause XOR game.

2.2.4 Equations Corresponding to Perfect Games

A strategy is a perfect strategy for a given game if $v(\mathcal{G}, S) = 1$. These perfect strategies admit a nice characterization in terms of invalid and valid response sets.

Theorem 2.2.2. *A commuting operator strategy S with question set $\mathcal{Q}$ and valid response sets $\mathcal{Y}(\vec{i})$ is perfect for a game $\mathcal{G}$ iff*

$$\left(\left(\sum_{\vec{a} \in \mathcal{Y}(\vec{i})} \prod_{\alpha \in [k]} P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)} \right) - I \right) \rho = 0 \text{ for all } \vec{i} \in \mathcal{Q} \tag{2.2.27}$$

Equivalently, a strategy with question set $\mathcal{Q}$ and invalid response sets $\mathcal{N}(\vec{i})$ is perfect for $\mathcal{G}$ iff

$$\left(\prod_{\alpha \in [k]} P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)} \right) \rho = 0 \text{ for all } (\vec{i}, \vec{a}) \in (\mathcal{Q}, \mathcal{N}(\vec{i})). \tag{2.2.28}$$

Proof. By definition, a strategy for a nonlocal game is perfect iff

$$\frac{1}{|\mathcal{Q}|} \sum_{\vec{i} \in \mathcal{Q}} \mathrm{Tr} \left[\sum_{\vec{a} \in \mathcal{Y}(\vec{i})} \prod_{\alpha \in [k]} P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)} \rho \right] = 1. \tag{2.2.29}$$

Now, for all $\vec{i} \in \mathcal{Q}$ we have

$$\sum_{\vec{a} \in \mathcal{Y}(\vec{i})} \prod_{\alpha \in [k]} P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)} \leq \sum_{\vec{a} \in [m]} \prod_{\alpha \in [k]} P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)} = I \tag{2.2.30}$$

hence

$$\mathrm{Tr}\left[\sum_{\vec{a}\in\mathcal{Y}(\vec{i})}\prod_{\alpha\in[k]}P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)}\rho\right]\leq \mathrm{Tr}[\rho]=1 \tag{2.2.31}$$

and a game is perfect iff we have for all $\vec{i}\in\mathcal{Q}$:

$$\sum_{\vec{a}\in\mathcal{Y}(\vec{i})}\mathrm{Tr}\left[\left(\prod_{\alpha\in[k]}P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)}\right)\rho\right]=1 \tag{2.2.32}$$

$$\implies \mathrm{Tr}\left[\left(\left(\sum_{\vec{a}\in\mathcal{Y}(\vec{i})}\prod_{\alpha\in[k]}P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)}\right)-I\right)\rho\right]=0 \tag{2.2.33}$$

Again using that

$$\sum_{\vec{a}\in\mathcal{Y}(\vec{i})}\prod_{\alpha\in[k]}P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)}\leq I \tag{2.2.34}$$

we see

$$\left(\sum_{\vec{a}\in\mathcal{Y}(\vec{i})}\prod_{\alpha\in[k]}P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)}\right)-I \tag{2.2.35}$$

is negative semidefinite, hence Equation (2.2.33) implies

$$\left(\left(\sum_{\vec{a}\in\mathcal{Y}(\vec{i})}\prod_{\alpha\in[k]}P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)}\right)-I\right)\rho=0 \tag{2.2.36}$$

which finishes the first part of the proof.

To convert this condition from terms of $\mathcal{Y}$ to terms of $\mathcal{N}$ fix $\vec{i}\in\mathcal{Q}$ and use

$$\sum_{\vec{a}\in\mathcal{Y}(\vec{i})\cup\mathcal{N}(\vec{i})}\mathrm{Tr}\left[\left(\prod_{\alpha\in[k]}P(\alpha)^{\vec{i}(\alpha)}_{\vec{a}(\alpha)}\right)\rho\right]=\mathrm{Tr}[(I)\,\rho]=1. \tag{2.2.37}$$

In words we are summing over all responses valid or invalid to question $\vec{i}$. Subtract the first

line of Equation (2.2.33) from this to get $\sum_{\vec{a}\in\mathcal{N}(\vec{i})} = 0$. This is a sum of nonnegative terms, so each is 0:

$$\mathrm{Tr}\left[\left(\prod_{\alpha\in[k]} P(\alpha)_{\vec{a}(\alpha)}^{\vec{i}(\alpha)}\right)\rho\right] = 0 \;\forall\; \vec{a}\in\mathcal{N}(\vec{i}) \tag{2.2.38}$$

Since operators $P(\alpha)_a^i$ and ρ inside the trace are positive semidefinite we get their product is 0, thus finishing the proof.

$$\left(\prod_{\alpha\in[k]} P(\alpha)_{\vec{a}(\alpha)}^{\vec{i}(\alpha)}\right)\rho = 0 \;\forall\; \vec{a}\in\mathcal{N}(\vec{i}) \tag{2.2.39}$$

□

More generally, we can obtain a characterization of games described in terms of a game polynomial $\Phi_{\mathcal{G}}$, provided $\Phi_{\mathcal{G}}$ is written as a (weighted) average of contractions. This characterization uses the view of strategies as representations of $\mathcal{UA}$ described in Section 2.2.2 and applies naturally to XOR games.

Theorem 2.2.3. *A game $\mathcal{G}$ with game polynomial*

$$\Phi_{\mathcal{G}} = \sum_{t=1}^{T} \frac{1}{\nu_t} g_t \tag{2.2.40}$$

with T an integer, each ν_t a positive real number with $\sum_t \nu_t = 1$, and each g_t a polynomial in $\mathcal{UA}$ satisfying $g_t g_t^* \leq 1$ has a perfect strategy S iff there exists a representation π of $\mathcal{UA}$ into bounded operators on a Hilbert space $\mathcal{H}$ and state $\psi\in\mathcal{H}$ satisfying

$$(\pi(g_t) - I)\rho = 0 \tag{2.2.41}$$

for all $1\leq t\leq T$. If this condition is satisfied then the representation π and state ρ give the perfect commuting operator strategy S.

Proof. Similarly to the proof of Theorem 2.2.2 we note the strategy S is perfect iff

$$\sum_{t=1}^{T} \frac{1}{\nu_t} \operatorname{Tr}[\pi(g_t)\rho] = 1 \tag{2.2.42}$$

$$\implies \operatorname{Tr}[\pi(g_t)\rho] = 1 \; \forall \; 1 \leq t \leq T \tag{2.2.43}$$

$$\implies \operatorname{Tr}[(\pi(g_t) - I)\rho] = 0 \; \forall \; 1 \leq t \leq T \tag{2.2.44}$$

$$\implies (\pi(g_t) - I)\rho = 0 \; \forall \; 1 \leq t \leq T \tag{2.2.45}$$

where we used on the second line that

$$\operatorname{Tr}[\pi(g_t)\rho] \leq \|\pi(g_t^*)\| \operatorname{Tr}[|\rho|] \leq 1 \tag{2.2.46}$$

by Holder's inequality, with $\|\|$ denoting the operator norm, and that for any vector $x \in \mathcal{H}$

$$\Re[x^*(\pi(g_t) - I)x] = \Re[x^*(\pi(g_t)x] - x^*x \leq 0 \tag{2.2.47}$$

with equality iff $(\pi(g_t) - I)x = x$ on the last line. □

2.3 NullSS for Perfect Nonlocal Games

In this section we show nonlocal games with perfect commuting operator strategies can be studied using NullSS. Section 2.3.1 gives a quick overview of commutative and noncommutative NullSS. Section 2.3.2 introduces a noncommutative NullSS which applies naturally to nonlocal games with perfect commuting operator strategies. Finally Section 2.3.3 connects the noncommutative NullSS of the previous section with the algebraic language introduced in Section 2.2 and gives an explicit "algberaic" characterization of nonlocal games with perfect commuting operator strategies.

2.3.1 Background on NullSS

Hilbert's NullSS

We begin our discussion of NullSS with Hilbert's NullSS; a foundational result in (commutative) algebraic geometry. The statement is the following

Theorem 2.3.1 (Hilbert's NullSS)**.** *Let K be an algebraically closed field, and $K[X_1, X_2, ..., X_d]$ be the commutative algebra of d-variate polynomials over K. Let $P = \{p_1, ..., p_T\}$ be a set of polynomials in the algebra, and let q be some other polynomial in the algebra. Then the following are equivalent:*

1. *$q(\vec{x}) = 0$ for all $\vec{x} \in K^d$ with $p_1(\vec{x}) = p_2(\vec{x}) = ... = p_T(\vec{x}) = 0$.*
2. *There exists an integer r with $q^r \in I(P)$, where $I(P)$ is the ideal of the algebra $K[X_1, X_2, ..., X_d]$ generated by the set of polynomials P.*

Note that we can write the condition $q^r \in I(P)$ more concretely as $q^r = \sum_{i=1}^{T} b_i p_i$, where $b_1, ..., b_T$ are polynomials in the algebra $K[X_1, ..., X_d]$. From this characterization it is clear that condition 2 implies condition 1. Hilbert's NullSS shows that this implication is actually bidirectional.

A useful corollary of Hilbert's NullSS is the following result, sometimes called the weak NullSS.

Corollary 2.3.2. *Let K be an algebraically closed field, and $K[X_1, X_2, ..., X_d]$ be the commutative algebra of d-variate polynomials over K. Let $P = \{p_1, ..., p_T\}$ be a set of polynomials in the algebra. Then the following are equivalent:*

1. *There exists a $\vec{x} \in K^d$ with $p_1(\vec{x}) = p_2(\vec{x}) = ... = p_T(\vec{x}) = 0$.*
2. *$1 \notin I(P)$, where $I(P)$ is the ideal of the algebra $K[X_1, X_2, ..., X_d]$ generated by the set of polynomials P.*

Proof. This result follows from Hilbert's NullSS with $q = 1$. If there is no vector $\vec{x}$ with $p_1(\vec{x}) = p_2(\vec{x}) = ... = p_T(\vec{x}) = 0$ then the polynomial $q = 1$ vanishes on every vector $\vec{x}$ all polynomials in the set P vanish, hence $1 \in I(P)$. On the other hand if $1 \in I(P)$ then it is clear that we cannot have a vector $\vec{x}$ with $p_1(\vec{x}) = p_2(\vec{x}) = ... = p_T(\vec{x}) = 0$. □

Corollary 2.3.2 gives an algebraic criterion which can be used to check whether a system of polynomial equations has a solution. Next, we discuss noncommutative NullSS, which can be used to check similar criterion for systems of polynomial equations with noncommuting variables.

Noncommutative NullSS

The first question encountered when generalizing NullSS to a noncommutative setting is what constitutes a zero in the noncommutative setting. In general, there are 3 definitions of zeros that are sometimes used:

1. **Hard Zeros**, where we have that a nc polynomial $p = 0$ identically.
2. **Directional Zeros**, where we have that a representation of an nc polynomail p vanishes when acting on a state, or $\pi(p)\psi = 0$ for a representation π of p into $\mathcal{B}(\mathcal{H})$ and some state $\psi \in \mathcal{H}$, with $\mathcal{H}$ a Hilbert space.
3. **Determinental Zeros**, where we have that $\det(p) = 0$ for some nc polynomial p.

Here we focus on directional zeros, which correspond naturally to the eigenvalue equations encountered when considering nonlocal games.

The simplest NullSS in this setting concerns real nc polynomials in a free algebra. Here we have the following "perfect" NullSS.

Theorem 2.3.3 (Free Algebra Directional Zeroes NullSS [33]). *Let $\mathcal{F} = \mathbb{R}\langle x_1, x_2, ..., x_n\rangle$ be an algebra over the reals with free nc variables $x_1, x_2, ..., x_n$. Let $q \in \mathcal{F}$ be a polynomial, and $p \in \mathcal{F}$ be an analytic polynomial. Then the following are equivalent:*

1. *For all representations π mapping $\mathcal{F}$ into matrices and real vectors v of the appropriate dimension we have $\pi(q)v = 0$ whenever $\pi(p)v = 0$*
2. *$q \in L\mathcal{I}(p)$, where $L\mathcal{I}(p)$ is in the left ideal of $\mathcal{F}$ generated by the polynomial p.*

With a little work the polynomial p in this nullSS can be upgraded to a set of polynomials. Then, by setting $q = 1$ this nullSS can be used to characterize when a system of equations has no matrix eigenvalue solution.

But this "simple" nc NullSS does not apply to the nonlocal games case for at least two reasons. Firstly, this nullSS only applies to polynomials in the free algebra, while the algebra $\mathcal{UA}$ corresponding to nonlocal game observables is manifestly not free. Secondly, and releatedly, this nullSS only considers matrix solutions, while we must consider infinite dimensional operator solutions to address all commuting operator strategies.

In the next section we present a more sophisticated nc NullSS which does apply to the nonlocal games situation.

2.3.2 A general noncommutative NullSS

Let $\mathcal{A}$ be a complex finitely presented pre C^* algebra. Let $\mathcal{I}$ (resp. $L\mathcal{I}$) denote an (resp. left ideal) in $\mathcal{A}$. A positive *-representation π of $\mathcal{A}$ is *-preserving and maps the SoS cone into positive operators. SOS_M denotes all sums of u^*u with $u \in M$.

Theorem 5.1 and Corollary 5.4 in [9], see [9, 10] say

Theorem 2.3.4. *Suppose that $\{p_\lambda\}_{\lambda\in\Lambda}$ is a subset of $\mathcal{A}$. If $a \in \mathcal{A}$ satisfies $\pi(a)v = 0$ for every hilbert space representation π such that $\pi(p_\lambda)v = 0$ for all $\lambda \in \Lambda$, then*

1.

$$-a^*a \in clos[SOS_{\mathcal{A}} - SOS_{L\mathcal{I}}] \tag{2.3.1}$$

The converse is also true, provided this holds without the closure.

2.

$$-a^*a \in SOS_{\mathcal{A}} - clos[cone(S)] \tag{2.3.2}$$

where $S := \{p_\lambda\}_{\lambda\in\Lambda}$.

3.

$$-a^*a \in clos[SOS_{\mathcal{A}} + L\mathcal{I} + L\mathcal{I}^*] \tag{2.3.3}$$

In these assertions when $\mathcal{A}$ is the free algebra and $L\mathcal{I}$ is finitely generated, representations π into finite dimensional Hilbert spaces will suffice to imply these algebraic certificates.

We do not define closure here, since soon we will not need it for the $a = 1$ case.

Proof of Theorem 2.3.4 Here we give a proof which ties the parts of the theorem to corresponding theorems in [9]. This is unintuitive so in Section Section 2.3.2 we sketch the idea of the proof.

The forward side is in [9], (1) is Theorem 5.1.

(2) is Corollary 5.4 and does require closure for $a = 1$. It just applies the old

$$p^*q + q^*p = (p + tq)^*(p + tq)/t - tq^*q - p^*p/t$$

trick to get

$$-a^*a + tq^*q = SOS_{\mathcal{A}} + (p + tq)^*(p + tq)/t - p^*p/t.$$

Then $t \to 0$ stays in the closure of the right side.

The converse is so trivial and is not even stated in [9], so we prove it now. Suppose the certificate holds and $\pi(p_\lambda)v = 0$ for all λ. Then *sum of* $(v^*\pi(S)^*\pi(S)v) = 0$, so

$$-v^*\pi(a)^*\pi(a)v = v^*\pi(SOS_{\mathcal{A}})v \tag{2.3.4}$$

forcing both sides to be 0; thus $\pi(p_\lambda)v = 0$ for all λ forces $\pi(a)v = 0$.

(3) Combines Corollary 5.3 and Lemma 5.5 of [9].

The finite dimensional assertion is proved constructively and this is the major part of [9]. □

Intuition behind the proof of Theorem 2.3.4

Here is a special case of Theorem 2.3.4, included here since a sketch of its proof is supplies the readers intuition. Also this lesser level of generality is all that is needed here for nonlocal games.

Theorem 2.3.5. *Suppose $\mathcal{A}$ is a complex finitely presented C^* algebra with generators $\{x_j\}$, set of relations Γ, and norm denoted $\|\cdot\|$ and $L\mathcal{I}$ is a finitely generated left ideal in $\mathcal{A}$. The the following are equivalent*

1. *There exists a Hilbert space $\mathcal{H}$, a positive *-representation π of $\mathcal{A}$ into $B(\mathcal{H})$ and $\psi \in \mathcal{H}$ satisfying $\pi(f)\psi = 0$ for all $f \in L\mathcal{I}$.*

2. $-1 \notin SOS_{\mathcal{A}} + L\mathcal{I} + L\mathcal{I}^*$.

Proof (Sketch – full details in [9]). Easy side: suppose $-1 \in SOS_{\mathcal{A}} + L\mathcal{I} + L\mathcal{I}^*$. If π, ψ satisfying condition 1 exist, then $\langle\psi|\pi(-1)|\psi\rangle = \langle\psi|-1|\psi\rangle \geq 0$; contradiction.

Harder side: Let $\mathcal{S}$ denote the span of 1 and $L\mathcal{I} + L\mathcal{I}^*$. By Hahn Banach (fancy form Eidelheit Kakutani) there is a continuous linear functional $\widehat{L} : \mathcal{S} + SOS_{\mathcal{A}} \to \mathbb{C}$ satisfying

$$\widehat{L}(-1) = -1 \qquad \widehat{L}(L\mathcal{I}) \geq 0 \quad \hat{L}(SOS_{\mathcal{A}}) \geq 0 \tag{2.3.5}$$

Since $L\mathcal{I}$ is a subspace $\widehat{L}(L\mathcal{I}) = 0$ (else multiplying by -1 would produce an element in $L\mathcal{I}$ with $\widehat{L}$ negative). Also by construction $\widehat{L}(f) = f(0) \geq 0$ for any f which is a sum of hermetian squares in $\mathcal{S}$. (this is why we add a SOS.) That is, $\widehat{L}$ is a positive linear functional, hence by the Krein Extenson Theorem it has a positive linear extension L to $\mathcal{A}$.

Now perform the GNS construction. Define the bilinear form

$$(a, b) := L(a^*b) \tag{2.3.6}$$

on $\mathcal{A}$. Set $N := \{a|L(a^*a)\} = 0$ and because

$$0 \leq L(a^*r^*ra) \leq L(a^*a)L(a^*r^*rr^*ra) = 0 \tag{2.3.7}$$

by Cauchy-Schwarz we see N is a left ideal; hence $L(N) = 0$, hence $N \neq \mathcal{A}$. Since

$$L\mathcal{I}^*L\mathcal{I} \subset L\mathcal{I}, \tag{2.3.8}$$

we have $L\mathcal{I} \subset N$.

Now consider the quotient space $\mathcal{A}/N$ which we identify with a Hilbert space $\hat{\mathcal{H}}$ with norm induced by the inner product defined above. Also define the quotient map $\phi : \mathcal{A} \to \hat{\mathcal{H}}$,

$$\phi(a) := a + N \tag{2.3.9}$$

and take $\psi := \phi(1)$. Let π be the standard GNS $*-$representation of $\mathcal{A}$, so $\pi(x_i)\phi(x_j) :=$

$\phi(x_i x_j)$. This satisfies

$$(\pi(f)\psi, \pi(g)\psi) = L(f^* g) \tag{2.3.10}$$

which implies $\pi(f)\psi = 0$ for all $f \in N$. Hence $\pi(f)\psi = 0$. Noting GNS construction maps into bounded operators (again, for details see [9]), we are done. □

2.3.3 NullSS and Perfect Games

Combining the NullSS of Section 2.3.2 with the perfect game condition discussed in Section 2.2.4 gives a new characterization of games with perfect commuting operator strategies in terms of left ideals and sums of squares of the universal game algebra $\mathcal{UA}$.

We begin by defining some left ideals of $\mathcal{UA}$ which build on the valid and invalid response sets discussed in Section 2.2.4. First, for any game $\mathcal{G}$ with question set $\mathcal{Q}$ and valid responses $\mathcal{V}(\vec{i})$ define the subset of valid vanishing polynomials $\overline{\mathcal{V}}$ by

$$\overline{\mathcal{V}} := \left\{ \sum_{\vec{a} \in \mathcal{V}(\vec{i})} \prod_{\alpha} e(\alpha)_{a(\alpha)}^{i(\alpha)} - 1 \right\}_{\vec{i} \in \mathcal{Q}}. \tag{2.3.11}$$

These are polynomials of the form $y - 1$ for all $y \in \mathcal{V}(\vec{i})$ with $\vec{i} \in \mathcal{Q}$. Similarly, define the invalid vanishing polynomials $\mathcal{N}$ by

$$\mathcal{N} := \left\{ \prod_{\alpha} e(\alpha)_{a(\alpha)}^{i(\alpha)} \right\}_{(\vec{i},\vec{a}) \in (\mathcal{Q}, \mathcal{N}(\vec{i}))}. \tag{2.3.12}$$

Note both $\overline{\mathcal{V}}$ and $\mathcal{N}$ contain sets of polynomials in $\mathcal{UA}$ which correspond to polynomials of projectors which must vanish on ρ in any perfect commuting operator strategy. This motivates the following result.

Theorem 2.3.6. *Let $\mathcal{G}$ be a a nonlocal game, and $\overline{\mathcal{V}}, \mathcal{N}$ be the valid and invalid vanishing polynomails associated with the game. Then the following are equivalent:*

1. $\omega_{co}^*(\mathcal{G}) = 1$

2. $-1 \notin L\mathcal{I}\left(\overline{\mathcal{Y}}\right) + L\mathcal{I}\left(\overline{\mathcal{Y}}\right)^* + SOS_{\mathcal{UA}}$

3. $-1 \notin L\mathcal{I}\left(\mathcal{N}\right) + L\mathcal{I}\left(\mathcal{N}\right)^* + SOS_{\mathcal{UA}}$

Proof. Immediate from Theorems 2.2.2 and 2.3.5, plus the view of strategies as representations introduced in Section 2.2.2. □

This theorem apples to all games (according to the definitions here) and characterizes which games do vs. do not have a quantum strategy. Unfortunately, the freedom given by the SOS terms in this algebraic certificate can be hard to use. Hence, we turn next to situations with no SOS term.

2.4 NullSS without SOS and Subgroup Membership

It is helpful to divide perfect game condition into two sub questions. The first is checking whether $-1 \in L\mathcal{I} + L\mathcal{I}^*$ that is, whether

$$1 \in L\mathcal{I} + L\mathcal{I}^*.$$

Intuitively, this question feels "algebraic", and we will show in Section 2.4.2 that in special cases it reduces to the subgroup membership problem.

The second problem is checking whether

$$-1 \in L\mathcal{I} + L\mathcal{I}^* + SOS_{\mathcal{UA}} \tag{2.4.1}$$

given that

$$1 \notin L\mathcal{I} + L\mathcal{I}^*. \tag{2.4.2}$$

This question is more analytic, and adds substantial complexity to applications. In special cases we have that

$$-1 \notin L\mathcal{I} + L\mathcal{I}^* \implies -1 \notin L\mathcal{I} + L\mathcal{I}^* + SOS_{\mathcal{UA}} \tag{2.4.3}$$

and hence the second problem is trivial. This seems closely related to the existence of projections which are conditional expectations and respect SOS. The next section investigates this link further.

2.4.1 Conditional Expectations and SOS Projections

Now we give simplifications of our NullSS which use the existence of either SOS projections or SOS conditional expectations. The term "SOS" projection is introduced here (though the concept is certainly standard), while the term conditional expectation is standard and a definition can be found, for example, in [56]. We repeat these definitions here.

Definition 2.4.1 (SOS Projection)**.** *Given a *-algebra $\mathcal{A}$ and a *-subalgebra $\mathcal{C}$ an SOS-projection $P : \mathcal{A} \to \mathcal{C}$ is a projection (i.e. $P^2 = P$) satisfying the additional property that $P(SOS_{\mathcal{A}}) \subseteq SOS_{\mathcal{C}}$, that is, that sums of squares in $\mathcal{A}$ are mapped to sums of squares in the subalgebra.*

Definition 2.4.2 (Conditional Expectation)**.** *Given a unital *-algebra $\mathcal{A}$ and a unital *-subalgebra $\mathcal{C}$ a linear map $p : \mathcal{A} \to \mathcal{C}$ is called a conditional expectation if it satisfies*

1. *$p(a)^* = p(a^*)$ for all $a \in \mathcal{A}$.*
2. *$p(b_1 a b_2) = b_1 p(a) b_2$ for all $a \in \mathcal{A}$, $b_1, b_2 \in \mathcal{C}$.*
3. *$p(1_{\mathcal{A}}) = p(1_{\mathcal{B}})$.*
4. *$p(SOS_{\mathcal{A}}) \subseteq SOS_{\mathcal{A}} \cap \mathcal{C}$.*

Definition 2.4.3 (SOS Conditional Expectation)**.** *Given a unital *-algebra $\mathcal{A}$ and a unital *-subalgebra $\mathcal{C}$ a SOS conditional expectation (called a strong conditional expectation in [56]) is a conditional expectation that also satisfies the SOS projection property, so*

1. *$p(SOS_{\mathcal{A}}) \subseteq SOS_{\mathcal{C}}$.*

We now show existence of these mappings can simplify the nonlocal games NullSS.

Lemma 2.4.4. *$-1 \notin L\mathcal{I} + L\mathcal{I}^* + SOS_{\mathcal{UA}}$ iff there exists a subalgebra $\mathcal{C} \subseteq \mathcal{UA}$ with*

1. *An SOS projection* $P : \mathcal{UA} \to \mathcal{C}$ *and*

2. $1 \in \mathcal{C}$ *and* $L\mathcal{I} + L\mathcal{I}^* \in \mathcal{C}$ *and*

3. $-1 \notin L\mathcal{I} + L\mathcal{I}^* + SOS_{\mathcal{C}}$.

Proof. Taking $\mathcal{C} = \mathcal{UA}$ and $P = I$ makes the only if direction trivial.

To see the other direction, assume existence of a subalgebra $\mathcal{C}$ satisfying the conditions of the theorem. Then assume for contradiction that $-1 \in L\mathcal{I} + L\mathcal{I}^* + SOS_{\mathcal{UA}}$. Applying the SOS projection P to both sides of this equation gives

$$P(-1) \in P(L\mathcal{I} + L\mathcal{I}^* + SOS_{\mathcal{UA}}) \tag{2.4.4}$$

$$\implies -1 \in L\mathcal{I} + L\mathcal{I}^* + SOS_{\mathcal{C}}, \tag{2.4.5}$$

a contradiction. □

Possibly interesting is that Lemma 2.4.4 only requires the SOS projection property, not the conditional expectation property, making it quite general. On the other hand, the requirement that $L\mathcal{I} + L\mathcal{I}^* \subseteq \mathcal{C}$ makes it tricky to come up with a useful subalgebra $\mathcal{C}$. The next theorem loosens the $L\mathcal{I} + L\mathcal{I}^*$ condition by upgrading the SOS projection to a conditional expectation.

Lemma 2.4.5. *Let* F *denote a list of nc polynomials generating the left ideal* $L\mathcal{I}$ *in* $\mathcal{UA}$. *Given a subalgebra* $\mathcal{C} \subseteq \mathcal{UA}$ *with* $F \subseteq \mathcal{C}$, *let* $L\mathcal{I}_{\mathcal{C}}$ *denote the left ideal of* $\mathcal{C}$ *generated by* F. *Then* $-1 \notin L\mathcal{I} + L\mathcal{I}^* + SOS_{\mathcal{UA}}$ *iff there exists a subalgebra* $\mathcal{C} \subseteq \mathcal{UA}$ *with*

1. *An SOS conditional expectation* $P : \mathcal{UA} \to \mathcal{C}$ *and*

2. $1 \in \mathcal{C}$ *and* $F \subseteq \mathcal{C}$ *and*

3. $-1 \notin L\mathcal{I}_{\mathcal{C}} + L\mathcal{I}_{\mathcal{C}}^* + SOS_{\mathcal{C}}$.

Proof. As with Lemma 2.4.4 taking $\mathcal{C} = \mathcal{UA}$, $P = I$ makes one direction trivial.

To see the other first note that any polynomial $p \in L\mathcal{I}$ can we written

$$p = \sum_i a_i b_i \tag{2.4.6}$$

with $b_i \in F$ and a_i arbitrary. Then, using that P is left promodular gives that

$$P(p) = \sum_i P(a_i b_i) \tag{2.4.7}$$

$$= \sum_i P(a_i) b_i \in L\mathcal{I}_{\mathcal{C}}. \tag{2.4.8}$$

We conclude that $P(L\mathcal{I}) = L\mathcal{I}_{\mathcal{C}}$. Then, to prove the theorem assume for contradiction that there exists a subalgebra $\mathcal{C}$ satsifying the conditions of the theorem and that $-1 \in L\mathcal{I} + L\mathcal{I}^* + SOS_{\mathcal{UA}}$. Then

$$P(-1) \in P(L\mathcal{I} + L\mathcal{I}^* + SOS_{\mathcal{UA}}) \tag{2.4.9}$$

$$\implies -1 \in L\mathcal{I}_{\mathcal{C}} + L\mathcal{I}_{\mathcal{C}}^* + SOS_{\mathcal{C}}, \tag{2.4.10}$$

a contradiction. □

Now we do a preperation step for finding a subalgebra $\mathcal{C}$ that makes Lemma 2.4.5 valuable. We consider the subalgebra generated by $\{F, 1\}$. This is the smallest possible subalgebra which satisfies Condition 2. We show it also satisfies Condition 3 provided 1 is not in the subalgebra generated by F.

Lemma 2.4.6. *Let $\mathcal{A}$ be a C^*-algebra and F be a set of polynomials in $\mathcal{A}$. Also let $\mathcal{C} = \mathbb{C}\langle\{F, 1\}\rangle$ be the subalgebra of $\mathcal{A}$ generated by the set of polynomials $\{F, 1\}$. If*

$$1 \notin \mathbb{C}\langle F\rangle \tag{2.4.11}$$

then

$$-1 \notin L\mathcal{I}_{\mathcal{C}} + L\mathcal{I}_{\mathcal{C}}^* + SOS_{\mathcal{C}} \tag{2.4.12}$$

Proof. First note that any polynomial $p \in \mathcal{C}\langle F, 1\rangle$ can be written as

$$p = p_F + \alpha \tag{2.4.13}$$

where $\alpha \in \mathbb{C}$ and p_F is a polynomial in the elements from $\mathbb{C}\langle F\rangle$. It follows that any polynomial $p' \in L\mathcal{I}_\mathcal{C}$ can be written as

$$p' = (p_F + \alpha)\, f = p'_F \in \mathbb{C}\langle F\rangle \tag{2.4.14}$$

with $f \in F$. A similar result holds for any polynomial in $L\mathcal{I}_\mathcal{C}^*$. Additionally, any polynomial $p'' \in SOS_\mathcal{C}$ can be written as

$$\begin{aligned} p'' &= \sum_i (p_{F,i} + \alpha_i)(p_{F,i} + \alpha_i)^* && (2.4.15)\\ &= \sum_i \left(p_{F,i}^2 + \alpha_i^* p_{F,i} + \alpha_i p_{F,i}^*\right) + \sum_i \alpha_i^2 && (2.4.16)\\ &= p''_F + \alpha'' && (2.4.17) \end{aligned}$$

with each $p_{F,i} \in \mathbb{C}\langle F\rangle$ hence p''_F in $\mathbb{C}\langle F\rangle$ and $\alpha'' > 0 \in \mathbb{C}$.

Now assume for contradiction that $-1 \in L\mathcal{I}_\mathcal{C} + L\mathcal{I}_\mathcal{C}^* + SOS_\mathcal{C}$. Then, combining the above observations we can write

$$-1 = p'_F + p''_F + \alpha'' \tag{2.4.18}$$

with $p'_F, p''_F \in \mathbb{C}\langle F\rangle$ and $\alpha'' > 0 \in \mathbb{C}$. Rearranging the above expression gives

$$\begin{aligned} -(1 + \alpha'') &= p_F + p''_F \in \mathbb{C}\langle F\rangle && (2.4.19)\\ \implies 1 &\in \mathbb{C}\langle F\rangle && (2.4.20) \end{aligned}$$

where we used that $\alpha'' > 0$ and so $1 + \alpha'' \neq 0$.

□

Combining Lemmas 2.4.5 and 2.4.6 results in the following simplified NullSS.

Theorem 2.4.7. *Let $\mathcal{A}$ be a C^*-algebra and F be a set of polynomials in $\mathcal{A}$. Also let $\mathcal{C} = \mathbb{C}\langle\{F, 1\}\rangle$ be the subalgebra of $\mathcal{A}$ generated by the set of polynomials $\{F, 1\}$. If there exists an SOS conditional expectation mapping $\mathcal{A}$ onto $\mathcal{C}$ then the following are equivalent.*

1. *There exists a Hilbert space representation* $\pi : \mathcal{A} \to B(\mathcal{H})$ *and vector* $v \in \mathcal{H}$ *with* $\pi(f)v = 0$ *for all* $f \in F$.

2. $-1 \notin L\mathcal{I}(F)_{\mathcal{A}} + L\mathcal{I}(F)^*_{\mathcal{A}} + SOS_{\mathcal{A}}$

3. $-1 \notin L\mathcal{I}(F)_{\mathcal{A}} + L\mathcal{I}(F)^*_{\mathcal{A}}$

4. $1 \notin L\mathcal{I}(F)_{\mathcal{C}} + L\mathcal{I}(F)^*_{\mathcal{C}}$

5. $1 \notin \mathbb{C}\langle F \rangle$

Proof. We have $1 \Leftrightarrow 2$ by Theorem 2.3.5, $2 \Rightarrow 3 \Rightarrow 4$ by set inclusion, and $4 \Rightarrow 2$ by Lemma 2.4.5. Finally, we have $4 \Leftrightarrow 5$ by Lemma 2.4.6. □

2.4.2 The NC Toric Ideal Group Algebra Simplification

We now give a further simplification of the perfect games NullSS in the case where $\mathcal{A} = \mathbb{C}[G]$ is a group algebra and F is a set of polynomials in $\mathcal{A}$ of the form (monomial $-$ 1). The ideal generated by such a set of polynomials is called a (nc) toric ideal.

Note. Here, and everywhere in this paper, we define a monomial to be any element of the form βg with $\beta \in \mathbb{C}$ and $g \in G$.[1]

Our first observation is that there is a natural SOS conditional expectation mapping from $\mathcal{A}$ to $\mathbb{C}\langle \{F, 1\} \rangle$ provided that F is of this form.

Theorem 2.4.8. *Let* $\mathcal{A} = \mathbb{C}[G]$ *be a group algebra and* F *be a set of polynomials in* $\mathcal{A}$ *with each* $f_t \in F$ *of the form* $r_t - 1$ *with* r_t *a monomial. Then there is an SOS conditional expectation mapping* $\mathcal{A}$ *onto* $\mathbb{C}\langle \{F, 1\} \rangle$.

Proof. Define the subgroup G' of G to consist of all $g \in G$ with $\beta g = r_t$ for some $\beta \in \mathbb{C}$ and $r_t - 1 \in F$. Then $\mathbb{C}\langle \{F, 1\} \rangle = \mathbb{C}\langle \{\{r_t\}, 1\} \rangle = \mathbb{C}[G']$. Since $G' < G$ there exists an SOS conditional expectation mapping $\mathbb{C}[G]$ onto $\mathbb{C}[G']$ by Example 5 of [56]. □

Before stating this simplified NullSS we need one final theorem simplifying the subalgebra memebership problem for toric subalgebras of group algebras.

[1] An alternate definition of monomial, **which we do not use**, would be to define a monomial to be any element of the form $g \in G$, without the prefactor.

Relating the Subalgebra and Subgroup Membership Problems

Our starting point is the following lemma, which gives a standard form for polynomials in $\mathbb{C}\langle F\rangle$.

Lemma 2.4.9. *Let $\mathcal{A} = \mathbb{C}[G]$ be a group algebra and F be a set of polynomials in $\mathcal{A}$ with each $f_t \in F$ of the form $f_t = r_t - 1$ with r_t a monomial. Finally let $R = \langle\{r_t\}\rangle$ denote the group generated by multiplication of the monomials $\{r_t\}$ appearing in the set of polynomials F (and their inverses). Then any polynomial $p \in \mathbb{C}\langle F\rangle$ can be written in the form*

$$p = \sum_{u,v} \beta_{u,v}(u - v) \tag{2.4.21}$$

with $u, v \in R$ and $\beta_{u,v} \in \mathbb{C}$.

Proof. First note that writing any polynomial $p' \in \mathbb{C}\langle F, 1\rangle$ as a sum of monomials gives that p' can be written in the form

$$p' = \sum_{u} \beta_v v \tag{2.4.22}$$

with $u \in R$. Also note that any $f \in F$ can be written in the form $(u' - 1)$ with $u' \in R$ by definition. These two observations, combined with the fact that any polynomial $p \in \mathbb{C}\langle F\rangle$ can be written in the form

$$p = \sum_{f} p_f f \tag{2.4.23}$$

with $f \in F$ and $p_f \in \mathbb{C}\langle F, 1\rangle$ proves the result. (To complete the proof, simply apply both observations, multiply the resulting polynomials together and relabel $vu' = u$, then collect like terms). □

Theorem 2.4.10. *Let $\mathcal{A} = \mathbb{C}[G]$ be a group algebra and F be a set of polynomials in $\mathcal{A}$ with each $f_t \in F$ of the form $f_t = r_t - 1$ with r_t a monomial. Finally let $R = \langle\{r_t\}\rangle$ denote the group generated by multiplication of the monomials $\{r_t\}$ appearing in the set of polynomials F (and their inverses). Then $1 \in \mathbb{C}\langle F\rangle$ iff $R \cap \mathbb{C} \supsetneq \{1\}$, i.e. $\xi \notin R$ for all $\xi \in \mathbb{C}$ with $\xi \neq 1$.*

Proof. This proof is partially inspired by the proof of Theorem 2.7 in [43]. First note that for any monomials r_t and r_s we have

$$(r_t - 1)(r_s - 1) + (r_t - 1) + (r_s - 1) = r_t r_s - 1. \tag{2.4.24}$$

It follows that for any $t \in \langle\{r_t\}\rangle$ we have $t - 1 \in \mathbb{C}\langle F\rangle$ (note inverses are contained automatically since $\mathbb{C}\langle F\rangle$ is a $*$-algebra and $(t-1)^* = t^* - 1 = t^{-1} - 1$). Then, if $\beta \in \langle\{r_t\}\rangle$ for some $\beta \neq 1 \in \mathbb{C}$ we have

$$\beta - 1 \in \mathbb{C}\langle F\rangle \Leftrightarrow 1 \in \mathbb{C}\langle F\rangle . \tag{2.4.25}$$

This completes the proof in one direction.

To prove the result in the other direction we assume for contradiction that $\xi \notin H$ for all $\xi \neq 1 \in \mathbb{C}$ and that $1 \in \mathbb{C}\langle F\rangle$. Then, using Lemma 2.4.9, we can write

$$1 = \sum_{u,v} \beta_{u,v}(u - v) \tag{2.4.26}$$

$$= \sum_{u,v} u\left(\beta_{u,v} - \beta_{v,u}\right) \tag{2.4.27}$$

where we relabeled u and v in the second term in the sum on the second line. Now since $\beta \notin R$ we have, for any terms $u \neq u'$ in the sum above, that $u(u')^{-1} \notin \mathbb{C}$ (i.e. the terms differ by multiplication of some non-constant element of H). Then there can be no cancellation between different u and u' terms, and we see that

$$\sum_v (\beta_{1,v} - \beta_{v,1}) = 1 \tag{2.4.28}$$

and

$$\sum_v (\beta_{u,v} - \beta_{v,u}) = 0 \tag{2.4.29}$$

for all $u \neq 1$. But this is a contradiction, since

$$\sum_v (\beta_{1,v} - \beta_{v,1}) + \sum_{u \neq 1} \sum_v (\beta_{u,v} - \beta_{v,u}) = \sum_{u,v} (\beta_{u,v} - \beta_{v,u}) = 0 \tag{2.4.30}$$

□

NC Toric Left NullSS without SOS Terms

Now we combine the results in Sections 2.4.1 and 2.4.2 to obtain the following specialized NullSS. We point out that this NullSS generalizes Theorem 2.7 in [43].

Theorem 2.4.11. *Let $\mathcal{A} = \mathbb{C}[G]$ be a group algebra and F be a set of monomials in $\mathcal{A}$ with each $f_t \in F$ of the form $f_t = r_t - 1$ with r_t a monomial. Also let $\mathcal{C} = \mathbb{C}\langle\{F, 1\}\rangle$ be the subalgebra of $\mathcal{A}$ generated by the set of polynomials $\{F, 1\}$. Finally let $R = \langle\{r_t\}\rangle$ denote the group generated by multiplication of the monomials $\{r_t\}$ appearing in the set of polynomials F (and their inverses). Then the following are equivalent:*

1. *There exists a Hilbert space representation $\pi : \mathcal{A} \to B(\mathcal{H})$ and vector $v \in \mathcal{H}$ with $\pi(f)v = 0$ for all $f \in F$.*
2. *$-1 \notin L\mathcal{I}(F)_{\mathcal{A}} + L\mathcal{I}(F)^*_{\mathcal{A}} + SOS_{\mathcal{A}}$*
3. *$-1 \notin L\mathcal{I}(F)_{\mathcal{A}} + L\mathcal{I}(F)^*_{\mathcal{A}}$*
4. *$1 \notin L\mathcal{I}(F)_{\mathcal{C}} + L\mathcal{I}(F)^*_{\mathcal{C}}$*
5. *$1 \notin \mathbb{C}\langle F\rangle$*
6. *$R \cap \mathbb{C} \supsetneq \{1\}$.*

Proof. Items 1 through 5 are equivalent by Theorems 2.4.7 and 2.4.8. Items 5 and 6 are equivalent by Theorem 2.4.10. □

2.4.3 NullSS for Perfect Unitary Games

We now apply the simplified NullSS of this section to nonlocal games. We first define a class of games on which the NullSS naturally applies.

Definition 2.4.12. *A game $\mathcal{G}$ is said to be a unitary game if there exists a change of variables under which we have $\mathcal{UA} = \mathbb{C}[G]$ a group algebra for some group G and*

$$\Phi_{\mathcal{G}} = \sum_{t=1}^{T} \frac{1}{\nu_t} h_t \tag{2.4.31}$$

with T an integer, each ν_t a positive real number with $\sum_t \nu_t = 1$, and each h_t a monomial in $\mathbb{C}[G]$ (Recall that we are allowing monomials to include constant prefactors, so this means each h_t is of the form $\beta_t g_t$ for some $\beta \in \mathbb{C}$ and $g_t \in G$). The monomials h_t are called the clauses of the game $\mathcal{G}$.

Then the following theorem is a quick consequence of our NullSS and Theorem 2.2.3.

Theorem 2.4.13. *Let $\mathcal{G}$ be a unitary game with a set of clauses $\{h_t\}$. Let $H = \langle \{h_t\} \rangle$ be the subgroup generated by multiplication of the clauses and their inverses. Then $\mathcal{G}$ has a perfect commuting operator strategy iff $\xi \notin H$ for all $\xi \neq 1 \in \mathbb{C}$.*

Proof. By Theorem 2.2.3 $\mathcal{G}$ has a perfect commuting operator strategy iff there exists a representation π mapping $\mathcal{UA}$ into bounded operators on a Hilbert space $\mathcal{H}$ and a state ρ in $\mathcal{H}$ with $\pi(h_t - 1)\rho = 0$ for all clauses h_t. By Theorem 2.4.11 this happens iff $\xi \notin H$ for all $\xi \neq 1 \in \mathbb{C}$. □

Example 2.4.14. From Section 2.2.3 we have that XOR games have a game polynomial of the form

$$\Phi_{\mathcal{G}} = \Phi_{\mathcal{G}} = \frac{1}{2} + \frac{1}{2T} \sum_{t=1}^{T} (-1)^{s_t} \prod_{\alpha \in [k]} x^{(\alpha)}_{\vec{i}_t(\alpha)}, \tag{2.4.32}$$

with the $x^{(\alpha)}_{\vec{i}_t(\alpha)}$ cyclic unitaries. We also know that $\mathcal{UA}$ can be viewed as a group algebra generated by $x^{(\alpha)}_{\vec{i}_t(\alpha)}$ satisfying the relations described in Section 2.2.2.

Then, defining clauses

$$h_t = (-1)^{s_t} \prod_{\alpha \in [k]} x^{(\alpha)}_{\vec{i}_t(\alpha)} \tag{2.4.33}$$

for all $t = T$ and noting that the h_t are monomials by definition we have that an XOR game has a perfect commuting operator stategy iff $\xi \notin \langle \{h_t\} \rangle$ for all $\xi \neq 1 \in \mathbb{C}$.

2.5 Chapter Summary

In this chapter we have developed a general algebraic characterization of perfect commuting operator strategies for nonlocal games. In particular, Theorem 2.3.6 gives an algebraic criterion that applies to any nonlocal game and characterizes the existence of a perfect commuting operator strategy. We then showed how to simplify this criterion for special classes of games by borrowing and then building on some results from the study of nc algebras. The end result of this simplification was Theorem 2.4.11, which related existence of a perfect commuting operator strategy to an instance of the subgroup membership problem for class of nonlocal games, which included XOR games as a special example. The result of applying this theorem to XOR games is laid out in Example 2.4.14 above.

In subsequent chapters we will move away from studying general games and algebras, and focus on understanding the subgroup membership characterization of XOR games. In doing so, we will reprove some results from this chapter in the special case of XOR games. The hope is that the proofs provided in this chapter will provide some general context to those specific results and help illuminate the mathematical structure that underlies them.

Chapter 3

Refutations, Symmetric XOR Games, and MERP Strategies

In this chapter we begin the study XOR games with perfect commuting operator value. Using the framework developed in Chapter 2, we know that these games have a perfect commuting operator strategy iff an instance of the subgroup membership problem has a no answer. In this chapter we introduce the concept of a *refutation* – a proof of a yes answer to the subgroup membership problem associated with an XOR game. We then reprove the result of Chapter 2 in the special case of XOR games, showing that an XOR game has perfect commuting operator value iff it does not have a refutation. However, rather than referencing a noncommutative Nullstellensatz, the proof in this chapter references completeness of the ncSoS hierarchy (which is actually built on a noncommutative Positivstellensatz [19, 32]). This lets us connect the length of refutations to the time it takes the ncSoS algorithm to prove a game does not have a perfect commuting operator strategy,[1] We also obtain a weak bound on how far away from 1 the commuting operator value of a game is when a refutation exists.

We then introduce a *parity refutation* (PREF), which is a weaker object than a refutation (meaning that a game with a refutation also has a parity refutation). Importantly, existence

[1] A small disclaimer should be made here for those familiar with the ncSoS algorithm. The bounds we obtain are bounds on the syntactic degree required for ncSoS to prove a game does not have value 1. This only implies a runtime bound if the ncSoS algorithm does not simplify expressions at a semantic degree higher than the current "degree" of the algorithm.

of a parity refutation for a game can be decided in polynomial time, while there is no known algorithm to decide if a refutation exists for an XOR game in general. We show that for a class of games called symmetric games existence of a PREF implies existence of a refutation, meaning that symmetric games with commuting operator value 1 can be identified in polynomial time. Finally, we construct a strategy, called MERP, which we show achieves value 1 on any game that does not have a PREF.

Section 3.3 recaps all the notation required to understand the results of this section, including some notation specific to XOR games which hasn't been introduced previously. In particular, because all games $\mathcal{G}$ introduced in this section are XOR games, we describe them using an object called a *game matrix*, introduced in Definition 3.3.3, instead of using one of the more general descriptions of games introduced in Chapter 1.

3.1 Background

Constraint satisfaction problems (CSPs) are a fundamental object of study in theoretical computer science. In quantum information theory, there are two natural analogues of CSPs, which both play important roles: local Hamiltonians and (our focus) non-local games. Non-local games originate from Bell's pioneering 1964 paper, which showed how to test for quantum entanglement in a device with which we can interact only via classical inputs and outputs. In modern language, the tests developed by Bell are games: a referee presents two or more players with classical questions drawn from some distribution and demands answers from them. Each combination of question and answers receives some score and the players cooperate (but do not communicate) in order to maximize their expected score. These games are interesting because often the players can win the game with a higher probability if they share an entangled quantum state, so a high average score can certify the presence of quantum entanglement. Such tests are not only of scientific interest, but have had wide application to proof systems [13, 35], quantum key distribution [22, 2, 64], delegated computation [54], randomness generation [17] and elsewhere.

To be able to use a nonlocal game as a test for entanglement, it is essential to be able to approximately compute two quantities: the best possible expected score when the players

share either classical correlations or entangled states, respectively called the "classical" and "quantum" (or "entangled") values of the game, and denoted ω and ω^*. To understand these quantities, think of a game with k players as inducing a constraint satisfaction problem with a k-ary predicate. Each question in the game is mapped to a variable in the CSP, and each k-tuple of questions and set of accepted responses (a "clause") asked by the referee corresponds to a constraint. Classically, a simple convexity argument shows that the players can always stick to *deterministic* strategies, where each question is assigned a fixed answer; thus, ω is in fact identical to the value of the CSP. Hence, thanks to various dichotomy theorems, we have a good understanding of the difficulty of computing ω: in some cases, we know a P algorithm, and for most others, we know it is NP-complete.

The quantum value ω^* is not as well understood. The main obstacle is that the set of entangled strategies is very rich: the "assignment" to each variable is no longer a value from a discrete set, but a linear operator over a Hilbert space of potentially unbounded dimension. As a result, we can say very little in terms of upper bounds on the complexity of computing ω^*. In fact, it is not known whether even a constant-factor (additive) approximation to ω^* is Turing-computable. For general games, the best we can say is that it is recursively enumerable: there is an algorithm, called the NPA or ncSoS hierarchy [46, 20], that in the limit of infinite time converges from above to the quantum value, but with no bound on the speed of convergence. On the hardness side, more is known, and what we know is grounds for pessimism: we have been able to show hardness results for approximating ω^* matching (e.g. [65]) and in some cases exceeding (e.g. [35]) the classical case by constructing special games that force entangled players to use particular strategies. Moreover, families of games have been found for which deciding whether $\omega^* = 1$ is uncomputable [59]. There are a few exceptions for which some positive results are known: for instance, the class of XOR games, in which the answers are bits and the payoff depends only on their XOR (for any given set of questions). In the classical case, these games are as hard as general games except in the "perfect completeness" regime: distinguishing $\geq 1 - \varepsilon$ satisfiability from $\leq \frac{1}{2} + \varepsilon$ satisfiability is NP-complete, but we can determine whether an XOR game is perfectly satisfiable in polynomial time using Gaussian elimination over $\mathbb{F}_2$. However, in the quantum case, it was shown by Tsirelon [11, 63] that for two-player XOR games, the lowest level of the ncSoS

algorithm converges exactly to the quantum value, rendering it computable in polynomial time via semidefinite programming. (A similar technique was also applied to approximating the entangled value of unique games [39].) Yet these techniques seemed unlikely to generalize to three or more players: it is known that distinguishing $\geq 1 - \varepsilon$ satisfiability from $\leq \frac{1}{2} + \varepsilon$ for an entangled 3-player XOR game is NP-hard [65], and deciding the existence of perfect strategies for the closely-related family of linear systems games is uncomputable [59].

Another question which has been very fruitful in the study of classical CSPs is understanding the typical value of a random instance. Research in this direction draws significantly on insights from statistical mechanics and has proven that there exist sharp satisfiable/unsatisfiable thresholds for random k-SAT and related games (often using the equivalent constraint-satisfaction formulation). But these techniques do not carry over to the quantum case. For random classical games, a basic method is to look at the expected number of winning responses (the "first moment method") or the variance (the "second moment method") as we randomize the payoff function within some family such as random k-SAT or random k-XOR. This suffices, for example, to show that random 3-XOR games with n variables and Cn clauses are satisfiable with high probability if and only if $C \lessapprox 0.92$ [21]. Since quantum strategies do not form a discrete (or even finite-dimensional) set, these methods are not possible. Nor is it obvious how to use more refined tools such as Shearer's Lemma or the Lovasz Local Lemma, which address the question of when sets of overlapping constraints can be simultaneously satisfied. Indeed there are famous examples (such as the Magic Square game) of quantum "advantage" (i.e. the quantum value of a game is higher than the classical value) when there exist strategies for apparently contradictory constraints that succeed with probability 1. These suggest that the barriers to extending our classical intuition are not merely technical but reflect a genuinely different set of rules.

3.2 Results

Our work introduces new techniques that let us make progress on the study of both worst-case complexity and random instances of XOR games with more than two players, in the regime where we are trying to decide whether $\omega^* = 1$. We think of a nonlocal game as a system of

equations whose variables are linear operators, corresponding to the quantum measurements used by the players; a strategy is a solution to this system. Our main innovation is to consider a "dual" system of equations, whose solutions are objects that we call *refutations*. A refutation is a proof that the "primal" system of operator equations induced by the game is infeasible, and thus that $\omega^* \neq 1$. Surprisingly, for games that are symmetric under exchange of the players, we show that the dual system reduces to a set of linear equations over $\mathbb{Z}$, which can be solved efficiently. This leads to our first result (Theorem 3.2.1), an algorithm for efficiently deciding whether $\omega^* = 1$ for a symmetric k-player XOR game, which brings the best known upper bound on this problem down from recursively enumerable [46, 20] to P. Subsequently, by taking the dual of the dual, we are able to explicitly construct a set of quantum strategies (we call these Maximal Entanglement, Relative Phase, or MERP, strategies) that attain value 1 for all symmetric games with $\omega^* = 1$ (Theorem 3.2.2). In Chapter 5 we will show that the symmetry assumption is indeed necessary for our algorithm to work: we exhibit a simple non-symmetric game called the 123 game, for which a simple, non-MERP strategy achieves $\omega^* = 1$, while our algorithm is unable to detect this (Theorem 5.1.1).

The theorem statements of these results are as follows. Proof sketches are in Section 3.3.

Theorem 3.2.1. *There exists an algorithm that, given a k-player symmetric XOR game $\mathcal{G}$ with alphabet size n and m clauses, decides in time $\mathsf{poly}(n, m)$ [2] whether $\omega^*(\mathcal{G}) = 1$ or $\omega^*(\mathcal{G}) < 1$.*

Proof. Section 3.4.3. Sketch in 3.3.3. □

Theorem 3.2.2. *For every k-XOR game $\mathcal{G}$ for which the algorithm of Theorem 3.2.1 shows $\omega^*(\mathcal{G}) = 1$, there exists a k-qubit tensor-product strategy achieving value 1, and a description of the strategy can be computed in polynomial time.*

Proof. Section 3.5. Sketch in Sections 3.3.4 and 3.3.5. □

[2] Note that m and k do not scale independently for symmetric games. Any symmetric game may be specified by m' base clauses that are then symmetrized via at most $k!$ permutations each, meaning $m \leq k!m'$. We could thus naively rewrite this runtime as $\mathsf{poly}(n, k!m')$ to extract the k dependence. Because the core information about the symmetric game is really only contained in the m' clauses, one might expect that it is possible to remove the factor of $k!$, and we hope to address this in a future work.

3.3 Technical Overview

We begin by formally defining a k-XOR game and its classical and quantum values.

Definition 3.3.1. *Define a* ***clause*** $c = (q, s)$ *to be any* $(k+1)$*-tuple consisting of a* ***query*** $q \in [n]^k$ *and* ***parity bit*** $s \in \{-1, 1\}$*. In a* k-***XOR game*** $\mathcal{G}$ *associated with a set of clauses* M*, a verifier selects a clause* $c_i = (q_i, s_i)$ *uniformly at random from* M*. Next, the question* $q_i^{(\alpha)}$ *is sent to the* α*-th player of the game, for all* $\alpha \in [k]$*. The players then each send back a single output* $\in \{-1, 1\}$*, and win the game if their outputs multiply to* s_i*.*

The key property of a game $\mathcal{G}$ is its value – the maximum win probability achievable by players who cooperatively choose a strategy before the game starts, but cannot communicate while the game is being played. We distinguish various versions of the value by physical restrictions placed on the players.

Definition 3.3.2. *For a given game* $\mathcal{G}$*, the* ***classical value*** $\omega(\mathcal{G})$ *is the maximum win probability achievable by players sharing no entanglement.*

The ***tensor-product value*** *is the supremum win probability obtainable by players who share a quantum state but are restricted to making measurements on distinct factors of a tensor-product Hilbert space.*

Finally, the ***commuting operator value*** $\omega^*(\mathcal{G})$ *is the supremum win probability obtainable by players who may make any commuting measurements on a shared quantum state, not necessarily over a tensor-product Hilbert space.* $\omega^*(\mathcal{G})$ *is often also referred to as the field-theoretic value of* $\mathcal{G}$*.*

In this chapter, we focus primarily on a description of the commuting-operator value but in many cases can show that it coincides with the tensor-product value.

For the purpose of analyzing both the classical and commuting operator value of k-XOR games, we find it useful to define a linear algebraic representation for the game. The linear algebraic view represents queries as a matrix and parity bits as a vector. In doing so, it abstracts away from the specifics of labels and player/query indices to reveal the underlying game structure.

Definition 3.3.3. *Given a k-XOR game with m queries and alphabet size n, define the* ***game matrix*** *A as an $m \times kn$ matrix describing query-player-question incidence. Specifically, A can be written as a segmented matrix with k distinct column blocks of size n each, where the jth column of block α consists of a 1 in row i if the αth player receives question j for query q_i, and a 0 otherwise:*

$$A_{i,(\alpha-1)n+j} := \begin{cases} 1 \text{ if } q_i^{(\alpha)} = j \\ 0 \text{ otherwise} \end{cases} . \tag{3.3.1}$$

For such a game, define the length-m ***parity bit vector*** *$\hat{s} \in \mathbb{F}_2^m$ by*

$$\hat{s}_i := \begin{cases} 0 \text{ if } s_i = 1 \\ 1 \text{ if } s_i = -1 \end{cases} . \tag{3.3.2}$$

An XOR game $\mathcal{G}$ is completely specified by providing the game matrix A and parity bit vector $\hat{s}$: $G \sim (A, \hat{s})$. For example, the GHZ game [26] is defined by the clauses (here we use the labels $\{x, y\}$ for the questions instead of the typical $\{0, 1\}$):

$$\mathcal{G}_{GHZ} := \left\{ \begin{bmatrix} x \\ x \\ x \\ +1 \end{bmatrix}, \begin{bmatrix} y \\ y \\ x \\ -1 \end{bmatrix}, \begin{bmatrix} y \\ x \\ y \\ -1 \end{bmatrix}, \begin{bmatrix} x \\ y \\ y \\ -1 \end{bmatrix} \right\} . \tag{3.3.3}$$

We translate the GHZ queries into A_{GHZ} and parity bits into $\hat{s}_{GHZ}$ by:

$$\implies A_{GHZ} := \left(\begin{array}{cccc} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ \hline 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ \hline 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{array}\right)^T \begin{array}{l} \leftarrow (\text{Alice}, x) \\ \leftarrow (\text{Alice}, y) \\ \leftarrow (\text{Bob}, x) \\ \leftarrow (\text{Bob}, y) \\ \leftarrow (\text{Charlie}, x) \\ \leftarrow (\text{Charlie}, y) \end{array} \tag{3.3.4}$$

$$= \left(\begin{array}{cc|cc|cc} 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{array}\right) \quad \text{and} \quad \hat{s}_{GHZ} := \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}. \tag{3.3.5}$$

Many of our results apply to two special classes of XOR games: symmetric XOR games and random XOR games.

Definition 3.3.4. *A **symmetric** k-**XOR** game is an XOR game that additionally satisfies a clause symmetry property: for every clause $c_i = (q_i, s_i)$ in the game, the game must also contain all clauses $c_i' = (q_i', s_i)$ where q_i' is a permutation of the questions in q_i and the parity bit is unchanged.*

Definition 3.3.5. *A **random** k-**XOR** game on m clauses and n variables is an XOR game with the m clauses chosen independently at random from a uniform distribution over $[n]^k \times \{-1, 1\}$.*

3.3.1 Strategies

We next introduce both classical and commuting operator strategies and state claims regarding their value and constraints on when these strategies play perfectly given an XOR game. These claims are proved in Section 3.5.3. For any game, constructing a strategy and computing its value lower-bounds the value of the game. In the commuting operator case, this is generally intractable and motivates the subsequent refutations picture.

Classical Strategies

For any game, the optimal classical strategy can be taken to be a deterministic assignment of answers. In the case of XOR games we will see that it is natural to view this assignment as a vector in $\mathbb{F}_2^{kn}$.

Definition 3.3.6. *A **deterministic classical strategy** dictates that player α outputs $\eta(\alpha, j) \in \{-1, 1\}$ when they receive question j from the verifier. Note that valid outputs must satisfy*

$$\eta^2(\alpha, j) = 1. \tag{3.3.6}$$

*To exploit the linear algebraic picture, it is useful to define a length-kn **classical strategy vector** $\hat{\eta} \in \mathbb{F}_2^{kn}$ analogous to the parity bit vector. It is defined by the relation*

$$\eta(\alpha, j) = (-1)^{\hat{\eta}_{n(\alpha-1)+j}} = \cos\big(\pi \hat{\eta}_{n(\alpha-1)+j}\big). \tag{3.3.7}$$

Here the cos anticipates a generalization that we will see in the quantum case when we construct MERP strategies.

Claim 3.3.7. *If the players play a game $G \sim (A, \hat{s})$ following strategy $\hat{\eta}$, the vector $\hat{o} = A\hat{\eta}$ determines their output, i.e. query i has answer $(-1)^{\hat{o}_i}$. The value of classical strategy $\hat{\eta}$ is*

$$v(G, \hat{\eta}) := \frac{1}{m} \sum_{i=1}^{m} \frac{1 + (-1)^{\hat{o}_i - \hat{s}_i}}{2} = \frac{1}{2} + \frac{1}{2m} \left(\sum_{i=1}^{m} \cos(\pi \left[(A\hat{\eta})_i - \hat{s}_i \right]) \right), \tag{3.3.8}$$

where again we have used an apparently unnecessary cos*, anticipating a quantum generalization. We also treat $\mathbb{F}_2$ and $\{0, 1\}$ as equivalent here.*

These observations lead to a well known procedure using Gaussian elimination to find a classical value-1 strategy or determine that no such strategy exists.

Definition 3.3.8. *Define the **classical constraint equations** for game $\mathcal{G}$ by*

$$A\hat{\eta} = \hat{s} \tag{3.3.9}$$

over $\mathbb{F}_2$. *Equivalently,*

$$\prod_{\alpha=1}^{k} \eta(\alpha, q_i^{(\alpha)}) = s_i, \ \forall i \in [m]. \tag{3.3.10}$$

Claim 3.3.9. *Every solution* $\hat{\eta}$ *to (3.3.9) corresponds to a strategy* η *achieving value 1 on game* $G \sim (A, \hat{s})$*, and vice versa. In other words, a game* $\mathcal{G}$ *has classical value* 1 *iff (3.3.9) has a solution.*

When $\omega(\mathcal{G}) < 1$, on the other hand, there does not exist an efficient algorithm for finding optimal classical strategies (assuming $\mathsf{P} \neq \mathsf{NP}$) [31].

Commuting Operator Strategies

Definition 3.3.10. *Consider a k-XOR game with n variables. For each* $j \in [n]$*, let the Positive-Operator Valued Measure (POVM)* $\{P(\alpha)_1^j, P(\alpha)_{-1}^j\}$ *give the* α*-th player's* ***commuting operator strategy*** *upon receiving question* j *from the verifier. These POVMs act on some shared state* $|\Psi\rangle$*, and different players' POVM elements commute due to the no-communication requirement on the players.*

Using the Naimark dilation theorem, we can restrict our players' strategies to be Projection-Valued Measures (PVMs). We make this restriction for the remainder of the chapter. This allows us to define the following observables.

Definition 3.3.11. *Given a strategy* $\{P(\alpha)_1^j, P(\alpha)_{-1}^j\}$*, define the* ***strategy observable***

$$X_j^{(\alpha)} := P(\alpha)_1^j - P(\alpha)_{-1}^j.$$

Since $\{P(\alpha)_1^j, P(\alpha)_{-1}^j\}$ is a PVM, $X_j^{(\alpha)}$ is a Hermitian operator. Indeed commuting operator strategies are equivalent to imposing the constraints for $\alpha \neq \beta$

$$[X_j^{(\alpha)}, X_{j'}^{(\beta)}] = 0 \qquad \text{(operators held by distinct players commute)} \tag{3.3.11a}$$

$$\left(X_j^{(\alpha)}\right)^2 = I \qquad \text{(square identity, analogous to (3.3.6))} \tag{3.3.11b}$$

The condition for commuting-operator strategies to achieve value 1 is the following generalization of (3.3.9).

Definition 3.3.12. *For a k-XOR game $\mathcal{G}$, define the* ***commuting-operator constraint equations****:*

$$Q_i \left|\Psi\right\rangle := \left(\prod_{\alpha} X_{q_i^{(\alpha)}}^{(\alpha)}\right) \left|\Psi\right\rangle = s_i \left|\Psi\right\rangle, \quad \forall\, i \in [m] \tag{3.3.12}$$

These equations stipulate that applying the strategy observables for a given question to the shared state $|\Psi\rangle$ produces the correct output for that question.

Claim 3.3.13. *A game $\mathcal{G}$ has commuting operator value 1 iff there exists some state and strategy observables that satisfy* (3.3.11) *and* (3.3.12)*.*

While there is an efficient algorithm to solve the classical constraint equations, no such algorithm is known to exist for the commuting operator constraint equations. This difficulty forces us to consider alternative techniques for characterizing the commuting operator value of XOR games.

3.3.2 Refutations

In addition to lower bounding the value of a game by constructing strategies for it, we can also upper bound a game's value by showing no high-value strategy can exist. In particular, we construct proofs that a game cannot have value 1, which we call refutations. Classically, refutations are well understood, and emerge naturally from the dual to the classical constraint equations.

Definition 3.3.14. *Define a* ***classical refutation*** $y \in \mathbb{F}_2^m$ *as any vector satisfying the equations dual to (3.3.9),*

$$\begin{bmatrix} A^T \\ \hat{s}^T \end{bmatrix} y = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{3.3.13}$$

where once again the algebra is over $\mathbb{F}_2$.

Fact 3.3.15. Either a classical refutation y exists satisfying (3.3.13) or a classical strategy $\hat{\eta}$ exists satisfying (3.3.9).

The proof is standard but because dualities like Fact 3.3.15 play a major role in our work, we review it here.

Proof. By the definition of im and ker, we have $\operatorname{im} A \subseteq (\ker A^T)^\perp$. The rank-nullity theorem implies that $\dim \operatorname{im} A = \dim(\ker A^T)^\perp$, meaning that in fact

$$\operatorname{im} A = (\ker A^T)^\perp. \tag{3.3.14}$$

Therefore

$$\hat{s} \notin \operatorname{im} A \quad \Leftrightarrow \quad \hat{s} \notin (\ker A^T)^\perp \quad \Leftrightarrow \quad \exists y \in \ker A^T,\ \hat{s}^T y \neq 0. \tag{3.3.15}$$

□

Another way to view y as a refutation is by interpreting it as the indicator vector of a subset of clauses. Recall from (3.3.10) that clause i corresponds to the equation $\prod_\alpha \eta(\alpha, q_i^{(\alpha)}) = s_i$ over the variables $\eta(\cdot,\cdot)$. If y satisfies (3.3.13) then multiplying the equations corresponding to clauses with $y_i = 1$ yields

$$\prod_{i:y_i=1} \prod_{\alpha \in [k]} \eta(\alpha, q_i^{(\alpha)}) = \prod_{i:y_i=1} s_i \tag{3.3.16}$$

From $A^T y = 0$ and (3.3.6) it follows that the LHS of (3.3.16) equals 1. From $s^T y = 1$ it follows that the RHS of (3.3.16) equals -1. Thus the existence of y satisfying (3.3.13) means there is no η satisfying (3.3.10).

In this chapter we consider the commuting operator analogue of classical refutations. We would like to construct a dual to (3.3.12), meaning a characterization of certificates for the unsatisfiability of (3.3.12). As there is no analogue to the linear algebraic methods used in the classical case, we will instead attempt to generalize (3.3.16).

Cleve and Mittal [15] make use of a noncommutative generalization of (3.3.16), which they call the substitution method, to exhibit refutations of some Binary Constraint System games. We will use a similar method for XOR games in which we multiply together constraints of the form (3.3.12) to obtain a contradiction. Our contribution will be to give a simple characterization of when such refutations exist in the case of symmetric k-XOR games and in some cases, random asymmetric 3-XOR games. Indeed, our characterization will resemble the classical dual equations (3.3.13) although the route by which we obtain it is quite different.

To explain this in more detail, we introduce some definitions.

Definition 3.3.16. *Let Z_1 and Z_2 be two operators formed from products of strategy observables. We say Z_1 is equivalent to Z_2, written $Z_1 \sim Z_2$, if $Z_1 = Z_2$ is an identity for all strategy observables satisfying* (3.3.11)*.*

Definitions 3.3.12 and 3.3.16 then motivate the definition of a (quantum) refutation, analogous to Definition 3.3.14. From now on, a "refutation" will be a quantum refutation unless otherwise specified.

Definition 3.3.17. *Let $\mathcal{G}$ be some k-XOR game with m clauses. A **refutation for** $\mathcal{G}$ is defined to be a sequence $(i_1, i_2, \ldots, i_\ell) \in [m]^\ell$ satisfying*

$$Q_{i_1} Q_{i_2} \ldots Q_{i_\ell} \sim I \qquad \textit{and} \qquad s_{i_1} s_{i_2} \ldots s_{i_\ell} = -1. \tag{3.3.17}$$

Refutations certify that $\omega^* < 1$, analogous to the way that classical refutations certify that $\omega < 1$. In Theorem 3.4.1, we show that in fact any game with $\omega^* < 1$ has a refutation. The proof of this fact relies on a connection between refutations and the ncSoS hierarchy analogous to a connection made by Grigoriev [27] between classical refutations and the SoS hierarchy.

It is not obvious that one can find refutations more easily than one can find strategies. However, we next establish a necessary condition for a game to admit a refutation, and thus an easily-identified subclass of XOR games that certainly do not have a refutation meaning they have $\omega^* = 1$.

3.3.3 Games with no Parity-Permuted Refutations (noPREF Games)

noPREF games are a subclass of entangled XOR games for which it is easy to show no refutation can exist. To motivate their construction and prove some properties about them, we must first redefine refutations from a combinatorial perspective. A more complete treatment of these ideas is given in Section 3.4.

Definition 3.3.18 (Combinatorial Construction of Refutations, informal)**.** *For a k-XOR game $\mathcal{G}$, consider the combinatorial version of the query q_i*[3] *to be a vector with k* ***coordinates*** *(the player indices) with* ***letter*** $q_i^{(\alpha)}$ *at coordinate α. Define the set of* ***words*** *contained in $\mathcal{G}$ to be all vectors formed by concatenating the queries of $\mathcal{G}$ coordinate-wise (by player). The* ***sign*** *of a word contained in $\mathcal{G}$*

$$W = q_{i_1} q_{i_2} \dots q_{i_\ell} \tag{3.3.18}$$

is defined as

$$s_W := s_{i_1} s_{i_2} \dots s_{i_\ell}. \tag{3.3.19}$$

We will refer to each coordinate of the word as a ***wire****. The identity I under the concatenating action is the word that is blank on every wire.*

Define an equivalence relation generated by all wire-by-wire permutations of the following base relations (in this setting the product of two vectors indicates their coordinate-wise concatenation).

1. *(Repeated elements cancel) :* $\begin{bmatrix} j \\ \vdots \end{bmatrix} \begin{bmatrix} j \\ \vdots \end{bmatrix} \sim \begin{bmatrix} \\ \vdots \end{bmatrix}$ $all j \in [n]$

2. *(Elements on different wires commute) :* $\begin{bmatrix} j \\ j' \\ \vdots \end{bmatrix} \sim \begin{bmatrix} j \\ \\ \vdots \end{bmatrix} \begin{bmatrix} \\ j' \\ \vdots \end{bmatrix} \sim \begin{bmatrix} \\ j' \\ \vdots \end{bmatrix} \begin{bmatrix} j \\ \\ \vdots \end{bmatrix}$ $all j, j' \in [n]$

A ***refutation*** *is defined to be a sequence $(i_1, i_2, i_3, ...i_\ell) \in [m]^\ell$ for which*

$$q_{i_1} q_{i_2} ... q_{i_\ell} \sim I \qquad \text{and} \qquad s_{i_1} s_{i_2} ... s_{i_\ell} = -1. \tag{3.3.20}$$

We claim that this definition of a refutation is equivalent to the one given in Section 3.3.2. Intuitively, such a construction explicitly manipulates the operator identities required by each clause of $\mathcal{G}$ in a way that exploits the operator requirements of (3.3.11) to produce a refutation as in Definition 3.3.17. We prove this fact in Section 3.4. We next motivate the noPREF class of games by making the following key observation.

[3] We overload the notation q_i here to indicate both the definitional and combinatorial version of a query, with the relevant meaning clear from context.

Observation 3.3.19. All elements contained in queries at even positions in a refutation must cancel with queries at odd positions.

To exploit this observation, we find it useful to define a broader equivalence relation $\overset{p}{\sim}$ that allows for a parity-preserving permutation on each wire before canceling and commuting letters.

Definition 3.3.20 (Informal)**.** *We say k-XOR word W_A is **parity-permuted equivalent** to W_B—denoted $W_A \overset{p}{\sim} W_B$—if $W_A \sim W'_B$ where some permutations of the even positions and odd positions on each wire of W_B can produce W'_B.*

Since this is just a broadening of the equivalence given in Definition 3.3.18, $W_1 \sim W_2 \implies W_1 \overset{p}{\sim} W_2$. With this equivalence relation in hand, we can state a necessary condition for the existence of a refutation of a game $\mathcal{G}$.

Definition 3.3.21. *A game $\mathcal{G}$ contains a **Parity-Permuted Refutation (PREF)** if the game $\mathcal{G}$ contains a word which is $\overset{p}{\sim} I$ with sign -1.*

*The set of **PREF Games** are the set of XOR games that contain PREFs. The set of **noPREF Games** are the set of XOR games that do not.*

Theorem 3.3.22 (Necessary condition for refutation)**.** *If a game $\mathcal{G}$ admits a refutation, it contains a PREF.*

Proof (sketch). This follows essentially immediately from the observation that $\sim$ implies $\overset{p}{\sim}$. □

Corollary 3.3.23. *Every noPREF game has commuting operator value 1.*

Proof. This follows directly from Theorem 3.3.22 and the completeness of refutations (Theorem 3.4.1). □

The significance of noPREF games is made clear by the two following theorems. For both, a short proof sketch is given, while the full proofs are delegated to Section 3.4.

Theorem 3.3.24 (Informal)**.** *There exists a poly-time algorithm that decides membership in the set of noPREF games.*

Proof (sketch). The key observation here is that a game $G \sim (A, \hat{s})$ contains a PREF if and only if there is a solution to the set of equations

$$A^T z = 0 \tag{3.3.21}$$

$$\hat{s}^T z = 1 \pmod 2 \tag{3.3.22}$$

for some $z \in \mathbb{Z}^m$. If (3.3.21) and (3.3.22) can be satisfied, the game $\mathcal{G}$ contains a PREF built by interleaving the multisets of clauses

$$\mathcal{O} = \{q_i \text{ with multiplicity } |z_i| alli : z_i > 0\} \tag{3.3.23a}$$

$$\mathcal{E} = \{q_i \text{ with multiplicity } |z_i| alli : z_i < 0\} \tag{3.3.23b}$$

such that their elements are placed in odd and even positions, respectively. The reverse direction requires a technical lemma relating the even and odd clauses of a PREF. Then standard techniques for solving linear Diophantine equations complete the proof. □

The vector z defined in the proof of Theorem 3.3.24 is sometimes referred to as a PREF specification due to (3.3.23).[4]

Theorem 3.3.25 (Informal)**.** *The noPREF characterization is complete for symmetric games. That is, every value 1 symmetric game is in the noPREF set.*

Proof (sketch). We use the structure of symmetric games to construct shuffle gadgets – short words that move letters from one wire to another when they are appended onto an existing word. We then show shuffle gadgets are sufficient to construct a refutation given a PREF contained in the game. This shows that containing a PREF is both necessary and sufficient for a symmetric game to have a refutation. Then a symmetric game is either in the set of noPREF games or has value < 1. □

Theorems 3.3.24 and 3.3.25 together show that the class of symmetric value 1 games has a poly-time deterministic algorithm, while previously the question of whether such games took

[4]Or a MERP refutation, for reasons described in Section 3.3.5

value 1 was not known to be decidable. This progress is due to the noPREF characterization of games.

Given that noPREF games form a large class of value 1 games, it is reasonable to try to construct a commuting operator strategy to play them. We can ask if there exists a strategy dual to the PREF criteria, similar to what we have described in the classical and commuting operator cases. In particular, we ask if a game $\mathcal{G}$ not satisfying (3.3.21) and (3.3.22) guarantees existence of a solution to the constraint equations indicating some simple family of strategies can achieve value 1 for $\mathcal{G}$.

Somewhat miraculously, the answer to this question turns out to be yes. We proceed by first defining this class of strategies, then showing that their constraint equations are dual to the PREF criteria for any game.

3.3.4 Maximal Entanglement, Relative Phase (MERP) Strategies

We introduce a family of "Maximal Entanglement, Relative Phase" (MERP) strategies: a useful subfamily of the set of tensor-product (and thus commuting-operator) strategies. MERP strategies are a generalization of the GHZ strategy to arbitrary games. Crucially, determining whether a MERP strategy achieves value 1 for a game, and if so a construction for such a strategy, can be described in time polynomial in m, n, and k.[5]

Furthermore, the conditions for a MERP strategy to achieve value 1 are dual to the PREF condition for a game, meaning MERP achieves value 1 on any noPREF game. In particular, this means MERP strategies achieve tensor-product value 1 on any symmetric XOR game with $\omega^* = 1$ (Theorem 3.3.25) as well as on a family of non-symmetric games (APD games, Section 5.2.3) with $\omega^* = 1$ and classical value $\omega \to \frac{1}{2}$.

We begin with the definition of a MERP strategy for a game $\mathcal{G}$.

Definition 3.3.26 (MERP)**.** *Given a k-XOR game $\mathcal{G}$ with m clauses, a **MERP strategy** for $\mathcal{G}$ is a tensor-product strategy in which:*

[5]For symmetric games, $m \sim \exp\{k\}$, so in this case one can decide MERP value 1 and describe a strategy in time polynomial in m and n.

1. *The k players share the maximally entangled state*

$$|\Psi\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle^{\otimes k} + |1\rangle^{\otimes k}\right] \tag{3.3.24}$$

with player α having access to the α-th qubit of the state.

2. *Upon receiving question j from the verifier, player α rotates his qubit by an angle $\theta(\alpha, j)$ about the Z axis, then measures his qubit in the X basis and sends his observed outcome to the verifier.*

   Explicitly, we define the states

$$|\theta(\alpha, j)_\pm\rangle := \frac{1}{\sqrt{2}}\left[|0\rangle \pm e^{i\theta(\alpha,j)}\,|1\rangle\right] \tag{3.3.25}$$

   and pick strategy observables

$$X_j^{(\alpha)} := |\theta(\alpha, j)_+\rangle\langle\theta(\alpha, j)_+| - |\theta(\alpha, j)_-\rangle\langle\theta(\alpha, j)_-|\,. \tag{3.3.26}$$

There exists a useful parallel between MERP strategies and classical strategies, which we summarize below. Almost identically to the classical value (3.3.8),

Claim 3.3.27. *Let the length-kn* ***MERP strategy vector*** *for a given MERP strategy be defined by*

$$\hat{\theta}_{(\alpha-1)n+j} := \frac{1}{\pi}\theta(\alpha, j). \tag{3.3.27}$$

The value achieved by that MERP strategy on game $\mathcal{G}$ is:

$$v^{MERP}(G, \hat{\theta}) := \frac{1}{2} + \frac{1}{2m}\left(\sum_{i=1}^{m} \cos\Big(\pi\left[(A\hat{\theta})_i - \hat{s}_i\right]\Big)\right). \tag{3.3.28}$$

Proof. Explicit calculation. Done in full in Section 3.5.2. □

Claim 3.3.27 allows us to write down the constraint equations for MERP strategies to achieve $v^{\text{MERP}} = 1$.

Definition 3.3.28. *Define the* ***MERP constraint equations*** *for game $\mathcal{G}$ by*

$$A\hat{\theta} = \hat{s} \pmod 2 \tag{3.3.29}$$

with $\hat{\theta} \in \mathbb{Q}^{kn}$.

(We could have equivalently required $\hat{\theta}$ to be in $\mathbb{R}^{kn}$. This is because $A, \hat{s}$ have integer entries and so any real solution to (3.3.29) will also be rational.)

Claim 3.3.29. *A MERP strategy achieves $v^{MERP} = 1$ on a game $\mathcal{G}$ iff its MERP constraint equations have a solution. A solution $\hat{\theta}$ corresponds to the MERP strategy in which player α uses $\theta(\alpha, j) = \pi\hat{\theta}_{(\alpha-1)n+j}$.*

Intuitively, MERP provides an explicit construction allowing players to return an *arbitrary phase* on each input, rather than the classical 0 or π. The MERP constraint equations then ensure that for each question the returned phases sum to $\pi\hat{s}_i$ up to multiples of 2π. For any game, Claim 3.3.29 allows us to efficiently determine whether some MERP strategy achieves value 1 via Gaussian elimination over $\mathbb{Q}$. We often refer to this optimal MERP strategy[6] for a game $\mathcal{G}$ as simply *the MERP strategy* for $\mathcal{G}$.

3.3.5 MERP - PREF Duality

The set of games for which MERP achieves value 1 is exactly the set noPREF. As in the classical and commuting operator cases, the MERP constraint equations (3.3.29) are dual to the PREF conditions:

Theorem 3.3.30. *For any game $\mathcal{G}$, either there exists a PREF specification, or a MERP strategy with value 1.*

Proof. Technical proof in the style of a Theorem of Alternatives, analogous to Fact 3.3.15. See Section 3.5.3. □

[6]Despite the language, we do not wish to suggest that there is a single optimal MERP strategy. Instead one should imagine some convention being used to specify a unique MERP strategy from the set of optimal ones.

Because of Theorem 3.3.30 we also refer to a PREF specification z as a *MERP refutation*.

Figure 3-1 summarizes the extensions of the classical duality relations presented in this chapter. The general quantum duality provides a complex but complete description of games with $\omega^* = 1$. The PREF conditions are efficient to compute, but are only *necessary* conditions for constructing commuting operator refutations, and thus the dual, MERP value 1, holds true for only a subset of all $\omega^* = 1$ games. We can make a stronger statement about symmetric games: PREFs are both necessary and sufficient for a symmetric game to have a refutation, so the duality ensures MERP achieves value 1 for all symmetric games with $\omega^* = 1$.

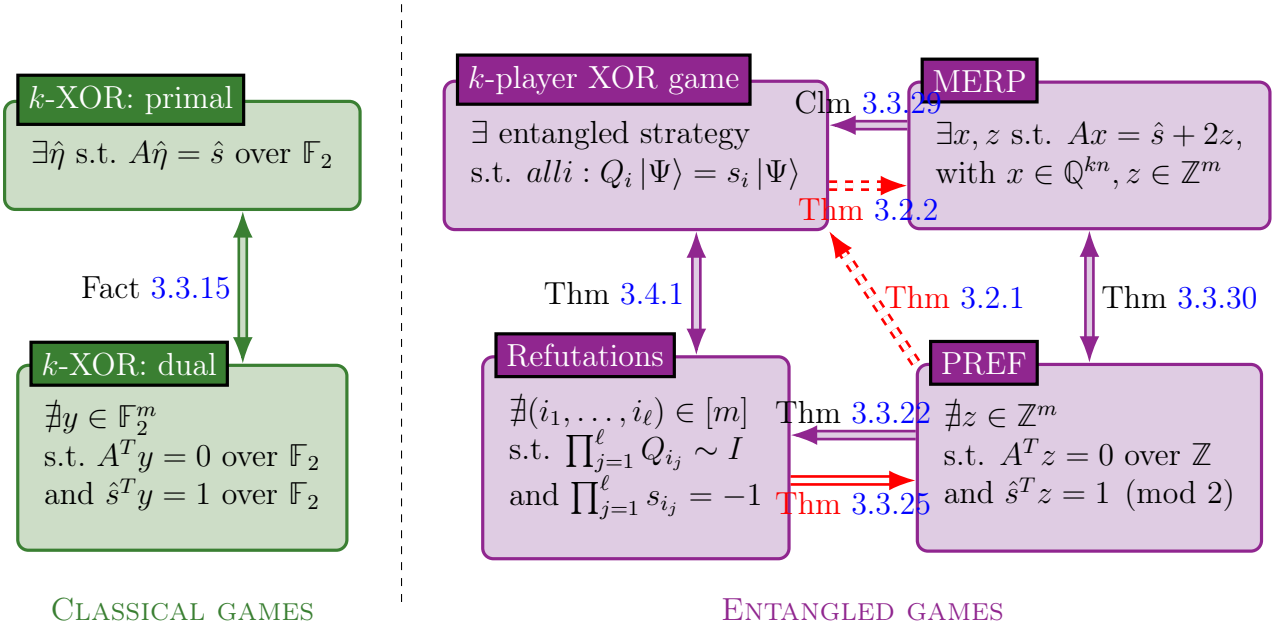

Figure 3-1: We extend the well-understood duality relation for classical XOR games (left) to a more complex set of dualities characterizing perfect strategies for entangled XOR games (right). The arrows indicate implications, with the red, unfilled arrows holding for symmetric games only. The dashed red arrows follow from the other arrows for symmetric games.

3.3.6 Implications

Finally we can use our main results to analyze some particular families of games and partially characterize the XOR game landscape.

In the $\omega^* = 1$ regime, we construct a family of games that generalize the GHZ game, termed the Asymptotically Perfect Difference (APD) family. Members are parameterized by scale K, with the K-th member having $k = 2^K - 1$ players, and $K = 2$ reproducing GHZ.

The APD family is contained in the noPREF set ($\omega^* = 1$) and has perfect difference in the asymptotic limit,

$$\lim_{K\to\infty} 2(\omega^* - \omega) = 1. \tag{3.3.30}$$

This demonstrates that XOR games include a subset for which (at least asymptotically) the best classical strategy is no better than random while a tensor-product strategy (MERP) can play perfectly. Details of this construction are given in Section 5.2.3. We also give, in Section 5.2.1, the construction for a (nonsymmetric) game for which $\omega^* = 1$ but which falls outside the noPREF set, which shows the incompleteness of the PREF criteria.

To study the $\omega^* < 1$ regime, we consider the behavior of randomly generated XOR games with a large number of clauses. We prove Theorem 5.1.4 by explicitly constructing a refutation for such games using insights developed in previous sections. Interestingly, we also show such games have a minimal length refutation that scales like $\Omega(n \log(n)/\log(\log(n)))$, which implies that it takes the ncSoS algorithm superexponential time to show that these games have $\omega^* < 1$ (Lemma 3.4.4 and Theorem 5.1.5). These results can be seen as quantum analogues of Grigoriev's [27] integrality gap instances for classical XOR games. Finally, we try to push the potentially superexponential runtime of ncSoS to its extremes. We demonstrate a family of symmetric games, called the Capped GHZ family, that provably have $\omega^* < 1$, but have minimum refutation length exponential in the number of clauses (Section 5.2.2). For games in this family the ncSoS algorithm requires time doubly exponential to prove that their commuting operator value is < 1 while the noPREF criterion can be used to conclude this fact in polynomial time.

3.4 Refutations

Refutations are a powerful tool for differentiating between XOR games with perfect commuting operator strategies ($\omega^* = 1$) and those with ω^* bounded away from 1. In Section 3.4.1, we prove Theorem 3.4.1 and Theorem 3.4.2, relating refutations to the commuting operator value of XOR games:

Theorem 3.4.1. *An XOR game $\mathcal{G}$ has commuting operator value $\omega^*(\mathcal{G}) = 1$ if and only if it*

admits no refutations.

Theorem 3.4.2. *Let $\mathcal{G}$ be an XOR game consisting of m queries, with $\mathcal{G}$ yielding a length-ℓ refutation. The commuting operator value of the game is bounded above by*

$$\omega^*(\mathcal{G}) \leq 1 - \frac{\pi^2}{4m\ell^2}. \tag{3.4.1}$$

Informally, Theorem 3.4.1 gives completeness and soundness of refutations when used as a proof system for checking if a game has $\omega^* < 1$. Theorem 3.4.2 improves the soundness.

We previously introduced the notion of the combinatorial view of refutations (Definition 3.3.18) and containing a PREF as a necessary condition for a game to have a refutation (Corollary 3.3.23). Section 3.4.2 presents the combinatorial view in more detail, and proves that a PREF specification and existence of a particular set of "shift gadgets" is a *sufficient* condition for a refutation to exist. Finally, Section 3.4.3 demonstrates that for symmetric XOR games, all desired "shift gadgets" are automatically available, meaning that a refutation exists if and only if a PREF specification exists, thus providing an efficient technique to decide whether any symmetric XOR game has perfect commuting operator value.

3.4.1 Upper Bound on Value

We begin by proving Theorem 3.4.1. The main tool we use is the non-commuting Sum of Squares (ncSoS) hierarchy, also known as the NPA hierarchy [46, 20]. Given a game $\mathcal{G}$, each level in the ncSoS hierarchy is a semidefinite program depending on $\mathcal{G}$ whose solution gives an upper bound on the value $\omega^*(\mathcal{G})$; higher levels correspond to larger semidefinite programs and tighter upper bounds. While we refer the reader to the references cited above for a full description, we include here a definition of the key object used in constructing the hierarchy: the *pseudoexpectation* operator.

Definition 3.4.3. *Given an XOR game $\mathcal{G}$, a **degree-d pseudoexpectation operator** or **pseudodistribution** is a linear function $\tilde{\mathbb{E}}[\cdot]$ that maps formal polynomials of degree at most d over the strategy observables $X_j^{(\alpha)}$ to complex numbers. A pseudoexpectation $\tilde{\mathbb{E}}[\cdot]$ is valid if*

- *for all polynomials p of degree at most $d/2$, $\tilde{\mathbb{E}}\left[p^\dagger p\right] \geq 0$,*

- *for all polynomials* p_1, p_2 *with* $\deg(p_1 p_2) \leq d-2$ *and indices* $\alpha \in [k]$ *and* $j \in [n]$,

$$\tilde{\mathbb{E}}\left[p_1 \left\{(X_j^{(\alpha)})^2 - I\right\} p_2\right] = 0. \tag{3.4.2}$$

- *for all polynomials* p_1, p_2 *with* $\deg(p_1 p_2) \leq d-2$ *and indices* $\alpha \neq \alpha' \in [k]$ *and* $j, j' \in [n]$,

$$\tilde{\mathbb{E}}\left[p_1 \left\{X_j^{(\alpha)} O^{\alpha'}(j') - O^{\alpha'}(j') X_j^{(\alpha)}\right\} p_2\right] = 0. \tag{3.4.3}$$

Intuitively speaking, these requirements state that any algebraic manipulations allowed by (3.3.11) *are also allowed under the pseudoexpectation, as long as they never result in a polynomial of degree greater than* d. *We further say that a pseudoexpectation satisfies a clause* $c_i = (q_i, s_i)$ *if for all polynomials* p_1, p_2 *with degrees summing to* $\leq d-k$, $\tilde{\mathbb{E}}\,[p_1(Q_i - s_i I)p_2] = 0$.

The full ncSoS algorithm involves optimizing over all valid pseudoexpectation operators that satisfy clauses in the game; it can be shown that this optimization reduces to a semidefinite program in matrices whose dimension is the number of monomials of degree at most $d/2$ in the observables $X_j^{(\alpha)}$. In the special case of determining whether the game value is 1, it reduces to checking for the existence of such a pseudoexpectation operator.

In [27], Grigoriev showed a connection between refutations of classical games and pseudodistributions which appear to satisfy all clauses of a classical XOR game. In our analysis, we will adapt some of these arguments to the quantum setting. In particular, Lemma 3.4.4 gives a quantum analogue of Grigoriev's central insight that, in the special case of deciding whether the game value is 1, the sum-of-squares hierarchy reduces to checking for the existence of a refutation.

In addition to being key to the proof of Theorem 3.4.1, Lemma 3.4.4 also gives a bound on the time it takes the ncSoS algorithm to show a XOR game has value < 1 in terms of the minimum length refutation admitted by the game.

Lemma 3.4.4. *For any* k*-XOR game* $\mathcal{G}$ *with no refutation of length* $\leq 2\ell$ *there exists a degree-*$k\ell$ *pseudodistribution whose pseudoexpectation satisfies every clause in* $\mathcal{G}$. *Consequently, it takes time at least* $\Omega((nk)^{k\ell})$ *for the ncSoS algorithm to prove* $\omega^*(\mathcal{G}) \neq 1$.

Proof. To construct this pseudodistribution, we follow a procedure of Grigoriev [27]. For each clause $c_i = (q_i, s_i)$, define

$$\tilde{\mathbb{E}}\left[Q_i\right] = \tilde{\mathbb{E}}\left[\prod_{\alpha} O^{\alpha}(q_i^{(\alpha)})\right] := s_i, \tag{3.4.4}$$

and for any *word*[7] w which can be obtained as a product of $N \leq \ell$ queries,

$$w := \prod_{x=1}^{N} Q_{i_x}, \tag{3.4.5}$$

define the pseudoexpectation of w to be the product of the parity bits s_{i_x} associated with each query Q_{i_x} in the operator construction:

$$\tilde{\mathbb{E}}\left[w\right] := \prod_{x=1}^{N} s_{i_x}. \tag{3.4.6}$$

We need to argue that this prescription is well-defined, i.e. that (3.4.5) and (3.4.6) never assign two different values to the same $\tilde{\mathbb{E}}\left[w\right]$. Suppose to the contrary that $w = \prod_{x\in[M]} Q_{i_x} = \prod_{y\in[N]} Q_{j_y}$ with $M, N \leq \ell$ but that $\prod_{x\in[M]} s_{i_x} \neq \prod_{y\in[N]} s_{j_y}$. Since (3.4.6) can only take on the values ± 1 we have $\prod_{x\in[M]} s_{i_x} \cdot \prod_{y\in[N]} s_{j_y} = -1$. Also each Q_i is Hermitian, so

$$1 = ww^{\dagger} = Q_{i_1} \cdots Q_{i_N} Q_{j_M} \cdots Q_{j_1}. \tag{3.4.7}$$

This constructs a refutation of length $M + N \leq 2\ell$, contradicting our hypothesis that no such refutation exists. We conclude that $\tilde{\mathbb{E}}\left[w\right]$ is well-defined for the choices of w resulting from (3.4.5).

For all other words (i.e. those that cannot be obtained as products of queries or have length $> \ell$), set their pseudoexpectation to 0. Finally, extend the definition by linearity to sums and scalar multiples of operator products.

Moreover, $\tilde{\mathbb{E}}\left[\cdot\right]$ induces an equivalence relation on words: we say that words $w_a \overset{\tilde{\mathbb{E}}}{\sim} w_b$ if $\tilde{\mathbb{E}}\left[w_a^{\dagger} w_b\right] \neq 0$. This relation therefore partitions the set of words into equivalence classes

[7] In this context, we borrow this terminology from the combinatorial picture to indicate any product of strategy observables.

$C_1, C_2, \ldots$. We pick a representative element w_i for each class C_i. A key feature of the equivalence relation is that for $w_a, w_b \in C_i$,

$$w_a \overset{\tilde{\mathbb{E}}}{\sim} w_b \quad \Longrightarrow \quad \tilde{\mathbb{E}}\left[w_a^\dagger w_b\right] = \tilde{\mathbb{E}}\left[w_a^\dagger w_i\right] \tilde{\mathbb{E}}\left[w_i^\dagger w_b\right]. \tag{3.4.8}$$

To show that $\tilde{\mathbb{E}}[\cdot]$ is a pseudodistribution, it suffices to show that for any polynomial p of degree at most $k\ell/2$ in the operators $X_j^{(\alpha)}$, $\tilde{\mathbb{E}}\left[p^\dagger p\right] \geq 0$. Group the monomials in p according to the equivalence classes, so that $p = p_1 + p_2 + \ldots$ where each p_i is a sum of terms from equivalence class C_i. It follows that

$$\tilde{\mathbb{E}}\left[p^\dagger p\right] = \sum_i \sum_j \tilde{\mathbb{E}}\left[p_i^\dagger p_j\right] = \sum_i \tilde{\mathbb{E}}\left[p_i^\dagger p_i\right]. \tag{3.4.9}$$

So we have reduced the problem to showing that $\tilde{\mathbb{E}}\left[q^\dagger q\right] \geq 0$ for any polynomial q, all of whose terms belong to the same equivalence class. Write q as a linear combination of words in equivalence class C_i,

$$q = \alpha_1 w_1 + \cdots + \alpha_s w_s. \tag{3.4.10}$$

Then

$$\begin{aligned}
\tilde{\mathbb{E}}\left[q^\dagger q\right] &= \tilde{\mathbb{E}}\left[\sum_{a,b=1}^{s} \alpha_a^* \alpha_b w_a^\dagger w_b\right] && (3.4.11)\\
&= \sum_{a,b=1}^{s} \alpha_a^* \alpha_b \tilde{\mathbb{E}}\left[w_a^\dagger w_b\right] && (3.4.12)\\
&\overset{(3.4.8)}{=} \sum_{a,b=1}^{s} \alpha_a^* \alpha_b \left(\tilde{\mathbb{E}}\left[w_i^\dagger w_a\right]\right)^\dagger \tilde{\mathbb{E}}\left[w_i^\dagger w_b\right] && (3.4.13)\\
&= \left|\sum_a \alpha_a \tilde{\mathbb{E}}\left[w_i^\dagger w_a\right]\right|^2 && (3.4.14)\\
&\geq 0. && (3.4.15)
\end{aligned}$$

The existence of this pseudodistribution implies that the ncSoS algorithm would need to run to level at least $k\ell$ in the ncSoS hierarchy to show $\mathcal{G}$ has commuting operator value < 1. This can be converted to a lower bound on the runtime by standard results in semidefinite

programing. □

Finally, we state and prove the duality between refutations and $\omega^* = 1$.

Theorem 3.4.1. An XOR game $\mathcal{G}$ has commuting operator value $\omega^*(\mathcal{G}) = 1$ if and only if it admits no refutations.

Proof. In one direction, Definition 3.3.12 immediately implies that if the game has commuting value 1, then there are no refutations.

In the other direction, suppose there are no refutations. Then, by Lemma 3.4.4 and taking $\ell \to \infty$ we see there exists a pseudodistribution under which every clause is satisfied, and this pseudodistribution satisfies the constraints of all levels of the ncSoS hierarchy [46]. Since it is known that the ncSoS hierarchy converges to the commuting value of the game, it follows that this value is 1. □

Classical refutations prove that a constraint satisfaction problem is not feasible, and so if there are m constraints they trivially yield an upper bound of $1 - 1/m$. In the commuting operator case, even this statement is not obvious. In particular, one could worry that a game with a quantum refutation still admits a sequence of commuting operator strategies with limiting value 1.

However, here we prove Theorem 3.4.2, showing that even in the commuting operator case, refutations yield explicit upper bounds on $\omega^*(\mathcal{G})$ that are strictly less than 1. An argument similar to the one presented here was known previously, and used to derive a comparable result in Section 5 of [15].

Theorem 3.4.2. Let $\mathcal{G}$ be an XOR game consisting of m queries, with $\mathcal{G}$ yielding a length-ℓ refutation. The commuting operator value of the game is bounded above by

$$\omega^*(\mathcal{G}) \leq 1 - \frac{\pi^2}{4m\ell^2}. \tag{3.4.16}$$

Proof. Recall from Definition 3.3.12 that the Hermitian operator Q_i is defined for some XOR game $\mathcal{G}$, and represents the collective measurements made by the players upon receiving query q_i. It has eigenvalues ± 1, which correspond to the value of the XOR'd bit received by the verifier. Define $\tilde{Q}_i := s_i Q_i$, so the 1 eigenspace of $\tilde{Q}_i$ corresponds to measurement outcomes

on which the players win the game given query q_i, and the -1 eigenspace corresponds to measurement outcomes on which the players lose the game. Let $(i_1, i_2, \ldots i_\ell)$ be the assumed refutation for $\mathcal{G}$. Letting $|\Psi\rangle$ be the state shared by the players, we have

$$\tilde{Q}_{i_1}\tilde{Q}_{i_2}\ldots\tilde{Q}_{i_\ell}|\Psi\rangle = (s_{i_1}s_{i_2}\ldots s_{i_\ell})\, Q_{i_1}Q_{i_2}\ldots Q_{i_\ell}|\Psi\rangle = -1\,|\Psi\rangle\,. \tag{3.4.17}$$

On the other hand, if we let $P_i = \frac{I-\tilde{Q}_i}{2}$ be the projector on to the -1 eigenspace of $\tilde{Q}_i$ then the losing probability is

$$\delta := \frac{1}{m}\sum_{i=1}^{m} \mathrm{Tr}\left[P_i\,|\Psi\rangle\langle\Psi|\right] \tag{3.4.18}$$

We now follow an argument similar to the union bound proof of [25]. Let $\angle(|\alpha\rangle, |\beta\rangle) = \arccos|\langle\alpha|\beta\rangle|$ and observe that it satisfies the triangle inequality, i.e. $\angle(|\alpha\rangle, |\gamma\rangle) \leq \angle(|\alpha\rangle, |\beta\rangle) + \angle(|\beta\rangle, |\gamma\rangle)$. Then

$$\pi \overset{(3.4.17)}{\leq} \sum_{x=1}^{\ell} \angle\Big(|\Psi\rangle, Q_{i_x}|\Psi\rangle\Big) \qquad \text{Note that } Q_i \text{ is unitary.} \tag{3.4.19}$$

$$= \sum_{x=1}^{\ell} \arccos\left(1 - 2\,\mathrm{Tr}\left[P_{i_x}|\Psi\rangle\langle\Psi|\right]\right) \tag{3.4.20}$$

$$\leq \sum_{x=1}^{\ell} 2\sqrt{\mathrm{Tr}\left[P_{i_x}|\Psi\rangle\langle\Psi|\right]} \tag{3.4.21}$$

$$\overset{(3.4.18)}{\leq} 2\sum_{x=1}^{\ell}\sqrt{m\delta} \tag{3.4.22}$$

$$= 2\ell\sqrt{m\delta}. \tag{3.4.23}$$

□

3.4.2 Tools for Constructing Refutations

Having demonstrated the utility of refutations, we return to the combinatorial picture of refutations and prove necessary and sufficient conditions for an XOR game to contain a refutation.

Combinatorics

We now formally reintroduce k-XOR games from a combinatorial standpoint. Several definitions mirror those in Section 3.3.2 but are presented here in a slightly different form to enable discussion of combinatorial proofs.

Definition 3.4.5. *A k-XOR **game** on m clauses with n questions is defined to be a set of m k-tuples, consisting of elements of $[n]$, with m associated parity bits. An individual k-tuple is called a query, and is denoted by*

$$q_i = \begin{bmatrix} q_i^{(1)} \\ q_i^{(2)} \\ \vdots \\ q_i^{(k)} \end{bmatrix}. \tag{3.4.24}$$

Definition 3.4.6. *A **word** W on alphabet $[n]$ is a k-tuple of the form*

$$W = \begin{bmatrix} w_{11}\ w_{12}\ \dots\ w_{1\ell_1} \\ w_{21}\ w_{22}\ \dots\ w_{2\ell_2} \\ \vdots \\ w_{k1} w_{k2} \dots w_{k\ell_k} \end{bmatrix} \tag{3.4.25}$$

*with all $w_{ij} \in [n]$. Each row of W is referred to as a **wire** of the word, and the α-th row is sometimes denoted by $W^{(\alpha)}$. When all wires have length ℓ, (so $\ell_1 = \ell_2 = \dots \ell_k = \ell$) we say W has length ℓ.*

The product of two words is defined to be their coordinate-wise concatenation. The notation $q_{i_1} q_{i_2} \dots q_{i_\ell}$ then refers to the length ℓ word given by

$$q_{i_1} q_{i_2} \dots q_{i_\ell} = \begin{bmatrix} q_{i_1}^{(1)}\ q_{i_2}^{(1)}\ \dots\ q_{i_\ell}^{(1)} \\ q_{i_1}^{(2)}\ q_{i_2}^{(2)}\ \dots\ q_{i_\ell}^{(2)} \\ \vdots \\ q_{i_1}^{(k)}\ q_{i_2}^{(k)}\ \dots\ q_{i_\ell}^{(k)} \end{bmatrix}. \tag{3.4.26}$$

Finally, define the ***identity word*** I *to be the empty k-tuple, which satisfies*

$$IW = IW = W \tag{3.4.27}$$

for any word W.

Definition 3.4.7. *A game $\mathcal{G}$* ***contains a word*** W ***with sign*** $s_W \in \{\pm 1\}$ *if*

$$W = q_{i_1} q_{i_2} \dots q_{i_\ell} \text{ and} \tag{3.4.28}$$

$$s_W = s_{i_1} s_{i_2} \dots s_{i_\ell} \tag{3.4.29}$$

for some $(i_1, i_2, \dots i_\ell) \in [m]^\ell$.

Definition 3.4.8. *Relations are used to express equivalence between words. There are two basic types (shown here for 3-XOR, and defined analogously for k-XOR).*

1. *(Commute Relations):*

$$\begin{bmatrix} j \\ j' \\ j'' \end{bmatrix} \sim \begin{bmatrix} \\ \\ j'' \end{bmatrix} \begin{bmatrix} \\ j' \\ \end{bmatrix} \begin{bmatrix} j \\ \\ \end{bmatrix} \sim \begin{bmatrix} \\ j' \\ \end{bmatrix} \begin{bmatrix} j \\ \\ \end{bmatrix} \begin{bmatrix} \\ \\ j'' \end{bmatrix} \sim \begin{bmatrix} j \\ \\ \end{bmatrix} \begin{bmatrix} \\ j'' \\ \end{bmatrix} \begin{bmatrix} \\ j' \\ \end{bmatrix} \qquad all j, j', j'' \in [n] \tag{3.4.30}$$

2. *(Cancellation Relations):*

$$\begin{bmatrix} j^2 \\ \\ \end{bmatrix} \sim \begin{bmatrix} j^2 \\ \\ \end{bmatrix} \sim \begin{bmatrix} \\ \\ j^2 \end{bmatrix} \sim I \qquad all j \in [n] \tag{3.4.31}$$

The relationship property is associative (as suggested by the notation), so more complicated equivalences can be constructed by concatenating the ones above.

Definition 3.4.9. *Given a k-XOR game $\mathcal{G}$, a length ℓ* ***refutation*** *for that game is a length*

ℓ word W contained in $\mathcal{G}$ with sign -1 and

$$W \sim I. \tag{3.4.32}$$

PREFs and Shuffle Gadgets

The key difference between entangled and classical strategies is that in the entangled case, the strategy observables do not all commute with each other. In other words, strings of queries can be acted on nontrivially by permutations. In this section we consider equivalence under a restricted class of parity-preserving permutations, and use the fact that *at least one element* of a class equivalent to some refutation must be contained in a game for it to admit a refutation, giving a tractable necessary condition for a refutation to exist. We then define gadgets that perform these permutations while preserving the associated parity bits. The result will be a useful set of sufficient conditions for a refutation to exist.

We recall the formal definitions related to these equivalence classes.

Definition 3.3.20. Given two 1-XOR words W_1, W_2, we say that W_1 is **parity-permuted equivalent** to W_2—denoted $W_1 \overset{p}{\sim} W_2$—if there exists a permutation π mapping even indices to even indices and odd indices to odd indices such that $W_1 \sim \pi(W_2)$.

For k-XOR words W_A, W_B, we say $W_A \overset{p}{\sim} W_B$ if $W_A^{(\alpha)} \overset{p}{\sim} W_B^{(\alpha)}$ for all $\alpha \in [k]$.

From the definition, we see that $\overset{p}{\sim}$ is necessary for $\sim$, i.e.

$$W_1 \sim W_2 \implies W_1 \overset{p}{\sim} W_2. \tag{3.4.33}$$

We can then conclude that a game $\mathcal{G}$ contains a refutation only if it contains a word $W \overset{p}{\sim} I$ with sign -1. To make this necessary condition more useful to us, we will move from an operational definition of the $\overset{p}{\sim}$ relation to a structural one. This is done by talking about the even and odd subsets of a given word. The relevant definitions are given below.

Definition 3.4.10. *Two multisets of queries $\mathcal{T}_1$ and $\mathcal{T}_2$ are said to be **multiplicity equivalent** if all player-question combinations occur with the same multiplicity in both sets. That is,*

$\mathcal{T}_1$ and $\mathcal{T}_2$ are multiplicity equivalent iff

$$\left|\left\{q \in \mathcal{T}_1 : q^{(\alpha)} = j\right\}\right| = \left|\left\{q' \in \mathcal{T}_2 : q'^{(\alpha)} = j\right\}\right| all \alpha, j. \tag{3.4.34}$$

Definition 3.4.11. *Given a word contained in a game $\mathcal{G}$*

$$W = q_{i_1} q_{i_2} ... q_{i_\ell} \tag{3.4.35}$$

define its even and odd multisets $\mathcal{E}$ and $\mathcal{O}$ in the natural way, so[8]

$$\mathcal{E} := \biguplus_{x \text{ even}} q_{i_x} \qquad and \qquad \mathcal{O} := \biguplus_{x \text{ odd}} q_{i_x}. \tag{3.4.36}$$

The key feature of the multiplicity equivalence condition is that a word contained in a game $\mathcal{G}$ is $\overset{p}{\sim} I$ iff its even and odd multisets are multiplicity equivalent. A slightly more general form of this statement is proved below.

Lemma 3.4.12. *Given two words W_1 and W_2 contained in $\mathcal{G}$, the following are equivalent:*

1. *$W_1 \overset{p}{\sim} W_2$.*
2. *The even and odd multisets of the word $W_1 W_2^{-1}$ are multiplicity equivalent.*

Proof. This proof is easiest if we generalize from the concept of even and odd multisets of clauses to even and odd multisets of variable-player combinations. In particular, given a word W (not necessarily contained in a game $\mathcal{G}$), its even and odd variable multisets are defined by

$$\mathcal{E}'(W) := \biguplus_{i \text{ even}, \alpha} (w_{i,\alpha}, \alpha) \tag{3.4.37}$$

$$\mathcal{O}'(W) := \biguplus_{i \text{ odd}, \alpha} (w_{i,\alpha}, \alpha) \tag{3.4.38}$$

where the tuple notion tracks the fact that variables given to different players are treated as distinct. To prove Lemma 3.4.12, we must now claim some basic facts about $\mathcal{E}'$ and $\mathcal{O}'$.

[8] Here and beyond we use the multiset operation $\biguplus$ to indicate union with addition of multiplicities. When applied to single elements we mean to treat each element as a single-element multiset.

(A) For a word W contained in $\mathcal{G}$, the even and odd multisets of W are multiplicity equivalent iff

$$\mathcal{E}'(W) = \mathcal{O}'(W). \tag{3.4.39}$$

(B) Applying a parity preserving permutation to a word W does not change $\mathcal{E}'(W)$ or $\mathcal{O}'(W)$.

(C) For any two words $W_1 \sim W_2$, we have

$$\mathcal{E}'(W_1) \uplus \mathcal{O}'(W_2) = \mathcal{E}'(W_2) \uplus \mathcal{O}'(W_1). \tag{3.4.40}$$

Claims (A) and (B) come directly from the definition of $\mathcal{E}'$ and $\mathcal{O}'$. To prove claim (C) we consider two words $W_1 \sim W_2$. If we never used a cancellation relation, we would immediately have

$$\mathcal{E}'(W_1) = \mathcal{E}'(W_2) \text{ and } \mathcal{O}'(W_1) = \mathcal{O}'(W_2). \tag{3.4.41}$$

Now a cancellation on a word always occurs between an element at an even position and one at an odd one, that is, it removes elements equally from $\mathcal{E}'$ and $\mathcal{O}'$. Letting $\mathcal{C}_1$ be the multiset of elements removed from $\mathcal{E}'(W_1)$ (and equivalently $\mathcal{O}'(W_1)$) by cancellation, with $\mathcal{C}_2$ defined similarly for W_2, we find

$$(\mathcal{E}'(W_1)\backslash\mathcal{C}_1) \uplus (\mathcal{O}'(W_2)\backslash\mathcal{C}_2) = (\mathcal{E}'(W_2)\backslash\mathcal{C}_2) \uplus (\mathcal{O}'(W_1)\backslash\mathcal{C}_1) \tag{3.4.42}$$

$$\Leftrightarrow \mathcal{E}'(W_1) \uplus \mathcal{O}'(W_2) = \mathcal{E}'(W_2) \uplus \mathcal{O}'(W_1). \tag{3.4.43}$$

Now, to prove Lemma 3.4.12 we note

$$W_1 \overset{p}{\sim} W_2 \tag{3.4.44}$$

$$\Leftrightarrow \exists \text{ parity preserving } \pi : \pi(W_1) \sim W_2 \quad \text{(definition)} \tag{3.4.45}$$

$$\Leftrightarrow \mathcal{E}'(\pi(W_1)) \uplus \mathcal{O}'(W_2) = \mathcal{E}'(W_2) \uplus \mathcal{O}'(\pi(W_1)) \quad (C) \tag{3.4.46}$$

$$\Leftrightarrow \mathcal{E}'(W_1) \uplus \mathcal{O}'(W_2) = \mathcal{E}'(W_2) \uplus \mathcal{O}'(W_1) \quad (B) \tag{3.4.47}$$

$$\Leftrightarrow \mathcal{E}'(W_1 W_2^{-1}) = \mathcal{O}'(W_1 W_2^{-1}) \quad \text{(reordering word)} \tag{3.4.48}$$

$$\Leftrightarrow \text{ The even and odd subsets of } W_1 W_2^{-1} \text{ are multiplicity equivalent.} \quad (A) \tag{3.4.49}$$

(3.4.48) is a somewhat subtle step, but follows formally (for example) from a proof by cases considering even and odd length words W_1 and W_2 and noting that the length of W_1 and W_2 must be equivalent mod 2. □

Definition 3.3.21. A game $\mathcal{G}$ contains a **Parity-Permuted Refutation (PREF)** if the queries of the game can be combined to form two multiplicity equivalent multisets for which the parity bits corresponding to the queries multiply to -1. Equivalently (Lemma 3.4.12), the game $\mathcal{G}$ contains a word which is $\overset{p}{\sim} I$ with sign -1.

The set of **PREF Games** are the set of XOR games that contain PREFs. The set of **noPREF Games** are the set of XOR games that do not.

We can finally restate and prove our necessary condition formally:

Theorem 3.3.22 (Necessary condition for refutation). If a game $\mathcal{G}$ admits a refutation, it contains a PREF.

Proof. By definition, a refutation R admitted by game $\mathcal{G}$ must be $\sim I$ and therefore $R \overset{p}{\sim} I$. R must also have sign -1. By Definition 3.3.21, game $\mathcal{G}$ then contains a PREF. □

Phrasing this necessary condition in terms of even and odd multiplicity equivalent multisets then provides an efficient means of computing whether or not a game satisfies this PREF criterion (Section 3.4.3).

We next consider the structural requirements on refutations to derive a stronger condition that is *sufficient* for a game to admit a refutation. As a first step we show that we can map between words which are $\overset{p}{\sim}$ to each other using a restricted class of permutations.

Lemma 3.4.13. *Let $W = w_1 w_2 \dots w_{2\ell}$ be a 1-XOR word of even length such that $W \overset{p}{\sim} I$; i.e. there exists a parity-preserving permutation $\pi \in S_{2\ell}$ such that $\pi(W) \sim I$. Then there exists a permutation $\pi' \in S_{2\ell}$, also satisfying $\pi'(W) \sim I$, also parity-preserving, and with an additional "pair preserving" property. This means that it permutes the pairs $(1,2),(3,4),\dots,(2\ell-1,2\ell)$ without separating or reordering the elements in each pair:*

$$\pi'(2i-1) = \pi'(2i) - 1 \qquad \forall i \in [\ell]. \tag{3.4.50}$$

Proof. Every letter in $\pi(W)$ will cancel with a unique other letter. We call a letter even or odd based on the parity of its location in $\pi(W)$. Deleting a canceled pair does not change the parity of any other location, and π also preserves the parities. Thus the letter in location $2i$ will cancel a letter in some odd position, which we call $2f(i)-1$ (i.e. $w_{2i} = w_{2f(i)-1}$). Since each odd letter cancels exactly one even letter, f is a permutation of $[\ell]$. Next we decompose f into disjoint cycles: $f = (i_1, i_2, \dots i_{\ell_1})(i_{\ell_1+1} \dots i_{\ell_2}) \dots (i_{\ell_{c-1}+1} \dots i_{\ell_c})$ where $\ell_c = \ell$. We claim that, written in two-line notation,

$$t' := \begin{pmatrix} 1 & 2 & \dots & \ell_c \\ i_1 & i_2 & \dots & i_{\ell_c} \end{pmatrix} \tag{3.4.51}$$

is a permutation of the pairs satisfying the desired properties. This map from f to t' is known as the Foata correspondence. Let π' be the corresponding pair-preserving permutation of $[2\ell]$.

Then

$$\pi'(w) = \underbrace{w_{2i_1-1}\, \overbrace{w_{2i_1} w_{2i_2-1}}\, \overbrace{w_{2i_2} \cdots \cdots w_{2i_{\ell_1}-1}}\, w_{2i_{\ell_1}}}\ \underbrace{w_{2i_{\ell_1+1}-1} \cdots w_{2i_{\ell_2}}}\,. \tag{3.4.52}$$

We can see that $\pi'(w)$ fully cancels following the pattern marked by the square brackets, with each cancellation using the fact that $w_{2i} = w_{2f(i)-1}$. □

A pair (and hence parity) preserving permutation $\pi' \in S_{2\ell}$ can be specified uniquely by some $\pi \in S_\ell$, given the relation

$$\pi(i) = \pi'(2i)/2. \tag{3.4.53}$$

We will frequently use of this alternate description of pair-preserving permutations, in a way made formal in Definition 3.4.18.

Before introducing this formally, we will the concept of a shuffle.

Definition 3.4.14. *A function $f : [\ell] \to [\ell]$ is called a shuffle function if the sequence*

$$f^{-1}(1), f^{-1}(2), \ldots, f^{-1}(\ell)$$

can be partitioned into two increasing subsequences. That is, for any shuffle function f, there exist disjoint increasing sequences s_A and s_B with $|s_A| + |s_B| = l$ and f^{-1} increasing on s_A and s_B.

Operationally, the set of shuffle functions are the set of permutations which can be obtained by partitioning the elements of $[\ell]$ into two sets, considering those sets as increasing sequences, and then mixing those sequences using a dovetail (riffle) shuffle.

Definition 3.4.15. *Let A be an arbitrary set, and let $t = (a_1, a_2, \ldots, a_\ell)$ be a sequence consisting of elements of A. Define the set of shuffles of t*

$$\text{shuffle}(t) := \{(a_{f(1)}, a_{f(2)}, \ldots, a_{f(\ell)}) : f \text{ a shuffle function}\} \tag{3.4.54}$$

and let this function act on sets in the natural way, so

$$\text{shuffle}(\mathcal{T}) := \bigcup_{t \in \mathcal{T}} \text{shuffle}(t) \tag{3.4.55}$$

where $\mathcal{T} \subseteq A^*$ *and* $A^* = \bigcup_{n \geq 0} A^n$ *is the set of all sequences of elements of* A*.*

Shuffles are a subset of the set of permutations. However, a standard result [3] regarding dovetail shuffles states that any permutation can be expressed as a short sequence of dovetail shuffles. Since our definition of shuffles contains a choice of partition that generalizes dovetail shuffles, the same result applies to our family of shuffles.

Lemma 3.4.16 (Theorem 1 of [3])**.** *Let* t *be any sequence of length* ℓ*,* $p \geq \lceil \log(\ell) \rceil$*, and let* t' *be any permutation of* t*. Then*

$$t' \in \text{shuffle}^p(t). \tag{3.4.56}$$

Our next goal is constructing a gadget from k-XOR clauses that allows us to shuffle pairs of letters on any wire of a word without changing the overall parity bit. The construction of this gadget relies on a simpler "shift gadget" which allows us to move words between wires. This definition and construction are given below.

Definition 3.4.17. *For any string of letters* $y = y_1 y_2 ... y_\ell$*, a* $1 \to 2$ ***shift gadget*** *for* y *is a word* $S^{1 \to 2}(y)$ *that equals the identity on all wires except the first two, and is equal to* $y^{-1} := y_\ell ... y_2 y_1$ *on wire* 1*, i.e. a word of the form*

$$q_{i_1} q_{i_2} \dots q_{i_\ell} := S^{1 \to 2}(y) \sim \begin{bmatrix} y^{-1} \\ h(y) \\ \\ \end{bmatrix}, \tag{3.4.57}$$

for some arbitrary string of letters $h(y)$*. For* $\alpha, \beta \in [k]$*, define* $\alpha \to \beta$ *shift gadgets analogously.*

Note that any shift gadget has a natural inverse

$$q_{i_\ell} q_{i_{\ell-1}} \dots q_{i_1} := S^{1\leftarrow 2}(y) \sim \begin{bmatrix} y \\ h(y)^{-1} \\ \\ \end{bmatrix} = \left(S^{1\to 2}(y)\right)^{-1}. \tag{3.4.58}$$

Intuitively, $S^{1\to 2}(y)$ removes y from the first wire and "saves" it on the second wire in the form of the string $h(y)$. $S^{1\leftarrow 2}(y)$ then "loads" y back onto the first wire while removing y' from the second wire. We now use these shift gadgets to construct a gadget that shuffles pairs of letters.

Definition 3.4.18. *Define* unpack $: ([n]^2)^{\ell/2} \to [n]^\ell$ *to map sequences of pairs into an "unpacked" sequence in the natural way, so that*

$$\text{unpack}\left((t_1, t_2), (t_3, t_4), \dots, (t_{\ell-1}, t_\ell)\right) = (t_1, t_2, \dots t_\ell). \tag{3.4.59}$$

Note that any permutation $\pi' \in S_\ell$ is pair preserving iff it satisfies

$$\pi' = \text{unpack} \circ \pi \circ \text{unpack}^{-1} \tag{3.4.60}$$

for some $\pi \in S_{\ell/2}$.

Lemma 3.4.19 (Shuffle Gadget)**.** *Let $t = (t_1, t_2, \dots t_{\ell/2})$ be a length $\ell/2$ sequence of pairs of letters, with each $t_i := (t_i^{(1)} t_i^{(2)}) \in [n]^2$. Let $\mathcal{G}$ be an XOR game that contains all shift gadgets in the set* [9]

$$\left\{ S^{1\to\alpha}(t_i^{(1)} t_i^{(2)}) : \alpha \in \{\alpha_1, \alpha_2\}, i \in [\ell/2] \right\}, \tag{3.4.61}$$

where $\alpha_1 \neq \alpha_2$ are elements of $[k] \setminus \{1\}$ and each shuffle gadget has length at most K. Then,

[9] There is nothing special about player 1 here but we state the lemma in terms of player 1 for notational simplicity.

for all $t' \in \text{shuffle}(t)$, $\mathcal{G}$ *contains a word* W *with sign* $s_W = +1$, *length at most* $K\ell$, *and*

$$W \sim \begin{bmatrix} \text{unpack}(t)^{-1} \, \text{unpack}(t') \\ \\ \end{bmatrix}.$$

Proof. Let f be the shuffle function satisfying $f(t) = t' \in \text{shuffle}(t)$. Since f is a shuffle function we can choose disjoint sequences s_A and s_B with $s_A \cup s_B = [\ell/2]$ and f^{-1} increasing on both. We construct a word of the desired form by first saving the pairs in s_A and s_B onto wires α_1 and α_2, respectively, then loading them back onto the first wire, interleaving in the appropriate order.

For any sequence s, let s^r be shorthand for that sequence written in reverse order. Define the function $g : [\ell/2] \to \{\alpha_1, \alpha_2\}$ by

$$g(i) = \begin{cases} \alpha_1 \text{ if } i \in s_A \\ \\ \alpha_2 \text{ if } i \in s_B \end{cases}.$$

Then the word W given below satisfies the lemma:

$$W = \prod_{i=1}^{\ell/2} \left(S^{1 \to g(i)}(t_i^{(1)} t_i^{(2)}) \right) \prod_{i=1}^{\ell/2} \left(S^{1 \leftarrow g(f^{-1}(i))}(t_{f^{-1}(i)}^{(1)} t_{f^{-1}(i)}^{(2)}) \right) \tag{3.4.62}$$

$$\sim \begin{bmatrix} \prod_{i \in s_{\ell/2}^r} (t_i^{(1)} t_i^{(2)})^{-1} \\ \prod_{i \in s_A^r} h(t_i^{(1)} t_i^{(2)}) \\ \prod_{i \in s_B^r} h(t_i^{(1)} t_i^{(2)}) \end{bmatrix} \begin{bmatrix} \prod_{i \in s_{\ell/2}} (t_{f^{-1}(i)}^{(1)} t_{f^{-1}(i)}^{(2)}) \\ \prod_{i \in s_A} h(t_i^{(1)} t_i^{(2)})^{-1} \\ \prod_{i \in s_B} h(t_i^{(1)} t_i^{(2)})^{-1} \end{bmatrix} = \begin{bmatrix} \text{unpack}(t)^{-1} \, \text{unpack}(t') \\ \\ \end{bmatrix}. \tag{3.4.63}$$

By assumption, $\mathcal{G}$ contains each shift gadget used in the construction of W, and each shift gadget has length at most K. Therefore W is contained in $\mathcal{G}$ and has length at most $2K(\ell/2) = K\ell$. For each shift gadget used, its inverse is also used. By construction, the sign of each shift gadget is the same as its inverse, so the overall sign of W is $s_W = +1$. □

Note 3.4.20. For any game $\mathcal{G}$ and sequence of pairs t that meets the conditions of Lemma 3.4.19, $\mathcal{G}$ will also meet the conditions for any sequence of pairs $u = \pi(t)$ produced through permutation π of t. Then, under the assumptions of Lemma 3.4.19 we get "for free" that a

word is contained in $\mathcal{G}$ with sign $+1$ and has the form

$$\begin{bmatrix} \text{unpack}(\pi(t))^{-1}\,\text{unpack}(f(\pi(t))) \\ \end{bmatrix} \tag{3.4.64}$$

with π any permutation on pairs and f any shuffle function (see Definition 3.4.14).

Combining our newly constructed shuffle gadget with our understanding of parity preserving permutations allows us to derive a set of sufficient conditions for a game $\mathcal{G}$ to contain a refutation. These will be used in a critical way in Section 3.4.3.

Lemma 3.4.21. *Let $\mathcal{G}$ be a k-XOR game containing a length ℓ word W whose first wire is given by*

$$W_1 = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1\ell} \end{bmatrix} \overset{p}{\sim} I. \tag{3.4.65}$$

Also let $\mathcal{G}$ contain all shift gadgets in the set

$$\{S^{1\to\alpha}(w_{1(2i-1)}w_{1(2i)}) : \alpha \in \{\alpha_1, \alpha_2\}, i \in [\ell/2]\}, \tag{3.4.66}$$

where $\alpha_1 \neq \alpha_2 \in [k]\backslash\{1\}$ and each gadget has length at most K.

Then $\mathcal{G}$ contains a word with sign $+1$ and length at most $K\ell\log(\ell)$ whose first wire is given by

$$W_1^{-1} = \begin{bmatrix} w_{1\ell} & w_{1(l-1)} & \dots & w_{11} \end{bmatrix}. \tag{3.4.67}$$

and which is $\sim I$ on all wires other than the first.

Proof. By Lemma 3.4.13 there exists a permutation π on $[\ell/2]$ satisfying

$$\begin{bmatrix} \text{unpack} \circ \pi((w_{11}w_{12}), (w_{13}w_{14}), \dots, (w_{1(l-1)}w_{1\ell})) \end{bmatrix} \sim I. \tag{3.4.68}$$

By Lemma 3.4.16, there then exists a sequence $(f_1, f_2, \dots f_p)$ of $p \leq \log(\ell)$ shuffle functions

with

$$f_p \dots f_2 f_1 = \pi. \tag{3.4.69}$$

Now let π' be an arbitrary permutation of $[\ell/2]$, f' be an arbitrary shuffle of $[\ell/2]$, and define the word $Y(\pi', f')$ to have first coordinate

$$Y_1(\pi', f') := \Big[\text{unpack} \circ \pi'((w_{11}w_{12}), \dots, (w_{1(l-1)}w_{1\ell}))\Big]^{-1} \Big[\text{unpack} \circ f'(\pi'((w_{11}w_{12}), \dots, (w_{1(l-1)}w_{1\ell})))\Big] \tag{3.4.70}$$

and all remaining $k-1$ coordinates the identity. By Lemma 3.4.19 and Note 3.4.20, we have that $\mathcal{G}$ contains a word with sign $+1$ and length at most $K\ell$ which is $\sim Y(\pi', f')$.

By concatenating a carefully chosen string of these words, we see $\mathcal{G}$ also contains a word with sign $+1$ and length at most $K\ell \log(\ell)$ which is

$$\sim Y(e, f_1)Y(f_1, f_2)Y(f_2 f_1, f_3) \dots Y(f_{k-1} f_{k-2} \dots f_1, f_k) \sim W_1^{-1}. \tag{3.4.71}$$

□

Lemma 3.4.21 suggests we can construct refutations for a game $\mathcal{G}$ by finding a word contained in $\mathcal{G}$ which is $\overset{p}{\sim} I$ and has sign -1, and then checking to see if $\mathcal{G}$ contains the necessary shift gadgets. First, we demonstrate that the first two wires of some permutation of such a word can be made to cancel without using any shift gadgets, then determine a sufficient set of shift gadgets required thereafter.

Lemma 3.4.22. *Let game $\mathcal{G}$ contain word $W' = q_{i_1} q_{i_2} \dots q_{i_\ell} \overset{p}{\sim} I$. There exists a permutation $\pi \in S_\ell$ such that*

$$W := q_{i_{\pi(1)}} q_{i_{\pi(2)}} \dots q_{i_{\pi(\ell)}} \overset{p}{\sim} I \tag{3.4.72}$$

and both $W^{(1)} \sim I$ and $W^{(2)} \sim I$ with $W^{(2)} = x_1 x_1 x_2 x_2 \dots x_{\ell/2} x_{\ell/2}$ where $x_i \in [n]$.

Proof. By Lemma 3.4.12, we have that the even and odd multisets of W', $\mathcal{E}$ and $\mathcal{O}$ respectively, are multiplicity equivalent. Thus, for each $\alpha \in [k]$, there exists a bijection $f_\alpha : \mathcal{E} \mapsto \mathcal{O}$ that maps a query $(q^{(1)}, \dots, q^{(k)}) \in \mathcal{E}$ to a query $(q'^{(1)}, \dots, q'^{(k)}) \in \mathcal{O}$ such that $q^{(\alpha)} = q'^{(\alpha)}$. From

the bijections f_1, f_2, we will define a new map $f : \mathcal{E} \cup \mathcal{O} \mapsto \mathcal{E} \cup \mathcal{O}$ that maps each query $q \in \mathcal{E}$ to $f_1(q) \in \mathcal{O}$ and each $q' \in \mathcal{O}$ to $f_2^{-1}(q') \in \mathcal{E}$. Since f_1 and f_2 are bijections, so is f. Applying the Foata correspondence, as in Lemma 3.4.13, to the permutation of $\mathcal{E} \cup \mathcal{O}$ associated with f yields a sequence of queries that make a word W with the property that the first two wires completely cancel to identity and wire 2 takes the desired form, i.e.

$$W = \begin{bmatrix} \overbrace{w_{11}\, w_{12} \cdot \cdots \cdot w_{1(\ell-1)}}\, w_{1\ell} \\ w_{21} w_{22} \cdot \cdots \cdot \underbrace{w_{2(\ell-1)}} w_{2\ell} \\ W^{(3)} \\ \cdots \\ W^{(k)} \end{bmatrix} \sim \begin{bmatrix} \\ W^{(3)} \\ \cdots \\ W^{(k)} \end{bmatrix},$$

where $W^{(3)}, \ldots, W^{(k)}$ are even-length strings of letters. □

For a refutation to exist we then simply need to be able to shuffle the pairs on the remaining wires $3, \ldots, k$.

Theorem 3.4.23 (Sufficient condition for refutation)**.** *Let $\mathcal{G}$ be a PR game which by definition contains some word $W' \overset{p}{\sim} I$ of some even length ℓ. Let $W \overset{p}{\sim} I$ be the pairwise permuted word as in Lemma 3.4.22. If $\mathcal{G}$ contains all shift gadgets in the set*

$$\left\{ S^{\alpha \to \alpha'}(W_{2i-1}^{(\alpha)} W_{2i}^{(\alpha)}) : \alpha \in \{3, \ldots, k\}, \alpha' \in \{1, 2\}, i \in [\ell/2] \right\} \tag{3.4.73}$$

then $\mathcal{G}$ contains a refutation.

Proof. By the definition of a PR game (Definition 3.3.21), $\mathcal{G}$ contains the word W with sign -1. By Lemma 3.4.21, $\mathcal{G}$ contains all words W''_α with α-th wire given by $(W''_\alpha)^{(\alpha)} = \left(W^{(\alpha)}\right)^{-1}$, all other wires $\sim I$, and sign $+1$. Therefore $\mathcal{G}$ contains the word

$$R := W \prod_\alpha W''_\alpha \sim I \tag{3.4.74}$$

with sign $s_R = -1$, which is a refutation. □

It turns out that for the special case of symmetric XOR games, the symmetric structure guarantees existence of all required shift gadgets automatically. Theorems 3.4.23 and 3.3.22 then give that a symmetric game contains a PREF if and only if it contains a refutation. Further, whether a game contains a PREF is an efficiently decidable criterion. A formal definition of symmetric XOR games and a proof of these facts are demonstrated in Section 3.4.3.

3.4.3 Algorithm for Symmetric Games

We begin with a formal definition of symmetric games.

Definition 3.4.24. *A k-XOR game $\mathcal{G}$ is **symmetric** if whenever it contains a clause $c = (q^{(1)}, q^{(2)}, \ldots, q^{(k)}, s)$, it also contains all clauses $c' = (q^{(\pi(1))}, q^{(\pi(2))}, \ldots, q^{(\pi(k))}, s)$, where $\pi : [k] \mapsto [k]$ permutes the query while the parity bit s is unchanged.*

Definition 3.4.25. *A **random symmetric k-XOR game** $\mathcal{G}_{sym}$ on $m = k!m'$ clauses is a game constructed by randomly generating m' clauses, then including all clauses related by permutations (as above) in $\mathcal{G}_{sym}$.*

For symmetric games, we can now prove that all required shift gadgets are certainly included.

Lemma 3.4.26. *Let W be a word contained in symmetric game $\mathcal{G}$ of even length ℓ with second wire of the form $W^{(2)} = x_1x_1x_2x_2 \ldots x_\ell x_\ell$, where $x_i \in [n]$. For any wire $\alpha \in \{3, \ldots, k\}$ and pairs of letters y_1, y_2 that appear at adjacent positions $2i-1, 2i$ in $W^{(\alpha)}$, there exists shift gadgets from $\alpha \to 2$ and from $\alpha \to 1$ for y_1y_2 with length $O(1)$.*

Proof. Since the game is symmetric, it suffices to show the existence of the gadget for $\alpha \to 2$. Let the queries containing y_1, y_2 in W be $q_1 = (q_1^{(1)}, q_1^{(2)}, \ldots, y_1, \ldots)$ and $q_2 = (q_2^{(1)}, q_2^{(2)}, \ldots, y_2, \ldots)$, respectively. Then by the assumption of symmetry, all permutations of these queries exist in the given game. We can thus construct the shift gadget $S^{\alpha \to 2}(y_1y_2)$ by

the product of four clauses as follows:

$$S^{\alpha\to 2}(y_1y_2) = \begin{bmatrix} q_2^{(1)} \\ q_2^{(2)} \\ \dots \\ y_2 \\ \dots \end{bmatrix} \begin{bmatrix} q_2^{(1)} \\ y_2 \\ \dots \\ q_2^{(2)} \\ \dots \end{bmatrix} \begin{bmatrix} q_1^{(1)} \\ y_1 \\ \dots \\ q_1^{(2)} \\ \dots \end{bmatrix} \begin{bmatrix} q_1^{(1)} \\ q_1^{(2)} \\ \dots \\ y_1 \\ \dots \end{bmatrix} = \begin{bmatrix} \\ h(y_1y_2) \\ \\ y_2y_1 \\ \\ \end{bmatrix}, \tag{3.4.75}$$

where $h(y_1y_2) := q_2^{(2)}y_2y_1q_1^{(2)}$ and the equality holds because y_1 and y_2 appear at an odd and following even position of W so by the form of the second wire $q_1^{(2)} = q_2^{(2)}$. □

We now prove Theorem 3.2.1, by showing that the PREF criterion is both necessary and sufficient for a symmetric game to have a refutation, and can also be expressed as a system of linear Diophantine equations and thus solved efficiently.

Theorem 3.2.1. There exists an algorithm that, given a k-player symmetric XOR game $\mathcal{G}$ with alphabet size n and m clauses, decides in time $\mathsf{poly}(n, m)$ whether $\omega^*(\mathcal{G}) = 1$ or $\omega^*(\mathcal{G}) < 1$.

Proof. By Theorem 3.4.1, deciding whether the commuting-operator value is 1 is equivalent to deciding whether the game admits a refutation (of any length). By Theorem 3.3.22 for a game to admit a refutation it is necessary that it contains a PREF. Further, Theorem 3.4.23 and Lemma 3.4.26 together show that for a symmetric game to admit a refutation it is also sufficient to contain a PREF. Thus for a symmetric game, deciding whether $\omega^* = 1$ reduces to determining whether or not the game contains a PREF.

We can now rephrase the condition for a game to contain a PREF as a system of linear Diophantine equations. For each query in the game $q_i = (q_i^{(1)}, \dots, q_i^{(k)})$, let z_i be an integer-valued variable representing the number of times query i appears in the even multiset of the PREF minus the number of times it appears in the odd multiset. The condition that these z_i in fact correspond to multiplicity equivalent sets can then be stated as a system of linear Diophantine equations,

$$A^T z = 0 \tag{3.4.76}$$

where A is the game matrix as defined in Definition 3.3.3 and we have collected the z_i into a

vector $z \in \mathbb{Z}^m$. The condition that the signs of the clauses in the PREF multiply to -1 can be expressed as an additional linear Diophantine equation in terms of z and parity bit vector $\hat{s}$ (Definition 3.3.3):

$$\hat{s}^T z = 1 \pmod 2. \tag{3.4.77}$$

By applying a standard algorithm, such as the one described in Chapter 5 of [58], this system can be solved in time polynomial in the size of the system, measured as the number of bits necessary to specify the system of equations. This means a runtime that is $\mathsf{poly}(n, m)$.

□

Note 3.4.27. Finding a solution to (3.4.76) and (3.4.77) tells us not only that a refutation exists but also bounds its length. In particular, by following the steps of the preceding proof, it can be shown that for a symmetric game with $\omega^*(\mathcal{G}) < 1$, the minimum-length refutation has length L satisfying

$$\Omega(\|z\|_1) \leq L \leq O(k\|z\|_1 \log \|z\|_1),$$

where z is a solution to (3.4.76) and (3.4.77) minimizing $\|z\|_1$.

Finally, note that this linear algebraic description of the necessary PREF criterion for an entangled refutation parallels the classical condition for refutation (Definition 3.3.14). The only distinction is that (3.4.76) is considered an equation over $\mathbb{F}_2$ for classical games and over $\mathbb{Z}$ for entangled games. As described in Section 3.5.3, these Diophantine equations then give rise to a dual condition similar to the classical picture: a MERP strategy achieves value 1 exactly when these equations do not admit a solution.

3.5 MERP Strategies

Section 3.3.4 introduced the family of Maximal Entanglement, Relative Phase (MERP) strategies. The primary feature of the MERP strategies is that deciding whether $v^{\text{MERP}} = 1$ and computing the accompanying MERP strategy vector may be done efficiently via Gaussian elimination. Beyond computability, the MERP strategies actually achieve value 1 on a large class of games where that is possible. Specifically, MERP achieves value 1 exactly where a PREF does not exist (noPREF games), including all symmetric value 1 games. This MERP -

PREF duality is analogous to the duality between a classical linear algebraic refutation and the construction of a classical value 1 strategy.

Here, we motivate the definition of MERP strategies (Section 3.5.1) and prove their value defined in Claim 3.3.27 (Section 3.5.2). We then investigate the duality between MERP value 1 and PREFs (Section 3.5.3).

3.5.1 Generalizing GHZ

The MERP family of strategies is motivated by the GHZ strategy for solving the GHZ game. We begin by reviewing the GHZ game and value 1 strategy.

Definition 3.5.1. *Recall that the* ***GHZ game*** *is defined by the clauses*

$$\mathcal{G}_{GHZ} := \left\{ \begin{bmatrix} x \\ x \\ x \\ +1 \end{bmatrix}, \begin{bmatrix} y \\ y \\ x \\ -1 \end{bmatrix}, \begin{bmatrix} y \\ x \\ y \\ -1 \end{bmatrix}, \begin{bmatrix} x \\ y \\ y \\ -1 \end{bmatrix} \right\}. \tag{3.5.1}$$

The GHZ strategy [26], defined as follows, achieves value 1 for this game.

Definition 3.5.2. *Define the* ***GHZ Strategy*** *for $\mathcal{G}_{GHZ}$ to be the tensor-product strategy in which:*

1. *The $k = 3$ players share the maximally entangled state*

$$|\Psi\rangle = \frac{1}{\sqrt{2}} \left[|000\rangle + |111\rangle \right] \tag{3.5.2}$$

   with player α having access to the α-th qubit of the state.

2. *Upon receiving symbol j from the verifier, player α rotates his qubit by an angle*

$$\theta(\alpha, j) = \begin{cases} 0 \ \text{if } j = x \\ \frac{\pi}{2} \ \text{if } j = y \end{cases} \tag{3.5.3}$$

about the Z axis, then measures his qubit in the ± basis and sends his observed outcome to the verifier. Defining the states $|\theta(\alpha,j)_\pm\rangle$ *by*

$$|\theta(\alpha,j)_+\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle + e^{i\theta(\alpha,j)}\,|1\rangle\right] \quad \textit{and} \tag{3.5.4}$$

$$|\theta(\alpha,j)_-\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle - e^{i\theta(\alpha,j)}\,|1\rangle\right] \tag{3.5.5}$$

the GHZ strategy may be given by the strategy observables

$$X_j^{(\alpha)} := |\theta(\alpha,j)_+\rangle\langle\theta(\alpha,j)_+| - |\theta(\alpha,j)_-\rangle\langle\theta(\alpha,j)_-|\,. \tag{3.5.6}$$

We now consider why this strategy is successful. Recall that a φ rotation in the Z basis is represented by the operator

$$e^{i\varphi/2}\,|0\rangle\langle 0| + e^{-i\varphi/2}\,|1\rangle\langle 1|\,. \tag{3.5.7}$$

Thus the rotations $\varphi_1,\varphi_2,\varphi_3$ applied by the players to their shared state $|\Psi\rangle$ results in

$$|\Psi_\varphi\rangle := \frac{1}{\sqrt{2}}\left[e^{-i\frac{\varphi}{2}}\,|000\rangle + e^{i\frac{\varphi}{2}}\,|111\rangle\right] \tag{3.5.8}$$

$$\varphi := \varphi_1+\varphi_2+\varphi_3. \tag{3.5.9}$$

Let $\sigma_X = \left(\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right)$ be the matrix corresponding to a measurement in the ± basis, and note that $\sigma_X\otimes\sigma_X\otimes\sigma_X\,|\Psi_\varphi\rangle = |\Psi_{-\varphi}\rangle$. This gives expected value of the measurements performed by the three players,

$$\langle\Psi_\varphi|\,X\otimes X\otimes X\,|\Psi_\varphi\rangle = \frac{e^{i\varphi}+e^{-i\varphi}}{2} = \cos\varphi. \tag{3.5.10}$$

Thus the *relative phase* between the kets $|000\rangle$ and $|111\rangle$ introduced by the Z rotations determines the probabilities that the players output $+1$ or -1. For the GHZ game, the prescription for Z rotations given in (3.5.3) results in relative phase $\varphi = 0$ for the first clause and $\varphi = \pi$ for the remaining three clauses, exactly matching the desired outputs.

This description of GHZ motivates the MERP family as a generalization. For a game $\mathcal{G}$, the MERP construction assigns a distinct angle to each player-question combination such

that the relative phase for each query in $\mathcal{G}$ gives optimal probability of winning. The set of games for which MERP can achieve value 1 is exactly the set for which the game admits independently setting the relative phase for each query to $\pi\hat{s}_i$. This is exactly the statement of Claim 3.3.29.

We proceed by recalling the definition of a MERP strategy in light of the GHZ analogue, proving our value claim, and finally demonstrating the duality with PREF games.

3.5.2 MERP Strategy Value

Recall the definition of a MERP strategy:

Definition 3.3.26. Given a k-XOR game $\mathcal{G}$ with m clauses, a **MERP strategy** for $\mathcal{G}$ is a tensor-product strategy in which:

1. The k players share the maximally entangled state

$$|\Psi\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle^{\otimes k} + |1\rangle^{\otimes k}\right] \tag{3.5.11}$$

   with player α having access to the α-th qubit of the state.

2. Upon receiving question j from the verifier, player α rotates his qubit by an angle $\theta(\alpha, j)$ about the Z axis, then measures his qubit in the X basis and sends his observed outcome to the verifier.

   Explicitly, we define the states

$$|\theta(\alpha,j)_\pm\rangle := \frac{1}{\sqrt{2}}\left[|1\rangle \pm e^{-i\theta(\alpha,j)}|0\rangle\right] \tag{3.5.12}$$

   and pick strategy observables

$$X_j^{(\alpha)} := |\theta(\alpha,j)_+\rangle\langle\theta(\alpha,j)_+| - |\theta(\alpha,j)_-\rangle\langle\theta(\alpha,j)_-|\,. \tag{3.5.13}$$

We now demonstrate that a MERP strategy achieves the claimed tensor-product (and thus commuting operator) value by explicit calculation.

Claim 3.3.27. The value achieved by that MERP strategy on game $\mathcal{G}$ is:

$$v^{\text{MERP}}(G,\hat{\theta}) := \frac{1}{2} + \frac{1}{2m}\left(\sum_{i=1}^{m} \cos\left(\pi\left[(A\hat{\theta})_i - \hat{s}_i\right]\right)\right) \tag{3.5.14}$$

$$= \frac{1}{2} + \frac{1}{2m}\left(\sum_{i=1}^{m} \cos\left(\sum_{\alpha=1}^{k} \theta(\alpha, q_i^{(\alpha)}) - \pi\hat{s}_i\right)\right). \tag{3.5.15}$$

Proof. Consider a particular clause $c_i = (q_i, s_i)$. We calculate the probability that a MERP strategy parameterized by $\theta(\alpha, q_i^{(\alpha)})$ returns output s_i correctly.

If players $1, \ldots, k$ apply rotations by $\varphi_1, \ldots, \varphi_k$ to their qubits in state $|\Psi\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle^{\otimes k} + |1\rangle^{\otimes k}\right]$ then they will be left with

$$\left|\Psi_{\varphi}^{k}\right\rangle := \frac{1}{\sqrt{2}}\left[e^{i\frac{\varphi}{2}}|0\rangle^{\otimes k} + e^{-i\frac{\varphi}{2}}|1\rangle^{\otimes k}\right] \tag{3.5.16}$$

$$\varphi := \varphi_1 + \ldots + \varphi_k. \tag{3.5.17}$$

Note that $X^{\otimes k}\left|\Psi_{\varphi}^{k}\right\rangle = \left|\Psi_{-\varphi}^{k}\right\rangle$. The expected value of the product of the k measurements is then

$$\left\langle\Psi_{\varphi}^{k}\right| X^{\otimes k}\left|\Psi_{\varphi}^{k}\right\rangle = \frac{e^{i\varphi} + e^{-i\varphi}}{2} = \cos\varphi. \tag{3.5.18}$$

We now plug in the values from the clause and the corresponding angles in the MERP strategy. The angles are $\varphi_\alpha = \theta(\alpha, q_i^\alpha)$ so that

$$\varphi = \sum_{\alpha\in[k]} \theta(\alpha, q_i^\alpha) = (A\hat{\theta})_i. \tag{3.5.19}$$

The probability of obtaining the correct answer for the clause is

$$\frac{1 + s_i\left\langle\Psi_{\varphi}^{k}\right| X^{\otimes k}\left|\Psi_{\varphi}^{k}\right\rangle}{2} = \frac{1 + s_i\cos(\varphi)}{2} = \frac{1 + \cos(\varphi - \pi\hat{s}_i)}{2}. \tag{3.5.20}$$

Averaging over all clauses and substituting (3.5.19) for φ we obtain (3.5.14) and (3.5.15). □

3.5.3 MERP - PREF Duality

It is well-known that the structure of the game matrix over $\mathbb{F}_2$ gives insight into the classical value of an XOR game. The construction of a classical value 1 strategy is dual to the existence of a classical refutation. In much the same way, the construction of a commuting operator value 1 MERP strategy is dual to the existence of a PREF.

MERP is restricted to achieving value 1 on only a subset of commuting operator value 1 XOR games. By the duality to PREF this subset is exactly those games that our algorithm can decide have value 1. In particular, this means that all symmetric games with value 1 can be played optimally using MERP, making it a powerful family of strategies.

We begin with a review of the classical value 1 - refutation duality, which informs our later proof of the MERP - PREF duality. From Claim 3.3.7, we have the value of a classical strategy

$$v(G,\hat{\eta}) = \frac{1}{2} + \frac{1}{2m}\left(\sum_i \cos(\pi\left[(A\hat{\eta})_i - \hat{s}_i\right])\right) \tag{3.5.21}$$

where the vector algebra is taken over $\mathbb{F}_2$. Using this linear algebraic form for the value, we can prove Claim 3.3.9.

Claim 3.3.9. Every solution $\hat{\eta} \in \mathbb{F}_2^{kn}$ to

$$A\hat{\eta} = \hat{s} \tag{3.3.9}$$

corresponds to a strategy η achieving value 1 on game $G \sim (A, \hat{s})$, and vice versa. In particular, a game $\mathcal{G}$ has classical value 1 iff (3.3.9) has a solution.

Proof. If a solution $\hat{\eta}$ exists,

$$v(G,\hat{\eta}) = \frac{1}{2} + \frac{1}{2m}\left(\sum_i \cos(\pi\left[(A\hat{\eta})_i - \hat{s}_i\right])\right) \tag{3.5.22}$$

$$= \frac{1}{2} + \frac{1}{2m}\left(\sum_i \cos(0)\right) = 1. \tag{3.5.23}$$

Conversely, to achieve value 1, we must have the argument of every cosine equal to some

multiple of 2π. Therefore we need $A\hat{\eta} - \hat{s} = 0$ over $\mathbb{F}_2$. □

Recall that this classical value 1 constraint has a dual set of equations, such that there exists a classical *refutation* that solves the dual equations if and only if the classical value 1 constraints are not satisfiable.

Fact 3.3.15. Either a classical refutation y exists satisfying

$$\begin{bmatrix} A^T \\ \hat{s}^T \end{bmatrix} y = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{3.3.13}$$

or a classical strategy $\hat{\eta}$ exists satisfying (3.3.9).

We use an analogous duality relation to prove the MERP - PREF duality shortly.

Before that, we mention one more consequence of this characterization of classical value 1 games – a linear algebraic specification, in terms of game matrix A, of the set of $\hat{s}$ for which the game $G \sim (A, \hat{s})$ has $\omega(\mathcal{G}) = 1$.

Definition 3.5.3. *Define the vector space $\mathcal{Y}_2 \subseteq \mathbb{F}_2^m$ by*

$$\mathcal{Y}_2 := \left\{ A\hat{\eta} : \hat{\eta} \in \mathbb{F}_2^{kn} \right\} = \mathrm{im}_{\mathbb{F}_2}(A) \tag{3.5.24}$$

Define the dimension of this vector space as

$$\sigma_2 := \dim \mathcal{Y}_2. \tag{3.5.25}$$

Corollary 3.5.4. *Given a game matrix A, the set of possible accompanying $\hat{s}$ that produce a game $G \sim (A, \hat{s})$ with classical value 1 is exactly the 2^{σ_2} parity-bit vectors in $\mathcal{Y}_2$.*

Proof. This follows immediately from Claim 3.3.9. □

The main use of Corollary 3.5.4 is to characterize the classical value of games with randomly chosen s_i (Section 5.2.3).

We now use an analogue of Fact 3.3.15 to demonstrate that the set of games on which MERP achieves commuting operator value 1 is exactly the complement of those for which a PREF specification exists. First, recall the MERP constraint equations that define the set of games for which MERP achieves value 1.

Claim 3.3.29. A MERP strategy achieves $v^{\text{MERP}} = 1$ on a game $\mathcal{G}$ iff its MERP constraint equations

$$A\hat{\theta} = \hat{s} \pmod 2 \tag{3.5.26}$$

have a solution $\hat{\theta} \in \mathbb{Q}^{kn}$.

Proof. If a solution $\hat{\theta}$ exists, (3.5.14) gives the MERP value using this strategy vector:

$$v^{\text{MERP}}(G, \hat{\theta}) = \frac{1}{2} + \frac{1}{2m}\left(\sum_{i=1}^{m} \cos(0)\right) = 1. \tag{3.5.27}$$

Conversely, the only way to achieve value m inside the sum over cosines is for the argument to each cosine to be a multiple of 2π. This is only possible if $(A\hat{\theta})_i - \hat{s}_i = 0 \pmod 2$ for each i. □

Theorem 3.3.30. Either there exists a MERP refutation $z \in \mathbb{Z}^m$ satisfying the PREF equations

$$A^T z = 0 \tag{3.5.28}$$

$$\hat{s}^T z = 1 \pmod 2 \tag{3.5.29}$$

or a MERP strategy with value 1 exists for game $G \sim (A, \hat{s})$.

Proof. We begin by reformatting the linear Diophantine equations (3.5.28) and (3.5.29) to remove the modulo 2 and collect the PREF constraints into a single matrix equation

$$\begin{bmatrix} A^T & 0 \\ \hat{s}^T & 2 \end{bmatrix} \begin{bmatrix} z \\ z' \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{3.5.30}$$

with $z' \in \mathbb{Z}$.

By [58, Corollary 4.1a], the dual to (3.5.30) is the system of constraints

$$\begin{bmatrix} A & \hat{s} \\ 0 & 2 \end{bmatrix} \begin{bmatrix} w \\ w' \end{bmatrix} \in \mathbb{Z}^{m+1} \qquad \text{and} \tag{3.5.31}$$

$$\begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} w \\ w' \end{bmatrix} \notin \mathbb{Z}. \tag{3.5.32}$$

Here "dual" means that (3.5.31) and (3.5.32) are satisfiable iff (3.5.30) is unsatisfiable. The bottom rows of (3.5.31) and (3.5.32) can be satisfied iff

$$w' = a + \frac{1}{2},\ a \in \mathbb{Z}. \tag{3.5.33}$$

The top row of (3.5.31) then becomes:

$$Aw + \hat{s}w' = a' \in \mathbb{Z}^m \tag{3.5.34}$$

$$\Leftrightarrow A(2w) + (2a+1)\hat{s} = 2a' \tag{3.5.35}$$

$$\Leftrightarrow A(2w) = \hat{s} \pmod 2. \tag{3.5.36}$$

Setting $\hat{\theta} = 2w$ and picking arbitrary $a \in \mathbb{Z}$, (3.5.33) and (3.5.36) can be satisfied iff there is a solution to

$$A\hat{\theta} = \hat{s} \pmod 2,\ \hat{\theta} \in \mathbb{Q}^{kn}. \tag{3.5.37}$$

□

Theorem 3.3.30 tells us that every game that we can decide has value 1 using the algorithm of Section 3.4.3 also has an accompanying MERP strategy with value 1. Further, we demonstrated in that section that a symmetric game contains a PREF iff it has value $\omega^* < 1$. We conclude that for symmetric games, the MERP family of strategies achieves value 1 everywhere it is possible to do so.

Following the classical case, it is also illuminating to note a linear algebraic specification, in terms of game matrix A, of the $\hat{s}$ for which a MERP strategy can achieve value 1 on game $G \sim (A, \hat{s})$. First, we define a mapping between the space in which the image of A lives,

$\text{im}_{\mathbb{Q}}(A) \subseteq \mathbb{Q}^m$, and the space in which the parity bits live, $\hat{s} \in \mathbb{F}_2^m$.

Definition 3.5.5. *Define a mapping*[10] $\varphi_2 : \mathbb{Q}^m \to \mathbb{F}_2^m$ *by*

$$\varphi_2(z) := \begin{cases} z \pmod 2 \text{ if } z \in \mathbb{Z}^m \\ 0 \text{ otherwise.} \end{cases}$$

Now, we can define an analogue to $\mathcal{Y}_2$, here considering A as a map over $\mathbb{Q}$ and naturally extending φ_2 to act on subsets of $\mathbb{Q}^m$.

Definition 3.5.6. *Define the vector space* $\mathcal{Y}_Q \subseteq \mathbb{F}_2^m$ *by*

$$\mathcal{Y}_Q := \varphi_2(\text{im}_{\mathbb{Q}}(A)). \tag{3.5.38}$$

We then find that, accounting for the φ_2 technicality due to the mod 2 involved in computing an overall output, the set of games with MERP value 1 is the image of A over $\mathbb{Q}$.

Corollary 3.5.7. *Given a game matrix A, the set of possible accompanying $\hat{s}$ that produce a game $G \sim (A, \hat{s})$ with MERP value 1 is exactly the parity bit vectors in $\mathcal{Y}_Q$.*

Proof. This follows directly from Claim 3.3.29. □

In this sense, Corollary 3.5.7 demonstrates that the advantage of MERP over a classical strategy is simply exploiting entanglement to enable the players to "output" values in $\mathbb{Q}$ instead of $\mathbb{F}_2$.

3.6 Chapter Summary

This chapter introduced two important new ideas for the study of XOR games. The first were PREFs, defined in Definition 3.3.21, which gave an efficiently checkable sufficient condition for XOR games to have a perfect commuting operator strategy. The second were MERP strategies, defined in Section 3.3.4 and elaborated on in Section 3.5.2, which gave a simple class of tensor product strategies for XOR games. Then, in Section 3.5.3 these two conditions were shown

[10] Note φ_2 is not in general a linear function, but it is linear over inputs in $\mathbb{Z}^m$.

to be related, with a MERP strategy existing for any game that had a perfect commuting operator by the PREF (really the noPREF) condition. We also explored some scenarios in which the noPREF condition was necessary and sufficient, showing in Sections 3.4.2 and 3.4.3 that the noPREF condition was a necessary and sufficient condition for symmetric XOR games to have a perfect commuting operator strategy. As a consequence, we saw that MERP strategies were optimal for these games.

In the next chapter we continue our study of XOR games by showing the noPREF condition is also necessary and sufficient for 3 player XOR games to have perfect commuting operator strategies. Before proving this result we recast the noPREF condition in more algebraic language, identifying with the subgroup membership characterization of perfect XOR games introduced in Chapter 2.

Chapter 4

3XOR Games

In this chapter we study 3 player XOR games with perfect commuting operator value. Our starting point is the characterization of games with perfect commuting operator value in terms of the subgroup membership problem introduced in Chapter 2. As in Chapter 3 we reprove this result in the special case of XOR games. This time our proof makes use of representation theory to construct strategies, providing yet another view on the relationship between the subgroup membership problem and the existence of perfect commuting operator strategies.

After this we reinterpret the PREF condition introduced in Chapter 3 as a "refutation (mod K)" then show, in an involved algebraic result, that all 3XOR games with perfect commuting operator strategies are noPREF games. This implies the existence of a polynomial time algorithm for deciding if a 3XOR game has a perfect commuting operator strategy, and implies that all 3XOR games with perfect commuting operator strategies have perfect MERP strategies.

Section 4.1 recaps basic definitions, gives precise statements of the main theorems, and proofs or proof sketches where appropriate. Section 4.2 gives proofs of the more involved algebraic results. The final sections fill in proof details and give perspectives, mostly about the subgroup K.

4.1 A Detailed Overview

We begin this section by recapping the notation necessary to state the main theorems of this work. Section 4.1.2 contains all the major theorem statements of this chapter.

4.1.1 Background and Notation

Games

As mentioned in Chapter 1, we think of XOR games as testing satisfiability of an associated system of equations. Our starting point for defining any kXOR game is a system of equations of the form

$$\hat{X}^{(1)}_{n_{11}} + \hat{X}^{(2)}_{n_{12}} + ... + \hat{X}^{(k)}_{n_{1k}} = s_1,\ \hat{X}^{(1)}_{n_{21}} + \hat{X}^{(2)}_{n_{22}} + ... + \hat{X}^{(k)}_{n_{2k}} = s_2,\ ...,\ \hat{X}^{(1)}_{n_{m1}} + \hat{X}^{(2)}_{n_{m2}} + ... + \hat{X}^{(k)}_{n_{mk}} = s_m$$

where $n_{i\alpha} \in [N]$, $s_i \in \{0,1\}$, $\hat{X}^{(\alpha)}_n$ are formal variables taking values in $\{0,1\}$ and the equations are all taken mod 2. N is called the *alphabet size* of the game, and m the number of *clauses*. The kXOR game associated to this system of equations has m question vectors. In a round of the game the verifier selects a $i \in [m]$ uniformly at random, then sends question vector $(n_{i1}, n_{i2}, ..., n_{ik})$ to the players, i.e. player j receives question n_{ij}. The players respond with single bit answers and win (get a score of 1 on) the round if the sum of their responses equals s_i mod 2. They get a score of 0 otherwise. Any kXOR game where clauses are chosen uniformly at random can be described by specifying the associated system of equations.[1]

For the case of 3XOR games, we will simplify notation slightly by omitting a subindex and instead writing our system of equations as

$$\hat{X}^{(1)}_{a_1} + \hat{X}^{(2)}_{b_1} + \hat{X}^{(3)}_{c_1} = s_1,\ \hat{X}^{(1)}_{a_2} + \hat{X}^{(2)}_{b_2} + \hat{X}^{(3)}_{c_2} = s_2,\ ...,\ \hat{X}^{(1)}_{a_m} + \hat{X}^{(2)}_{b_m} + \hat{X}^{(3)}_{c_m} = s_m$$

where $a_i, b_i, c_i \in [N]$ for all $i \in [m]$. The question vector sent to the players is then (a_j, b_j, c_j), with the players winning the round if their responses sum to s_j mod 2. We use the 3XOR notation for the remainder of this section.

[1] Because we are concerned with the case of perfect value XOR games, fixing the distribution clauses are drawn from to be uniform doesn't change the scope of our results.

Strategies

The most general classical strategy can be described by specifying a response for each player based on the question received and some shared randomness λ. If we are only concerned with strategies that maximize the players' score a minimax argument shows that we can ignore the shared randomness (fix λ to the value that maximizes the players' score in expectation), so optimal classical strategies can be described by fixing responses for each player to each possible question. To better align with the quantum case, we describe these strategies multiplicatively rather then additively. Define $X_i^{(\alpha)}$ to equal 1 if player α responds to question i with a 0, and $X_i^{(\alpha)} = -1$ if the player responds with a 1. Players win on the j-th question vector iff $X_{a_j}^{(1)} X_{b_j}^{(2)} X_{c_j}^{(3)} (-1)^{s_j} = 1$ so the expected score of the players conditioned on receiving the j-th question vector can be written

$$\frac{1}{2} + \frac{1}{2} X_{a_j}^{(1)} X_{b_j}^{(2)} X_{c_j}^{(3)} (-1)^{s_j}. \tag{4.1.1}$$

and the expected score this strategy achieves on a XOR game is given by

$$\frac{1}{2} + \frac{1}{2m} \sum_j X_{a_j}^{(1)} X_{b_j}^{(2)} X_{c_j}^{(3)} (-1)^{s_j}. \tag{4.1.2}$$

We refer to strategies where players share and measure a quantum state before deciding their response as *entangled strategies*.[2] In the most general entangled strategy, players share an state $|\psi\rangle$ and randomness λ. Then they receive a question, make a measurement on the quantum state based on the question and shared randomness, and then send a response to the verifier based on the measurement outcome. Mathematically, any strategy can be described by fixing the state $|\psi\rangle$ and POVMs (Positive Operator-Valued Measures) for each possible question sent to the players. As in the classical case, if we restrict to optimal strategies, we can ignore the shared classical randomness λ. Then we can describe an entangled strategy by specifying the shared state $|\psi\rangle$ and PVMs (Projective Valued Measures) for each possible player and question. We associate self-adjoint operators with these PVMs using the following

[2]The name quantum strategies, while more natural, can cause confusion with strategies where questions and responses are themselves quantum states. Entanglement is not necessary for these strategies, but the players' achieve a value exceeding their classical value only if the state they share is entangled.

prescription:

1. First, specify the shared state $|\psi\rangle$.

2. For each player $\alpha \in [k]$ and question $i \in [N]$, let $P_i^{(\alpha)}$ be the projector onto the subspace associated with a 1 response by player α to question i. Similarly, let $Q_i^{(\alpha)} = 1 - P_i^{(\alpha)}$ be the projector onto the subspace associated with a 0 response. Here 1 represents the identity operator.

3. For every α and i, define the *strategy observable* $X_i^{(\alpha)} = Q_i^{(\alpha)} - P_i^{(\alpha)}$.

The operators $X_i^{(\alpha)}$ satisfy some useful properties. Firstly, they are self-adjoint by construction with eigenvalues ± 1. From this, or from direct calculation, it follows that

$$\left(X_i^{(\alpha)}\right)^2 = \left(Q_i^{(\alpha)}\right)^2 + \left(P_i^{(\alpha)}\right)^2 + 2Q_i^{(\alpha)}P_i^{(\alpha)} = Q_i^{(\alpha)} + P_i^{(\alpha)} = 1 \tag{4.1.3}$$

where 1 represents the identity operator, and we have used the fact that $Q_i^{(\alpha)}$ and $P_i^{(\alpha)}$ are orthogonal projectors on the last line.

Secondly, the restriction that players be non-communicating means that a players chance of responding 1 (resp. 0) should be independent of another player's response. Hence

$$P_i^{(\alpha)}P_j^{(\beta)} = P_j^{(\beta)}P_i^{(\alpha)} \tag{4.1.4}$$

for any $i, j, \alpha \neq \beta$. Defining the group commutator of two observables $[y, z] := yzy^{-1}z^{-1}$ we see

$$\left[X_i^{(\alpha)}, X_j^{(\beta)}\right] = 1 \tag{4.1.5}$$

whenever $\alpha \neq \beta$.

Finally, we consider a product of operators corresponding to a question vector in the XOR game. A state is in the 1 eigenspace of $X_{a_j}^{(1)}X_{b_j}^{(2)}X_{c_j}^{(3)}$ iff the sum mod 2 of the players responses to the verifier upon measuring this state is 0. Similarly a state is in the -1 eigenspace iff the sum of the players responses upon measuring this state is 1. Then, players win on question

vector j with probability

$$\frac{1}{2} + \frac{1}{2} \langle\psi| X_{a_j}^{(1)} X_{b_j}^{(2)} X_{c_j}^{(3)} (-1)^{s_j} |\psi\rangle \tag{4.1.6}$$

and their overall score on the game is given by

$$\frac{1}{2} + \frac{1}{2m} \sum_j \left(\langle\psi| X_{a_j}^{(1)} X_{b_j}^{(2)} X_{c_j}^{(3)} (-1)^{s_j} |\psi\rangle \right). \tag{4.1.7}$$

An important consequence of Eq. (4.1.7) is that the players win the game with probability 1 iff

$$X_{a_j}^{(1)} X_{b_j}^{(2)} X_{c_j}^{(3)} (-1)^{s_j} |\psi\rangle = |\psi\rangle \tag{4.1.8}$$

for all $j \in [m]$. This is because each $X_i^{(\alpha)}$ has norm ≤ 1.

Bias.

XOR games can also be characterized by their *bias* $\beta(G)$, defined by $\beta(G) = 2\omega(G) - 1$.[3] The *entangled biases* β_{co}^* and β_{tp}^* are defined analogously. A completely random strategy for answering an XOR game will achieve a score of $1/2$, hence $\omega(G) \geq 1/2$ and $\beta(G) \in [0, 1]$, with identical bounds holding on the other biases. When comparing classical and entangled biases, the quantity usually considered is the ratio $\beta_{tp}^*(G)/\beta(G)$ (or $\beta_{co}^*(G)/\beta(G)$), called the quantum-classical gap.

For 2XOR games this gap can be related to the Grothendieck inequality, with

$$\beta_{co}^*(G)/\beta(G) = \beta_{tp}^*(G)/\beta(G) \leq K_G^{\mathbb{R}} \tag{4.1.9}$$

where $K_G^{\mathbb{R}}$ is the real Grothendieck constant[4]. For 3XOR games no such bound holds [50, 7],

[3] Some definitions vary by a factor of 2, defining $\beta(G) = \omega(G) - 1/2$

[4] Because $\omega_{co}^* = \omega_{tp}^*$ for 2XOR games, we also have $\beta_{co}^* = \beta_{tp}^*$

and there exist families of games $\{G_n\}_{n\in\mathbb{N}}$ with

$$\lim_{n\to\infty} \beta^*_{tp}(G_n)/\beta(G_n) = \infty. \tag{4.1.10}$$

All these families have the property that $\lim_{n\to\infty} \beta^*_{tp}(G_n) = 0$; it is open whether an unbounded quantum-classical gap can exist for kXOR games with β^*_{co} bounded away from zero. One special case where a bound on the quantum-classical gap is known is 3XOR games with the players restricted to a GHZ state [50] (later generatlized to Schmidt states in [6]). In this case the quantum-classical gap is bounded above by $4K_G^{\mathbb{R}}$ [6].

Groups

Now we introduce groups whose structure mimics the structure of the strategy observables introduced in Section 4.1.1. We describe these groups using the language of group presentations.

Given a kXOR game with alphabet size N, define the associated *game group* G to be the group with generators σ and $x_i^{(\alpha)}$ for all $i \in [n], \alpha \in [k]$, and relations:

1. $\left(x_i^{(\alpha)}\right)^2 = 1$ for all $i, j \in [n], \alpha \in [k]$
2. $\left[x_i^{(\alpha)}, x_j^{(\beta)}\right] = 1$ for all $i, j \in [n], \alpha \neq \beta \in [k]$
3. $\sigma^2 = \left[\sigma, x_i^{(\alpha)}\right] = 1$ for all $i, j \in [n], \alpha \neq \beta \in [k]$.

G is a right angled Coxeter group isomorphic to $\left({\mathbb{Z}_2}^{*N}\right)^{\times 3} \times \mathbb{Z}_2$. Here the $x_i^{(\alpha)}$ are group elements satisfying the same relations as the strategy observables defined in Section 4.1.1. σ is a formal variable playing the role of -1. Note $\sigma \neq 1$ in the group.

Given an 3XOR game testing the system of m equations

$$a_1 + b_1 + c_1 = s_1,\ a_2 + b_2 + c_2 = s_2,\ ...,\ a_m + b_m + c_m = s_m$$

we define the *clauses* $h_1, h_2, ..., h_m$ of the game by $h_i = x_{a_i}^{(1)} x_{b_i}^{(2)} x_{c_i}^{(3)} \sigma^{s_i} \in G$, where $\sigma^0 = 1$. We denote the set of all clauses by S and define the *clause group* $H \leq G$ to be the subgroup generated by the clauses, so $H = \langle S \rangle = \langle \{h_i : i \in [m]\} \rangle$. Important subgroups of groups G

and H are those consisting of even length words corresponding to each player. Define the *even subgroups* G^E, H^E by

$$G^E := \left\langle \left\{ x_i^{(\alpha)} x_j^{(\alpha)} : i,j \in [N], \alpha \in [k] \right\} \cup \{\sigma\} \right\rangle \text{ and } H^E := \langle \{h_i h_j : i,j \in [m]\} \rangle \quad (4.1.11)$$

Note that $H^E < G^E$.

Finally, define the *even commutator subgroup* K by

$$K = \left\langle \left\{ \left[x_i^\alpha x_j^\alpha, x_k^\alpha x_l^\alpha \right] : i,j,k,l \in [n], \alpha \in [3] \right\} \right\rangle^{G^E} \quad (4.1.12)$$

where X^{G^E} denotes the normal closure of the subgroup $X \leq G^E$ in G^E. Note that K is a normal subgroup of G^E by construction.

4.1.2 Precise Statements of Main Results

In this section we give theorem statements covering the main results of this chapter, along with some relevant theorems from previous work.

An algebraic characterization of perfect XOR Games

Our first result shows the problem of determining if $\omega_{co}^* = 1$ is equivalent to an instance of the subgroup membership problem on the game group G.

We should mention that some ingredients of this proof have appeared before in other contexts [47, 66]. The key innovation of this theorem is the algebraic formulation of the issue.

Theorem 4.1.1. *An XOR game has commuting operator value $\omega_{co}^* = 1$ iff $\sigma \notin H$, where σ, H are defined relative to the XOR game as described in Section 4.1.1.*

Proof. We first show that $\sigma \in H \Rightarrow w^* < 1$. Assume for contradiction that $\sigma \in H$ and $w^* = 1$. Then, since $\sigma \in H$, there exists a sequence of clauses $h_{t_1} h_{t_2} ... h_{t_l} = \sigma$ where each $h_{t_i} = x_{a_{t_i}}^{(1)} x_{b_{t_i}}^{(2)} x_{c_{t_i}}^{(3)} \sigma^{t_i} \in S$ is a generator of the clause group H. Because there exists a perfect value commuting operator strategy, that means there exits a state $|\psi\rangle$ and strategy observables

satisfying

$$x^{(1)}_{a_{t_i}} x^{(2)}_{b_{t_i}} x^{(3)}_{c_{t_i}} (-1)^{t_i} |\psi\rangle = |\psi\rangle \tag{4.1.13}$$

for all t_i (Eq. (4.1.8)). Because the strategy observables and -1 satisfy exactly the same relations as the associated group elements, we also have

$$\prod_i x^{(1)}_{a_{t_i}} x^{(2)}_{b_{t_i}} x^{(3)}_{c_{t_i}} (-1)^{t_i} = -1. \tag{4.1.14}$$

But then applying Eq. (4.1.13) and Eq. (4.1.14) gives

$$|\psi\rangle = \prod_i x^{(1)}_{a_{t_i}} x^{(2)}_{b_{t_i}} x^{(3)}_{c_{t_i}} (-1)^{t_i} |\psi\rangle = -|\psi\rangle , \tag{4.1.15}$$

a contradiction.

It remains to show $\sigma \notin H \Rightarrow w^* = 1$. A proof of this fact that relies on completeness of the nsSoS hierarchy is given in [66]. Here we give a standalone proof, which can be viewed as a special case of the GNS construction. We assume $\sigma \notin H$, and construct the strategy observables and state $|\psi\rangle$ explicitly.

First we define a Hilbert space $\mathcal{H}$ with orthogonal basis vectors corresponding to the left cosets of H in G. That is, $\mathcal{H}$ is spanned by basis vectors $\{|H\rangle , |g_1 H\rangle , ...\}$ with inner product

$$\langle g_1 H | g_2 H \rangle = \begin{cases} 1 & \text{if } g_1^{-1} g_2 \in H \\ 0 & \text{otherwise.} \end{cases} \tag{4.1.16}$$

Next we define the representation $\pi : G \to GL(\mathcal{H})$ to be the representation given by the left action of G on H, so

$$\pi(g_1) |g_2 H\rangle = |g_1 g_2 H\rangle . \tag{4.1.17}$$

Finally, define $|\psi\rangle = |H\rangle - |\sigma H\rangle$, and note that $\sigma \notin H$ by assumption implies $|\psi\rangle \neq 0$. We claim that strategy obserables $\pi(x_i^{(\alpha)})$ and state $|\psi\rangle$ achieve value $\omega^* = 1$ for the game. To

see this, first note that

$$\pi(\sigma)\left|\psi\right\rangle = \pi(\sigma)\left(\left|H\right\rangle - \left|\sigma H\right\rangle\right) = \left|\sigma H\right\rangle - \left|H\right\rangle = -\left|\psi\right\rangle \tag{4.1.18}$$

and for word $w \in H$ we have

$$\pi(w)\left|\psi\right\rangle = \pi(w)\left(\left|\sigma H\right\rangle - \left|H\right\rangle\right) = \left|\sigma w H\right\rangle - \left|wH\right\rangle = \left|\sigma H\right\rangle - \left|H\right\rangle = \left|\psi\right\rangle \tag{4.1.19}$$

since σ commutes with all elements of G. Then, for any $j \in [m]$ we have

$$\pi(x_{a_j}^{(1)})\pi(x_{b_j}^{(2)})\pi(x_{c_j}^{(3)})(-1)^{s_j}\left|\psi\right\rangle = \pi(x_{a_j}^{(1)}x_{b_j}^{(2)}x_{c_j}^{(3)}\sigma^{s_j})\left|\psi\right\rangle = \pi(h_j)\left|\psi\right\rangle = \left|\psi\right\rangle \tag{4.1.20}$$

and so the strategy achieves value $\omega^* = 1$ by Eq. (4.1.8). □

Theorem 4.1.1 implies that we could identify XOR games with value $\omega^* = 1$ by solving instances of the subgroup membership problem on the group G. Unfortunately, the subgroup membership problem on the group G is, in general, undecidable.[5] To get around this, we port the 3XOR problem to a simpler group obtained from G by modding out the normal subgroup K defined in Equation (4.1.12). On G/K the algebraic problem can be solved with a polynomial time algorithm.

Theorem 4.1.2. *Let σ, H^E, K be defined relative to an XOR game as described in Section 4.1.1. Let $[\sigma]_K$ be the coset containing σ after modding G^E out by K. Then we can check if $[\sigma]_K \notin H^E \pmod K$ in polynomial time.*

Proof. First note that $K \triangleleft G^E$ and $H^E < G^E$, so the question is well defined. To show a polynomial time algorithm, note that G^E/K is an abelian group – in fact we have modded out by exactly the commutator subgroup of G^E. The subgroup membership problem for any abelian group can be solved in polynomial time (see Theorem 4.4.1), so the result follows. □

[5] A game group G with $k \geq 2$ and $n \geq 3$ contains $\mathcal{F}_2 \times \mathcal{F}_2$ as a subgroup, where $\mathcal{F}_2$ is the free group on two elements. This group has undecidable subgroup membership problem by [45].

Sufficient conditions for $\omega^*_{co} = 1$

An obvious consequence of Theorem 4.1.2 comes from the observation

$$[\sigma]_K \notin H^E \pmod K \implies \sigma \notin H \implies \text{the associated XOR game has } \omega^*_{co} = 1. \quad (4.1.21)$$

Then, Theorem 4.1.2 tells us that a sufficient condition for an XOR game to have $\omega^*_{co} = 1$ can be checked in polynomial time. In fact we can say something stronger; when the condition given by Theorem 4.1.2 is met an optimal strategy can be chosen from a simple family of strategies which generalize the regular 3 qubit GHZ strategy. We introduce these strategies in Definition 4.1.3.

Definition 4.1.3. *[MERP strategies] A MERP (maximally entangled, relative phase) strategy for a kXOR game is one where the players share the k-qubit GHZ state* $|\psi\rangle = \frac{1}{\sqrt{2}}(|11...1\rangle + |00...0\rangle)$ *and, given question j, the* α*-th player measures the* α*-th qubit of the state with a strategy observable of the form*

$$M_j^{(\alpha)} := \exp\left(i\theta_j^{(\alpha)}\sigma_z\right)\sigma_x \exp\left(-i\theta_j^{(\alpha)}\sigma_z\right) \quad (4.1.22)$$

where σ_x, σ_z *are the Pauli X and Z matrices:* $\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ *and* $\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$. [6]

The angle $\theta_j^{(\alpha)}$ *depends on the player index* α *along with the question* j *sent to the player. To specify a MERP strategy we just need to specify the angles* $\theta_j^{(\alpha)}$ *for every* j *and* α*. For this reason we refer to the set of angles* $\{\theta_j^{(\alpha)} : \alpha \in [k], j \in [N]\}$ *as a description of the strategy.*

The MERP strategy observables for any choice of $\theta_j^{(\alpha)}$ are valid strategy observables, that is, they are hermitian with eigenvalues ± 1 and observables corresponding to different players commute.

We can now state the relationship between MERP strategies and the condition $\sigma \notin H \pmod K$.

[6] In the language of Section 4.1.1, the state $|\psi\rangle$ lives in the Hilbert space $\left(\mathbb{C}^2\right)^k$ and, given question j, player α measures a strategy observable of the form $I^{\otimes \alpha - 1} \otimes M_j^{(\alpha)} \otimes I^{\otimes k - \alpha}$ where I is the 2 by 2 identity matrix.

Theorem 4.1.4. *A kXOR game corresponds to a subgroup H with $[\sigma]_K \notin H^E \pmod K$ iff the game has $\omega^*_{co} = \omega^*_{tp} = 1$ with a perfect value MERP strategy. A description of this strategy can be found in polynomial time.*

Proof. This theorem is a rephrasing of Theorem 5.30 from [66], where the condition $[\sigma]_K \notin H \pmod K$ was referred to as existence of a PREF (parity refutation). The equivalence between the $\sigma \in H \pmod K$ condition and existence of a parity refutation is elaborated on in Section 4.3.3.

In Section 4.3.4 we prove the theorem in one direction by showing that MERP matrices satisfy the defining relations for K. The other direction is proved by defining a system of linear diophantine equations which are solved only when $[\sigma]_K \in H \pmod K$ then showing, via a theorem of alternatives, that these equations being unsatisfied implies a MERP strategy can achieve value 1.

□

The sufficient conditions are necessary

Theorems 4.1.1, 4.1.2 and 4.1.4 give a sufficent condition for XOR games to have value $\omega^*_{co} = 1$, and a complete characterization of strategies for games that meet this condition. Theorem 4.1.6, the main technical result of this chapter, gives the surprising result that this sufficient condition is also necessary for 3XOR games. The proof is purely algebraic, but involved. We give the full proof in Section 4.2, and sketch high level intuition for the result here.

As a warm up, we show that $\sigma \in H$ iff $\sigma \in H^E$.

Lemma 4.1.5. *For any XOR game, $\sigma \in H$ iff $\sigma \in H^E$*

Proof. The direction $\sigma \in H^E \Rightarrow \sigma \in H$ is immediate.

To see the other, note that each clause h_i contains exactly one generator $x_i^{(\alpha)}$ for each $\alpha \in [k]$. Then an odd length sequence of clauses contains an odd number of generators $x_i^{(\alpha)}$ for each $\alpha \in [k]$. Because all the relations of G involve words containing an even number of generators corresponding to each player α, the parity of the number of generators corresponding to each player remains fixed when applying the relations of G. Thus, any word

in G equals to the product of an odd number of clauses from H and contains an odd number (therefore at least one) generator corresponding to each player α and cannot equal σ.

From this, we conclude that if $\sigma \in H$ it is an even sequence of clauses $h_1 h_2 ... h_{2l} \in H^E$ which equals σ, thus $\sigma \in H^E$ as well. □

With Lemma 4.1.5 in hand, we turn our attention to Theorem 4.1.6.

Theorem 4.1.6. *σ is contained in H iff, after modding out by K, the coset containing σ is contained in H^E. That is:*

$$\sigma \in H \Leftrightarrow [\sigma]_K \in H^E \pmod{K}. \tag{4.1.23}$$

Some ideas of the proof. One direction is immediate: if $\sigma \in H$, then it is in H^E by Lemma 4.1.5, and $[\sigma]_K$ remains in H^E after modding out by K.

To begin the proof in the other direction, note $[\sigma]_K \in H^E \pmod{K}$ iff there exists a word $h \in H^E$ with $h = w_k \sigma$ and $w_k \in K$. To prove the result it suffices to show $w_k \in H^E$, since then

$$w_k^{-1} h = w_k^{-1} w_k \sigma = \sigma \in H^E. \tag{4.1.24}$$

In Section 4.2.4 we show this is true. Here we give some simple intuition about this result. At this point the proof and proof sketch diverge – our goal here is to give intuition, rather than a high level overview of the proof.

Consider a pair of clauses $h_1, h_2 \in S$ corresponding to question vectors which send the same question to the first player, so $h_1 = x_{a_1}^{(1)} x_{b_1}^{(2)} x_{c_1}^{(3)} \sigma^{s_1}$, $h_2 = x_{a_2}^{(1)} x_{b_2}^{(2)} x_{c_2}^{(3)} \sigma^{s_2}$ and $a_1 = a_2$. Similarly, let clauses h_3, h_4 be clauses which agree on the question sent to the second player

so $x_{b_3} = x_{b_4}$.[7] We then consider the commutator

$$[h_1h_2, h_3h_4] = \left[x_{a_1}^{(1)}x_{a_2}^{(1)}, x_{a_3}^{(1)}x_{a_4}^{(1)}\right]\left[x_{b_1}^{(2)}x_{b_2}^{(2)}, x_{b_3}^{(2)}x_{b_4}^{(2)}\right]\left[x_{c_1}^{(3)}x_{c_2}^{(3)}, x_{c_3}^{(3)}x_{c_4}^{(3)}\right]\left[\sigma^{s_1+s_2}, \sigma^{s_3+s_4}\right] \tag{4.1.25}$$

$$= \left[1, x_{a_3}^{(1)}x_{a_4}^{(1)}\right]\left[x_{b_1}^{(2)}x_{b_2}^{(2)}, 1\right]\left[x_{c_1}^{(3)}x_{c_2}^{(3)}, x_{c_3}^{(3)}x_{c_4}^{(3)}\right]\left[\sigma^{s_1+s_2}, \sigma^{s_3+s_4}\right] \tag{4.1.26}$$

$$= \left[x_{c_1}^{(3)}x_{c_2}^{(3)}, x_{c_3}^{(3)}x_{c_4}^{(3)}\right] \tag{4.1.27}$$

where we have used the fact that group elements corresponding to different players commute on the first line, that $x_{a_1}^{(1)}x_{a_2}^{(1)} = \left(x_{a_1}^{(1)}\right)^2 = 1$ on the second line, and that $[w, 1] = 1$ for any w and σ commutes with anything on the third.

The conclusion is that $\left[x_{c_1}^{(3)}x_{c_2}^{(3)}, x_{c_3}^{(3)}x_{c_4}^{(3)}\right] = [h_1h_2, h_3h_4] \in H^E$. Let χ denote the set of all commutators of pairs of generators $x_i^{(\alpha)}$ which lie in H^E, and note we just proved χ is necessarily nonempty. These commutators of pairs generate K, thus $K \cap H^E$ is nonempty as well. We can repeat the same argument as above with any two pairs of clauses that cancel on two different players, so for most XOR games χ will be reasonably large, and $K \cap H^E$ will be large as well. What we show in Section 4.2.4 is that $K \cap H^E$ is large enough that the w_k of Equation (4.1.24) is in $K \cap H^E$. The proof is done with involved bookkeeping organized with graphs which track the distribution of clauses over possible questions in the game. □

With the Theorems 4.1.1, 4.1.2, 4.1.4 and 4.1.6, we conclude the following result.

Theorem 4.1.7. *A 3XOR game has value $\omega_{co}^* = 1$ iff it has a perfect value MERP strategy, implying $\omega_{co}^* = \omega_{tp}^* = 1$. Additionally, there exists a polynomial time algorithm which decides if a 3XOR game has value $\omega_{co}^* = 1$, and outputs a description of the perfect value MERP strategy if one exists.*

Proof. By Theorem 4.1.1, an 3XOR game has $\omega_{co}^* = 1$ iff $\sigma \notin H$ in the associated group. By Theorem 4.1.6, this is also equivalent to the statement $[\sigma]_K \in H \pmod K$. By Theorem 4.1.4 this implies a MERP strategy, and the first part of the result follows.

[7] These pairs of clauses don't need to exist, but XOR games where each question is asked only once are particularly simple, with $\omega = 1$, so we assume we are not in this case.

To get the polynomial time algorithm, we just need to check if $[\sigma]_K \in H \pmod K$, which we can do in polynomial time by Theorem 4.1.2. If true, there exists a MERP strategy and we can find it by Theorem 4.1.4. If false, the same chain of implications as above shows $\omega_{co}^* < 1$. □

Bounds on the bias ratio

Finally, combining Theorem 4.1.7 with a result from [50] gives the following.

Theorem 4.1.8. *A 3XOR game with $\omega_{co}^* = 1$ also has classical value $\omega > 1/2 + \frac{1}{8K_G^{\mathbb{R}}} \geq 0.57$, where $K_G^{\mathbb{R}}$ is the real Grothendieck constant.*

Proof. By Theorem 4.1.7, a 3XOR game G with $\omega_{co}^* = 1$ must also have a perfect value MERP strategy. This strategy uses a GHZ state for the players, and a bound from [50] gives that

$$\beta_{GHZ}^*/\beta \leq 4K_G^{\mathbb{R}}, \tag{4.1.28}$$

where β_{GHZ}^* is the maximum bias achieved with a strategy using a GHZ state. But then

$$\beta(G) \geq \frac{\beta_{GHZ}^*(G)}{4K_G^{\mathbb{R}}} = \frac{1}{4K_G^{\mathbb{R}}} \tag{4.1.29}$$

$$\implies \omega(G) \geq \frac{1}{2} + \frac{1}{8K_G^{\mathbb{R}}} \tag{4.1.30}$$

and the result follows. □

4.2 Technical Details

This section begins with definitions, then compares the algebraic structure defined in this chapter to the one introduced in [14], then proves theorem 4.1.6.

4.2.1 Definitions

We briefly recap the definitions given in Section 4.1.1, then give some additional notation that will be useful in this section. In everything that follows [,] denotes the group commutator, so $[x, y] = xyx^{-1}y^{-1}$.

Recap

We consider a 3XOR game with questions drawn from an alphabet of size $[N]$. The game has m question vectors labeled $(a_1, b_1, c_1), ...(a_m, b_m, c_m)$ with $a_i, b_i, c_i \in [N]$. When asked the i-th question vector (a_i, b_i, c_i) players win the game if their responses sum (mod 2) to the parity bit $s_i \in \{0, 1\}$. Parity bits are defined for all $i \in [m]$.

There are several algebraic objects associated with the game. The first is the game group G, defined by

$$G := \left\langle x_i^{(\alpha)} : i \in [N], \alpha \in [3] \middle| \left[x_i^{(\alpha)}, x_j^{(\beta)}\right], \left(x_i^{(\alpha)}\right)^2 \forall\, i, j, \alpha \neq \beta \right\rangle \times \langle \sigma | \, \sigma^2 \rangle \,. \tag{4.2.1}$$

Group elements $x_i^{(\alpha)}$ correspond to the observable measured by player α upon receiving question i. The group element σ should be though of as a formal variable corresponding to -1 in the group. Note σ has order two ($\sigma^2 = 1$) and commutes with all elements of group ($[\sigma, w] = 1$ for any $w \in G$).

For all $i \in [m]$ we define the associated clause

$$h_i = x_{a_i}^{(1)} x_{b_i}^{(2)} x_{c_i}^{(3)} \sigma^{s_i}. \tag{4.2.2}$$

The clause set $S = \{h_i\}_{i \in [m]}$ contains all clauses of the game. The clause group $H = \langle S \rangle$ is the subgroup of G generated by the clauses.

The even game group G^E is the subgroup of G consisting of words with an even number of observables corresponding to each player (plus an optional σ), so

$$G^E = \left\langle \left\{ x_i^{(\alpha)} x_j^{(\alpha)} : i, j \in [N], \alpha \in [k] \right\} \cup \{\sigma\} \right\rangle . \tag{4.2.3}$$

The even clause group is the subgroup of G generated by an even number of clauses

$$H^E = \langle \{ h_i h_j : i, j \in [m] \} \rangle . \tag{4.2.4}$$

An important observation is that H^E is a subgroup of G^E.

Finally, K is the commutator subgroup of G^E, defined to be the normal closure of the set of commutators of the generators of G^E. In math:

$$K = \left\langle \left\{ \left[x_i^\alpha x_j^\alpha, x_k^\alpha x_l^\alpha \right] : i, j, k, l \in [n], \alpha \in [3] \right\} \right\rangle^{G^E} \tag{4.2.5}$$

Where $\langle X \rangle^Y$ denotes the normal closure of the set X in the group Y.

Projections and Clause Graphs

It will be helpful to have notation for referring to just the observables associated with a single player. To this end, define *player subgroups* $G_\alpha \leq G$ by

$$G_\alpha = \left\langle \left\{ x_i^{(\alpha)} : i \in N \right\} \right\rangle \tag{4.2.6}$$

and $G_\alpha^E \leq G^E$ by

$$G_\alpha^E = \left\langle \left\{ x_i^{(\alpha)} x_j^{(\alpha)} : i, j \in N \right\} \right\rangle \tag{4.2.7}$$

for all $\alpha \in \{1, 2, 3\}$. Because observables corresponding to different players commute, we can write any $w \in G$ as

$$w = w_1 w_2 w_3 \sigma^{s_w} \tag{4.2.8}$$

where $w_\alpha \in G_\alpha$ for all $\alpha \in \{1,2,3\}$, and $s_w \in \{0,1\}$. Similarly, any $w' \in G^E$ can be written as

$$w = w'_1 w'_2 w'_3 \sigma^{s'_w} \tag{4.2.9}$$

with $w'_\alpha \in G^E_\alpha$ and $s'_w \in \{0,1\}$.

For any $\alpha \in \{1,2,3\}$ we also define the projector onto player subgroups $\varphi_\alpha : G \to G_\alpha$ by defining its action on the generators of G:

$$\varphi_\alpha(x_i^{(\beta)}) = \begin{cases} x_i^{(\beta)} & \text{if } \alpha = \beta \\ 1 & \text{otherwise} \end{cases} \qquad \text{and} \qquad \varphi_\alpha(\sigma) = 1 \tag{4.2.10}$$

then extending φ_α to a homorphism on G. To see this defines a valid homomorphism note it preserves the group relations:

$$\varphi_\alpha \left(x_i^{(\beta)}\right)^2 = \begin{cases} \left(x_i^{(\beta)}\right)^2 = 1 & \text{if } \alpha = \beta \\ 1^2 = 1 & \text{otherwise} \end{cases} \tag{4.2.11}$$

with a similarly simple argument showing commutation relations are preserved. It is also helpful to define a projection φ_σ which acts on the generators of G as

$$\varphi_\sigma \left(x_i^{(\beta)}\right) = 1 \qquad \text{and} \qquad \varphi_\sigma(\sigma) = \sigma. \tag{4.2.12}$$

Combining Equation (4.2.8) with the definition of φ_α gives the equation

$$w = \varphi_1(w)\varphi_2(w)\varphi_3(w)\varphi_\sigma(w) \tag{4.2.13}$$

for any $w \in G$.

Next, we define the clause (hyper)graph[8] $\mathcal{G}_{123}$ which gives a useful way of visualizing the clause structure of a game. The graph has $3n$ vertices which we identify with the generators x_i^α of the group G. We label the vertices by the corresponding generator. (Hyper)edges in

[8] A hypergraph is a graph with edges passing through more than two vertices.

the hypergraph correspond to clauses, with a hyperedge going through vertices $x_{a_i}^{(1)}$, $x_{b_i}^{(2)}$, and $x_{c_i}^{(3)}$ for every clause $x_{a_i}^{(1)} x_{b_i}^{(2)} x_{c_i}^{(3)} \sigma^{s_i} \in S$. Note that the existence of the edge is independent of the value of s_i, so the clause graph contains no information about the parity bits.

Because edges in the hypergraph correspond to clauses $h \in S$, we can identify a path in $\mathcal{G}_{123}$ with a word $w \in H$. We will use this relationship frequently in the future.

We also define important subgraphs of $\mathcal{G}_{123}$ by taking the induced graphs on vertices corresponding to a subset of players.[9] For any $\alpha \neq \beta \in \{1, 2, 3\}$ we define the multigraph $G_{\alpha\beta}$ to be subgraph of $\mathcal{G}_{123}$ induced by the vertices corresponding to generators of G_α and G_β. See Figure 4-2 for an example. As with the graph $\mathcal{G}_{123}$, edges in the graph $\mathcal{G}_{\alpha\beta}$ can be identified with clauses in H and paths in $\mathcal{G}_{\alpha\beta}$ can be identified with words $w \in H$.

In Section 4.2.3 we show that we can restrict our attention to the case where $\mathcal{G}_{123}$ is a connected graph. The induced graph $\mathcal{G}_{\alpha\beta}$ can be disconnected, and the different connected components of this graph (and representative elements from each) play an important role in the proof in Section 4.2.4.

4.2.2 Comparison with Linear Systems Games

A reader familiar with the work of Cleve, Liu and Slofstra concerning linear systems games [14] may notice a similarity between the solution group defined in that paper and the clause group defined in this work. In this section we give a direct comparison between the two. Our goal in doing this is not to provide any deep insights – we simply hope a direct comparison will help a reader already familiar with linear systems games to better understand our work. We do not define linear systems games here, and point readers to [14] for a formal introduction to them. This section is not critical and a reader can safely skip it without impacting their understanding of the rest of this chapter.

Following [14], we consider a binary linear system of m equations on n variables $Mx = b$, with $M \in \mathbb{Z}_2^{m \times n}$ and $b \in \mathbb{Z}^m$. M_{ij} specifies an individual entry in the matrix M, and b_i specifies an entry from the vector b. The solution group of the binary linear system is a group with generators $g_1, g_2 ..., g_n, J$ and relations

[9] For a graph $\mathcal{X} = (V, E)$, the subhypergraph induced by a set of vertices $V' \subseteq V$ is the hypergraph with vertex set V' and edge set $E' = \{e \cap V' : e \in E\}$. Essentially, edges are all truncated to the vertices in V'.

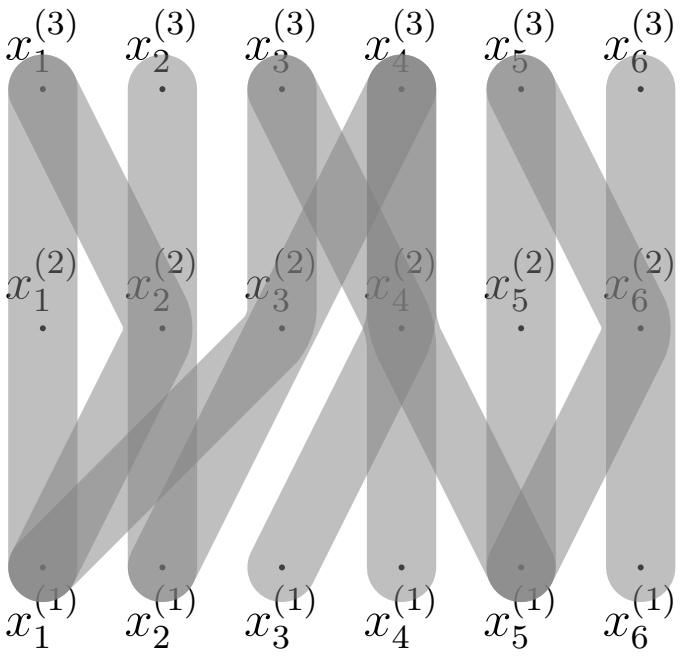

Figure 4-1: Sample hypergraph $\mathcal{G}_{123}$ for a game with alphabet size $N = 6$ and 11 clauses. The hypergraph is generated by clause set (σ terms omitted since they don't affect the graph):

$$S = \{x_1^{(1)}x_1^{(2)}x_1^{(3)}, x_1^{(1)}x_2^{(2)}x_1^{(3)}, x_2^{(1)}x_2^{(2)}x_2^{(3)}, x_1^{(1)}x_3^{(2)}x_3^{(3)}, x_2^{(1)}x_3^{(2)}x_4^{(3)}, x_3^{(1)}x_4^{(2)}x_4^{(3)},$$
$$x_4^{(1)}x_4^{(2)}x_3^{(3)}, x_5^{(1)}x_4^{(2)}x_4^{(3)}, x_5^{(1)}x_6^{(2)}x_5^{(3)}, x_5^{(1)}x_5^{(2)}x_5^{(3)}, x_6^{(1)}x_6^{(2)}x_6^{(3)}\}$$

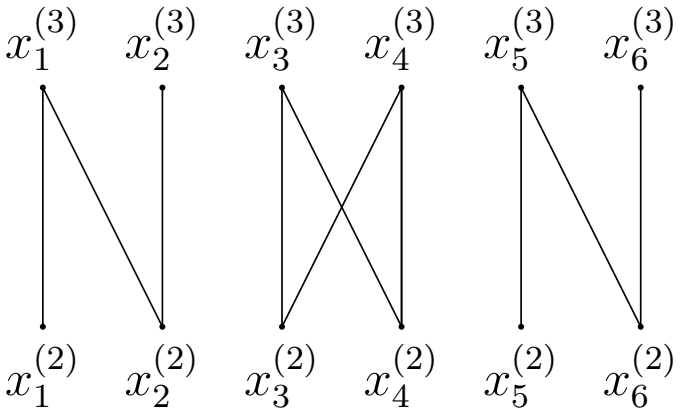

Figure 4-2: Induced graph $\mathcal{G}_{23}$ corresponding to the same clause set as Figure 4-1

1. $g_i^2 = 1$ for all $i \in [n]$ and $J^2 = 1$

2. $[g_i, J] = 1$ for all $i \in [n]$

3. $[g_i, g_j] = 1$ if x_i and x_j appear in the same equation (that is $M_{li} = M_{lj} = 1$ for some $l \in [m]$).

4. $\prod_i \left(g_i^{M_{li}}\right) J^{b_l} = 1$ for all $l \in [m]$.

In [14] the authors showed the following result:

Theorem 4.2.1 (Implied by Theorem 4 of [14], paraphrased)**.** *The linear system game associated to the system of equations $Mx = b$ has a perfect value commuting operator strategy iff in the associated solution group we have $J \neq 1$.*

Theorem 4.1.1 can be thought of as an analog of Theorem 4.2.1 for 3XOR games. We can restate Theorem 4.2.1 in a way that makes the comparison even more apparent.

Given a system of equations $Mx = b$, define the group G_{lsg} to be the group with generators $g_1, g_2, ..., g_n, J$ and relations 1-3 above. Note that $J \neq 1$ in this group. Next, define the subgroup $H_{lsg} \triangleleft G_{lsg}$ to be the normal closure in G_{lsg} of the words corresponding to equations in the system of equations $Mx = b$ (that is, the words involved in relation 4 above) so

$$H_{lsg} = \left\langle \left\{ \prod_i \left(g_i^{M_{li}}\right) J^{b_l} : l \in [m] \right\} \right\rangle^{G_{lsg}}. \tag{4.2.14}$$

Using these definitions, an equivalent statement of Theorem 4.2.1 is:

Theorem 4.2.2 (Restatement of Theorem 4.2.1)**.** *The linear system game associated to the system of equations $Mx = b$ has a perfect value commuting operator strategy iff $J \notin H_{lsg}$.*

We can compare the above theorem and Theorem 4.1.1 directly. We list, and briefly discuss, the key differences:

i) The group G contains an element for every question player combination, while G_{lsg} only contains an element for every question. In a commuting operator (or tensor product) strategy for an XOR game, different players can measure completely different

observables when sent the same question and so we need a different group element to correspond to each player-question combo.[10] Conversely, in linear systems games there is a close relationship between Alice and Bob's measurements given the same question, and both players measurement operators can be constructed from representations (right and left actions) of the same group elements.

i) Generators of G_{lsg} commute with each other if they appear in the same equation (relation 3 above). Generators of G satisfy no such relation. This difference reflects a difference between linear systmes games and XOR games strategies. In a linear systems game a single player must make simultaneous measurements of all the operators corresponding to a question in the game. This never happens in XOR games. From an algebraic point of view, these extra relations place a restriction on elements of G_{lsg} that is not placed on elements of G.

i) The group H_{lsg} is a normal subgroup of G_{lsg}, while H is not a normal subgroup of G. This has an algebraic consequence: asking if $J \in H_{lsg}$ is an instance of the word problem (mod out by the generators of H_{lsg}, then ask if J equals the identity), while asking if $\sigma \in H$ is an instance of the subgroup membership problem. The word problem is in a sense "easier" than the subgroup membership problem: there are groups with solvable word problem but undecidable subgroup membership problem [45]. Still, both problems are undecidable in general. This difference also has consequences for game strategies. In a linear systems game, an identity of the form

$$\prod_i \left(g_i^{M_{li}}\right) J^{b_l} = 1 \tag{4.2.15}$$

holds in the group, hence holds as an operator identity on the strategy observables as well. In an XOR game, the operator identities codified in H only need hold acting on the state $|\psi\rangle$ and there are games (for example, the GHZ game) where products of strategy observables act as the identity on $|\psi\rangle$, but the operators themselves do not multiply to the identity.

[10] Put (informally) in slightly different terms: XOR games can be very far from synchronous, as defined in [34].

We should also point out that a linear systems game can be defined for any system of equations of the form $Mx = b$, while XOR games require equations of a special form: exactly one variable corresponding to each player is involved in each equation. It is possible to define a slightly more general form of kXOR games with a subset of players, as opposed to all players, queried on each question but those are not considered here.

Theorem 4.1.7, in combination with [61] shows that there cannot exist a mapping from linear systems games to XOR games which preserves the commuting operator value of the game. The question of finding a natural map in the other direction remains open.

4.2.3 Connectivity of the Clause Graph

In Section 4.2.1 we introduced the *clause graph* $\mathcal{G}_{123}$ – a graphical representation of the clause structure of a 3XOR game. In this section we consider 3XOR games whose associated clause graph is not connected. Given such a game, we can always define smaller games involving only the clauses corresponding to a single connected component of the clause graph. Here, we show a 3XOR game has $\omega^*_{co} = 1$ iff each of these smaller games has a perfect commuting operator strategy.

This result can be understood from a strategies point of view. Recall that a clause $x^{(1)}_{a_i} x^{(2)}_{b_i} x^{(3)}_{c_i}$ corresponds to a question vector (a_i, b_i, c_i) that could be sent to the players in a round of the game. If a game has a disconnected clause graph $\mathcal{G}_{123}$, players will never be sent a question vector asking them to make measurements from different connected components of the graph. Thus, players can consider the measurements in each connected component of $\mathcal{G}_{123}$ independently when coming up with a strategy for the game. If they come up with strategies that win for each connected component of clauses they can always combine them (given a question, a player follows the strategy corresponding to the connected component that question came from) to create a strategy that wins on the larger game.

Below, we prove the result using algebraic techniques. The proof is slightly less natural in this setting, but provides a useful exercise in proving results about XOR games using the groups formalism.

Theorem 4.2.3. *Let G be a 3XOR game with clause set S, clause group H, and clause*

graph $\mathcal{G}_{123}$. Then $\sigma \in H$ iff there exists a subset of clauses $S' \subseteq S$ corresponding to all the edges in a connected component of $\mathcal{G}_{123}$ with $\sigma \in \langle S' \rangle$.

Proof. First note that if the clause graph $\mathcal{G}_{123}$ is connected Theorem 4.2.3 is trivial, since the the only subset of S corresponding to a connected component of $\mathcal{G}_{123}$ is S itself. Also note that one direction of the above claim is immediate by the observation that $\langle S' \rangle < \langle S \rangle$ and so $\sigma \in \langle S' \rangle \implies \sigma \in \langle S \rangle = H$.

To deal with the converse direction, consider a game G with clause group $H \ni \sigma$ and a disconnected clause graph $\mathcal{G}_{123}$. Let $S_1, S_2, ..., S_l$ be subsets of S corresponding to all the edges in the connected components of the clause graph. Not that sets $S_1, ...S_l$ partition the S. For all $i \in [l]$, define a map ρ_i which acts on the generators of H as

$$\rho_i(h_j) = \begin{cases} h_j & \text{if } h_j \in S_i \\ 1 & \text{otherwise} \end{cases} \tag{4.2.16}$$

We can extend ρ_i to act on a sequence of clauses in the natural way, so[11]

$$\rho_i(h_{r_1} h_{r_2} ... h_{r_t}) = \rho_i(h_{r_1}) \rho_i(h_{r_2}) ... \rho_i(h_{r_t}). \tag{4.2.17}$$

We have by assumption that $\sigma \in H$. Then there exists a sequence of clauses $h_{r_1} h_{r_2} ... h_{r_t} = \sigma$. We prove two claims:

1. For all $\alpha \in \{1, 2, 3\}$, $i \in [l]$ we have : $\varphi_\alpha(\rho_i(h_{r_1} h_{r_2} ... h_{r_t})) = 1$.
2. For some $i' \in [l]$, we have $\rho_{i'}(h_{r_1} h_{r_2} ... h_{r_t}) = \sigma$.

To prove the first, define the set V_i to consist of all generators $x_j^{(\alpha)}$ corresponding to vertices in the connected component of $\mathcal{G}_{123}$ containing clauses S_i. Then, for all $\alpha \in \{1, 2, 3\}$, define $V_i^{(\alpha)} = V_i \cap G_\alpha$ to be the subset of generators in V_i corresponding to player α. Finally, we

[11] Note that we do not claim ρ_i extended in this way is a homomorphism. It is not, because its action on σ may be undefined.

define the a homomorphism $\pi_i : G \to G$ by its action on the generators of G:

$$\pi_i(x_j^{(\alpha)}) = \begin{cases} x_j^{(\alpha)} & \text{if } x_j^{(\alpha)} \in V_i \\ 1 & \text{otherwise} \end{cases} \qquad \text{and} \qquad \pi_i(\sigma) = 1. \tag{4.2.18}$$

Routine calculation shows that π_i preserves the relations of G, and thus, is a valid homomorphism. Now, to prove claim 1 we show

$$\varphi_\alpha(\rho_i(h_{r_1}h_{r_2}...h_{r_t})) = \varphi_\alpha(\pi_i(h_{r_1}h_{r_2}...h_{r_t})) = \varphi_\alpha(\pi_i(\sigma)) = 1. \tag{4.2.19}$$

The second equality follows because we assumed $h_{r_1}h_{r_2}...h_{r_t} = \sigma$, and the third equality holds by definition of φ_α. All that remains to show is the first. We show this through direct computation. For ease of notation fix $\alpha = 1$. Then, for any clause h_{r_j} we have

$$\varphi_1(\rho_i(h_{r_j})) = \varphi_1(\pi_i(h_{r_j})) = x_{a_{r_j}}^{(1)} \tag{4.2.20}$$

if $h_{r_j} \in S_i$ and

$$\varphi_1(\rho_i(h_{r_j})) = \varphi_1(\pi_i(h_{r_j})) = 1 \tag{4.2.21}$$

otherwise, since $h_{r_j} \notin S_i \implies \varphi_1(h_{r_j}) \notin V_i$ by definition of V_i. Applying this observation to each term in the sequence $h_{r_1}...h_{r_l}$ shows the first equality.

Now, to prove the second claim, note Claim 1 in combination with Equation (4.2.13) gives

$$\rho_i(h_{r_1}h_{r_2}...h_{r_t}) = \varphi_1(\rho_i(h_{r_1}h_{r_2}...h_{r_t}))\varphi_2(\rho_i(h_{r_1}h_{r_2}...h_{r_t}))\varphi_3(\rho_i(h_{r_1}h_{r_2}...h_{r_t}))\varphi_\sigma(\rho_i(h_{r_1}h_{r_2}...h_{r_t})) \tag{4.2.22}$$

$$= \varphi_\sigma(\rho_i(h_{r_1}h_{r_2}...h_{r_t})). \tag{4.2.23}$$

If $\varphi_\sigma(\rho_i(h_{r_1}h_{r_2}...h_{r_t})) = \sigma$ for any $i \in [l]$ the above equation proves Claim 2. Assume for

contradiction that $\varphi_\sigma(\rho_i(h_{r_1}h_{r_2}...h_{r_t})) = 1$ for all $i \in [l]$. Then we have

$$\varphi_\sigma(h_{r_1}h_{r_2}...h_{r_t}) = \varphi_\sigma(h_{r_1})\varphi_\sigma(h_{r_2})...\varphi_\sigma(h_{r_t}) \tag{4.2.24}$$

$$= \varphi_\sigma\left(\prod_i \rho_i(h_{r_1})\right)\varphi_\sigma\left(\prod_i \rho_i(h_{r_2})\right)...\varphi_\sigma\left(\prod_i (h_{r_t})\right) \tag{4.2.25}$$

$$= \prod_i \varphi_\sigma\left(\rho_i\left(h_{r_1}h_{r_2}...h_{r_t}\right)\right) = 1 \tag{4.2.26}$$

Where we used the fact that σ commutes with all elements of G to reorder elements and get from the second line to the third, and our assumption for the sake of contradiction on the final line. But, by our assumption at the start of this section we also have $\varphi_\sigma(h_{r_1}h_{r_2}...h_{r_t}) = \sigma$. The contradiction proves Claim 2.

Finally, to complete the proof we note

$$\rho_{i'}\left(h_{r_1}...h_{r_t}\right) = \varphi_1\left(\rho_{i'}\left(h_{r_1}...h_{r_t}\right)\right)\varphi_2\left(\rho_{i'}\left(h_{r_1}...h_{r_t}\right)\right)\varphi_3\left(\rho_{i'}\left(h_{r_1}...h_{r_t}\right)\right)\varphi_\sigma\left(\rho_{i'}\left(h_{r_1}...h_{r_t}\right)\right) \tag{4.2.27}$$

$$= \sigma \tag{4.2.28}$$

by Equation (4.2.13), Claim 1, and Claim 2, and $\rho_{i'}\left(h_{r_1}h_{r_2}...h_{r_t}\right) \in S_{i'}$ by definition of $\rho_{i'}$. Thus the claim holds with $S' = S_{i'}$. □

To prove the strongest form of Theorem 4.1.6, we also need a version of Theorem 4.2.3 that applies to works $[\sigma]_k \in H^E \pmod K$. We give that theorem next. The proof is very similar to the proof of Theorem 4.2.3, with a few more technical details.[12]

Theorem 4.2.4. *Let G be a 3XOR game with clause set S, clause group H, and clause graph $\mathcal{G}_{123}$. Then $[\sigma]_k \in H^E \pmod K$ iff there exists a subset of clauses $S' \subseteq S$ corresponding to all the edges in a connected component of $\mathcal{G}_{123}$ for which $[\sigma]_k \in \langle S'\rangle \cap H^E \pmod K$.*

Proof. As with the proof of Theorem 4.2.3, the case where $\mathcal{G}_{123}$ is connected and the direction $[\sigma]_k \in \langle S'\rangle \cap H^E \pmod K \Rightarrow [\sigma]_k \in H^E \pmod K$ are immediate.

[12] Actually, theorem 4.2.4 in combination with Theorem 4.1.6 provide an alternate proof of Theorem 4.2.3. Here we proved Theorem 4.2.3 directly both because the proof serves as a good warm up to the proof of Theorem 4.2.4, and to enphasize the result can be proved independtly from Theorem 4.1.6.

To deal with the remaining case, let G be an XOR game with disconnected clause graph $\mathcal{G}_{123}$ and $\sigma \in H^E \pmod K$. Let $S_1, S_2, ..., S_l$ be subsets of S corresponding to all edges in the connected components of the clause graph. For each S_i, we pick some representative clause $\hat{h}_i \in S_i$. Then, define a map $\tilde{\rho}_i$ which acts on the generators of H as

$$\tilde{\rho}_i(h_j) = \begin{cases} h_j & \text{if } h_j \in S_i \\ \hat{h}_i & \text{otherwise.} \end{cases} \tag{4.2.29}$$

Extend $\tilde{\rho}_i$ to act on sequences of clauses in the natural way, so

$$\tilde{\rho}_i(h_1 h_2 ... h_l) = \tilde{\rho}_i(h_1)\tilde{\rho}_i(h_2)...\tilde{\rho}_i(h_l). \tag{4.2.30}$$

Note that for any generator of $h_j h_{j'}$ of the even clause group H^E we have

$$\tilde{\rho}_i(h_j h_{j'}) = \tilde{\rho}_i(h_j)\tilde{\rho}_i(h_{j'}) \in H^E \cap \langle S_i \rangle. \tag{4.2.31}$$

As in the proof of Theorem 4.2.3, define the subset of generators V_i to be the $x_i^{(\alpha)}$ corresponding to vertices in the same connected component as the edges in S_i. Then define the projector $\tilde{\pi}_i$ which acts on the generators of G as

$$\tilde{\pi}_i(x_i^{(\alpha)}) = \begin{cases} x_i^{(\alpha)} & \text{if } x_i^{(\alpha)} \in V_i \\ \varphi_\alpha(\hat{h}_i) & \text{otherwise} \end{cases} \qquad \text{and} \qquad \tilde{\pi}_i(\sigma) = 1. \tag{4.2.32}$$

An important observation is that $\tilde{\pi}_i$ maps commutators of even pairs of generators to commutators of even pairs of generators (or the identity) so $\tilde{\pi}_i(K) \leq K$.

By assumption we have $\sigma \in H^E \pmod K$. Then there exists an even length sequence of clauses $h_{r_1} h_{r_2} ... h_{r_t}$ with $w = \sigma w_k$ and $w_k \in K$. We claim:

1. For all $i \in [l]$, $\alpha \in \{1, 2, 3\}$ we have : $\varphi_\alpha(\tilde{\rho}_i(h_{r_1} h_{r_2} ... h_{r_t})) = \varphi_\alpha(\tilde{\pi}_i(w_k)) \in w_k$.

2. There exists an $i' \in [l]$ satisfying $\varphi_\sigma(\tilde{\rho}_{i'}(h_{r_1} h_{r_2} ... h_{r_t})) = \sigma$.

The proof of the first equality in Claim 1 follows identically to the proof of Claim 1 in Theorem 4.2.3. The second inequality holds because $\tilde{\pi}_i(K) \leq K$.

Proving Claim 2 requires a little more work. The complicating issue is that we can encounter a case where $\varphi_\sigma(\hat{\rho}_i(h_j)) = \sigma$ even if $h_j \notin S_i$. Thus the equation

$$\varphi_\sigma(h_j) = \varphi_\sigma\left(\prod_i \hat{\rho}_i(h_j)\right) \tag{4.2.33}$$

might not hold, and we can't simply copy the proof of Claim 2 in Theorem 4.2.3. However, copying the proof of Claim 2 does give us that there exists an $i' \in [l]$ for which $\varphi_\sigma(\rho_{i'}(h_{r_1}h_{r_2}...h_{r_t})) = \sigma$, that is, the claim holds without the tilde. Let $n_{i'}$ be the number of clauses in the sequence $h_{r_1}h_{r_2}...h_{r_t}$ not contained in $S_{i'}$, that is

$$n_{i'} = \left|\{j \in [l] : r_{r_j} \notin S_{i'}\}\right|. \tag{4.2.34}$$

We claim $n_{i'}$ is even. To see this, note that any word $w \in K$ contains each generator $x_i^{(\alpha)}$ an even number of times, since the even commutators contain the generators $x_i^{(\alpha)}$ an even number of times, and the $x_i^{(\alpha)}$ are self-inverse. Then the number occurrences of all the $x_i^{(1)} \notin V_{i'}$ in the word $h_{r_1}h_{r_2}...h_{r_t}$ must be even (the 1 here is arbitrary, all that matters is that we fix a player). But this is equal to $n_{i'}$ mod 2, and we conclude $n_{i'}$ is even. Finally, we note that

$$\varphi_\sigma(\tilde{\rho}_{i'}(h_{r_1}h_{r_2}...h_{r_t})) = \varphi_\sigma(\rho_{i'}(h_{r_1}h_{r_2}...h_{r_t}))(\varphi_\sigma(\hat{h}_{i'}))^{n_{i'}} = \varphi_\sigma(\rho_{i'}(h_{r_1}h_{r_2}...h_{r_t})) = \sigma. \tag{4.2.35}$$

Using the fact that $n_{i'}$ is even and σ has order two.

Combining Claims 1 and 2 with Equation (4.2.13) gives

$$\tilde{\rho}_{i'}(h_{r_1}h_{r_2}...h_{r_t})) = \prod_\alpha \left(\varphi_\alpha\left(\tilde{\rho}_{i'}(h_{r_1}h_{r_2}...h_{r_t})\right)\right) \varphi_\sigma\left(\tilde{\rho}_{i'}(h_{r_1}h_{r_2}...h_{r_t})\right) \tag{4.2.36}$$

$$= \prod_\alpha \left(\varphi_\alpha\left(\tilde{\rho}_{i'}(h_{r_1}h_{r_2}...h_{r_t})\right)\right) \sigma \tag{4.2.37}$$

with $\tilde{\rho}_{i'}(h_{r_1}h_{r_2}...h_{r_t})) \in \langle S_{i'}\rangle \cap H^E$ and $\prod_\alpha \left(\varphi_\alpha\left(\tilde{\rho}_{i'}(h_{r_1}h_{r_2}...h_{r_t})\right)\right) \in K$. □

To close this section we observe that Theorem 4.2.4 implies proving Theorem 4.1.6 for all 3XOR games with a connected clause graph $\mathcal{G}_{123}$ proves the result for all 3XOR games. To see why, consider a 3XOR game G with clause set S, a disconnected clause graph and

$\sigma \in H^E \pmod K$. Theorem 4.2.4 says that we can find a connected subset of clauses $S' \subset S$ with $\sigma \in \langle S' \rangle \cap H^E \pmod K$. Then, we restrict to the 3XOR game G' defined only on these clauses and note is has a fully connected clause graph. Theorem 4.1.6 then says $\sigma \in \langle S' \rangle$, which implies $\sigma \in \langle H \rangle$ for the original game G as well. For this reason, we assume the clause graph $\mathcal{G}_{123}$ is connected in Section 4.2.4.

4.2.4 Proof of Theorem 4.1.6

The proof is involved, and we will build up to it slowly over the course of many lemmas. First, we recap the theorem and give an outline of the proof. Note that notation, particularly the w, w' and $\tilde{w}$, in this outline is simplified, and does not match the notation used in the remainder of this section.

Theorem 4.1.6 (Repeated). σ is contained in H *iff*, after modding out by K, the coset containing σ is contained in H^E. That is:

$$\sigma \in H \Leftrightarrow [\sigma]_K \in H^E \pmod K. \tag{4.2.38}$$

Proof Outline. The forwards direction is immediate. The backwards direction takes work.

Our starting point is the observation that $[\sigma]_K \in H \pmod K$ implies there exists some $h \in H$ satisfying $h = \sigma w$, with $w \in K$. We will modify this word by right multiplying by words in H until we have removed the w portion, producing a word $\sigma \in H$. We refer to this process as "clearing" the word w from the word h. To begin, we break w into three words: since G_1, G_2 and G_3 group elements all commute with each other we can separate them out and write $w = w_1 w_2 w_3$ with each $w_\alpha \in G_\alpha \cap K$. Then we clear the word w one w_α at a time.

In Section 4.2.4 we show how to construct a word $\tilde{w} = w_1 \tilde{w}_2 \tilde{w}_3 \in H$, where words $\tilde{w}_2 \in G_2 \cap K$ and $\tilde{w}_3 \in G_3 \cap K$ are arbitrary. To do this we define a homomorphism φ_1^* which maps any word $v_1 \in G_1$ to a sequences of clauses in H whose product equals v_1 when projected to the G_1 subgroup.

Multiplying h by $\tilde{w}^{-1}$ produces a word $w'\sigma = w_2' w_3' \sigma$ with $w_2' \in G_2 \cap K$ and $w_3' \in G_3 \cap K$. Importantly w' contains no terms in the G_1 subgroup, that is, we have cleared the G_1 portion of the word w. Our next step is to right multiply by a word which will clear the w_2' term,

while not introducing any new terms in the G_1 subgroup. We do this by constructing another homomorphism $\varphi^*_{2,1}$, which takes a word in $v_2 = G_2$ and produces a word in H which equals v_2 in the G_2 subgroup and projects to the identity in the G_1 subgroup whenever possible. Details are given in Section 4.2.4.

Section 4.2.4 goes over the process of removing the w_1 and w_2 words from h. The final result is a word

$$w'' = \varphi^*_{2,1}\left(w'_2\right)^{-1} w' = w''_3\sigma$$

where $w''_3 \in G_3 \cap K$.

Finally, we want to clear the word w''_3 without introducing any words in the G_1 or G_2 subgroups. Unlike previous sections, we do not do this by constructing a homomorphism. Instead, in section Section 4.2.4 we construct a series of gadgets designed to make a word easier to clear. Then, in Section 4.2.4 we introduce gadgets into the word w''_3, and clear the word with the gadgets introduced. This procedure relies on the fact we have already cleared words w_1 and w_2 and special structure of the K subgroup. □

We now begin the proof in earnest.

Projectors and simple right inverse.

We start with some useful notation. Recall the projector $\varphi_\alpha : G \to G_\alpha$ onto group elements corresponding to player α defined in Section 4.2.1 . It is a homomorphism, defined by

$$\varphi_\alpha(x_i^{(\beta)}) := \begin{cases} x_i^{(\beta)} & \text{if } \alpha = \beta \\ 1 & \text{otherwise} \end{cases} \tag{4.2.39}$$

and

$$\varphi_\alpha(\sigma) = 1. \tag{4.2.40}$$

We also defined a projector onto the σ subgroup, $\varphi_\sigma : G \to \{\sigma, 1\}$ which satisfies

$$\varphi_\sigma(x_i^{(j)}) = 1 \text{ and } \varphi_\sigma(\sigma) = \sigma. \tag{4.2.41}$$

We use the notation φ^* to refer to right inverses of φ. Because the map φ is many to one, there are many choices of right inverse. We define several.

We first define the simple right inverse $\varphi_\alpha^* : G_\alpha \to H$ which maps each $x_i^{(\alpha)}$ to a single clause in S. For ease of notation, we give the definition when $\alpha = 1$. φ_1^* is a homomorphism which acts on the generators of G_1 by

$$\varphi_1^*(x_i^{(1)}) = h_j \tag{4.2.42}$$

where $j \in [m]$ is chosen so that $\varphi_1(h_j) = x_i^{(1)}$. Note that some clause $x_i^{(1)} x_j^{(2)} x_k^{(3)} \sigma^l$ must exist in S or else the question $x_i^{(1)}$ is never asked, and the group element $x_i^{(1)}$ can be removed from the game group (this can be viewed as a special case of the proof given in Section 4.2.3 that we can assume the game group is connected). If there are multiple clauses which contain the element $x_i^{(1)}$, we pick one arbitrarily. To verify φ_1^* is indeed a homomorphism, we can check

$$\begin{aligned}
\varphi_1^*(x_i^{(1)})^2 &= h_j^2 && (4.2.43)\\
&= x_{a_j}^{(1)} x_{b_j}^{(2)} x_{c_j}^{(3)} \sigma^{s_j} x_{a_j}^{(1)} x_{b_j}^{(2)} x_{c_j}^{(3)} \sigma^{s_j} && (4.2.44)\\
&= \left(x_{a_j}^{(1)}\right)^2 \left(x_{b_j}^{(2)}\right)^2 \left(x_{c_j}^{(3)}\right)^2 (\sigma^{s_j})^2 && (4.2.45)\\
&= 1. && (4.2.46)
\end{aligned}$$

φ_α^* for general α is defined similarly.

Identity preserving right inverse.

The next right inverse we define, $\varphi_{\alpha,\beta}^*$, acts as a right inverse to φ_α while also producing a word $h \in H$ satisfying $\varphi_\beta(h) = 1$ whenever such a mapping is possible. In order to define $\varphi_{\alpha,\beta}^*$ as a homomorphism, we restrict it's action to the subgroup of even length words G_α^E.

Our first result is a general "trick" we will use to construct homomorphisms on the even

subgroups.

Lemma 4.2.5. *Let $f : G_\alpha \to H$ be an arbitrary map. Define $\tilde{f} : G_\alpha^E \to H^E$ by its action on the generators of G_α^E*

$$\tilde{f}(x_i^{(\alpha)} x_j^{(\alpha)}) = f(x_i^{(\alpha)}) f(x_j^{(\alpha)})^{-1}, \tag{4.2.47}$$

and extend it to act on words in G_α^E in the natural way, so

$$\tilde{f}(x_i^{(\alpha)} x_j^{(\alpha)} x_k^{(\alpha)} x_l^{(\alpha)}) = \tilde{f}(x_i^{(\alpha)} x_j^{(\alpha)}) \tilde{f}(x_k^{(\alpha)} x_l^{(\alpha)}). \tag{4.2.48}$$

Then $\tilde{f}$ is a homomorphism.

Proof. The only non-trivial relation to check is that $\tilde{f}(x_i^{(\alpha)} x_j^{(\alpha)} x_j^{(\alpha)} x_k^{(\alpha)}) = \tilde{f}(x_i^{(\alpha)} x_k^{(\alpha)})$. But we see

$$\begin{aligned} \tilde{f}(x_i^{(\alpha)} x_j^{(\alpha)} x_j^{(\alpha)} x_k^{(\alpha)}) &= \tilde{f}(x_i^{(\alpha)}) \tilde{f}(x_j^{(\alpha)})^{-1} \tilde{f}(x_j^{(\alpha)}) \tilde{f}(x_k^{(\alpha)})^{-1} && (4.2.49) \\ &= \tilde{f}(x_i^{(\alpha)} x_k^{(\alpha)}) && (4.2.50) \end{aligned}$$

and the result follows. □

Next, we define $\varphi_{a,b}^*$ and prove its existence in the following lemma.

Lemma 4.2.6. *For all $\alpha \neq \beta \in [3]$, there exists a homomorphism $\varphi_{\alpha,\beta}^* : G_\alpha^E \to H^E$ satisfying*

A1. $\varphi_\alpha(\varphi_{\alpha,\beta}^*(w)) = w$ *for all* $w \in G_\alpha^E$.

A2. $\varphi_\beta\left(\varphi_{\alpha,\beta}^*(w)\right) = 1$ *whenever there exists an* $h \in H^E$ *satisfying* $\varphi_\beta(h) = 1$ *and* $\varphi_\alpha(h) = w$. *An important consequence of this is that* $\varphi_\beta\left(\varphi_{\alpha,\beta}^*\left(\varphi_\alpha(h)\right)\right) = 1$ *for any* $h \in H^E$ *satisfying* $\varphi_\beta(h) = 1$.

Proof. For ease of notation, we prove the result when $\alpha = 1$, $\beta = 2$. The proof is identical for other α, β.

Recall the (multi)graph $\mathcal{G}_{12}$, defined in Section 4.2.1. $\mathcal{G}_{12}$ has N^2 vertices, labeled by the group elements $x_1^{(1)}, x_2^{(1)}, ..., x_N^{(1)}, x_1^{(2)}, x_2^{(2)}, ..., x_N^{(2)}$. We identify vertices in the graph with

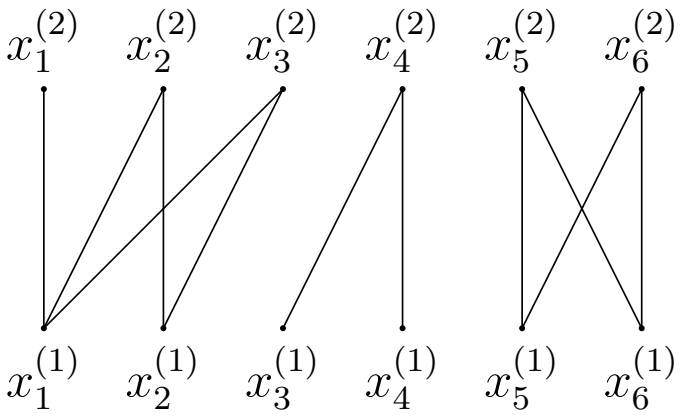

Figure 4-3: Sample graph $\mathcal{G}_{12}$ for a game with alphabet size $N = 6$ and $m = 11$ clauses. The middle component for example corresponds to clauses $x_3^{(1)}x_4^{(2)}x_{k_1}\sigma^{l_1}$ and $x_4^{(1)}x_4^{(2)}x_{k_2}\sigma^{l_2}$, where $k_1, k_2 \in [N]$ and $l_1, l_2 \in \{0, 1\}$ are arbitrary.

generators of game group G, and abuse notation slightly by referring to the two objects interchangeably. Edges in the graph correspond to clauses; the graph has one edge $(x_i^{(1)}, x_j^{(2)})$ for every clause $x_i^{(1)}x_j^{(2)}x_k^{(3)}\sigma^{(l)}$ in S. ($k \in [N]$ and $l \in \{0, 1\}$ are arbitrary.) Then $\mathcal{G}_{12}$ is bipartite, with the vertices $x_i^{(1)}$ for $i \in [N]$ forming one half of the graph and $x_j^{(2)}$ for $j \in [N]$ forming the other. See Figure 4-3 for an example.

Any path

$$P\left(x_{i_1}^{(1)}, x_{j_t}^{(2)}\right) = \left((x_{i_1}^{(1)}, x_{j_1}^{(2)}), (x_{j_1}^{(2)}, x_{i_2}^{(1)}), (x_{i_2}^{(1)}, x_{j_2}^{(2)}), ..., (x_{i_t}^{(1)}, x_{j_t}^{(2)})\right) \tag{4.2.51}$$

in the graph can be identified with some word

$$x_{i_1}^{(1)}x_{j_1}^{(2)}x_{k_1}^{(3)}\sigma^{l_1}x_{i_2}^{(1)}x_{j_1}^{(2)}x_{k_2}^{(3)}\sigma^{l_2}x_{i_2}^{(1)}x_{j_2}^{(2)}x_{k_3}^{(3)}\sigma^{l_3}...x_{i_t}^{(1)}x_{j_t}^{(2)}x_{k_{2t}}^{(3)}\sigma^{l_{2t}} \in H. \tag{4.2.52}$$

Abusing notation slightly, we refer to the word in H as the path $P\left(x_{i_1}^{(1)}, x_{j_t}^{(2)}\right)$.

Now, consider a word $P\left(x_{i_1}^{(1)}, x_{j_t}^{(2)}\right)$ corresponding to a path in $\mathcal{G}_{12}$ from a vertex associated with player 1 to a vertex associated with player 2. Note the path has odd length because $\mathcal{G}_{12}$ is bipartite, so the word $P\left(x_{i_1}^{(1)}, x_{j_t}^{(2)}\right)$ consists of an odd sequence of clauses. All generators in G_1, G_2 other than $x_{i_1}^{(1)}$ and $x_{j_t}^{(2)}$ are repeated adjacent to each other in the word $P\left(x_{i_1}^{(1)}, x_{j_t}^{(2)}\right)$.

These generators cancel, and so

$$P\left(x_{i_1}^{(1)}, x_{j_t}^{(2)}\right) = x_{i_1}^{(1)} x_{j_t}^{(2)} x_{k_1}^{(3)} x_{k_2}^{(3)} \ldots x_{k_{2t}}^{(3)} \sigma^{l_1+l_2+\ldots l_{2t}}. \tag{4.2.53}$$

Hence,

$$\varphi_1\left(P(x_{i_1}^{(1)}, x_{j_t}^{(2)})\right) = x_{i_1}^{(1)} \tag{4.2.54}$$

and

$$\varphi_2\left(P(x_{i_1}^{(1)}, x_{j_t}^{(2)})\right) = x_{j_t}^{(2)}. \tag{4.2.55}$$

Next, note that the multigraph $\mathcal{G}_{12}$ naturally partitions into components. Pick one representative vertex $x_j^{(2)}$ from each component. We define a map $r_{1,2}$ that takes generators of G_1 or G_2 (vertices in $\mathcal{G}_{12}$) to the unique representative vertex in the same component. Formally, $r_{1,2}$ maps a vertex $x_i^{(\alpha)}$ in $\mathcal{G}_{12}$ to the representative vertex in the connected component containing $x_i^{(\alpha)}$. Note that α could be either 1 or 2, but we defined representative vertices to be in G_2, so $r_{1,2}\left(x_i^{(\alpha)}\right) \in G_2$.[13]

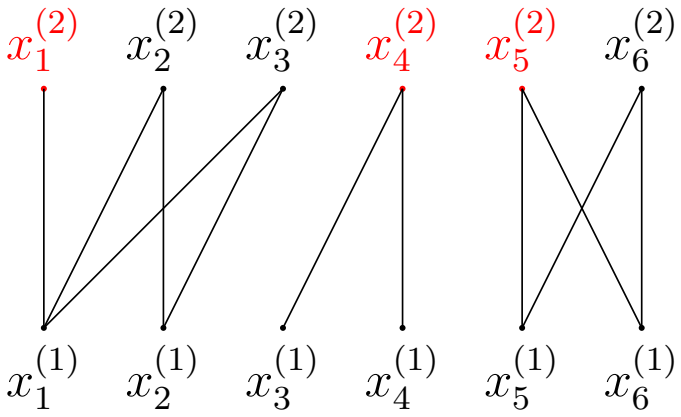

Figure 4-4: Sample graph repeated from Figure 4-3 with a choice of representative vertices indicated in red.

$r_{1,2}$ maps generators which square to the identity to generators which square to the

[13]Representative elements in G_1 are used to define the map $r_{2,1}$, which is used construct the right inverse $\varphi_{2,1}^*$. To keep track of this somewhat subtle notation, recall $\varphi_{\alpha,\beta}^*$ is the right inverse of φ_α that tries to preserve the identity in G_β subgroup.

identity, so we can extend it to a homomorphism acting on words in G_1 or G_2. We abuse notation and refer to both of these homomorphisms as $r_{1,2}$. [14]

Next, fix paths $\overline{P}\left(x_i^{(1)}, r_{1,2}\left(x_i^{(1)}\right)\right)$ between the vertices of G_1 and the connected representative vertex (see Figure 4-5). Define the homomorphism $\varphi_{1,2}^* : G_1^E \to H^E$ by its action on the generators of G^E,

$$\varphi_{1,2}^*\left(x_i^{(1)} x_j^{(1)}\right) := \overline{P}\left(x_i^{(1)}, r_{1,2}\left(x_i^{(1)}\right)\right) \overline{P}\left(x_j^{(1)}, r_{1,2}\left(x_j^{(1)}\right)\right)^{-1}. \tag{4.2.56}$$

Recall the abuse of notation defined above, so $\overline{P}\left(x_i^{(1)}, r_{1,2}\left(x_i^{(1)}\right)\right)$ defines both a path in the graph $\mathcal{G}_{12}$ and a word in H. $\varphi_{1,2}^*$ is a valid homomorphism by Lemma 4.2.6.

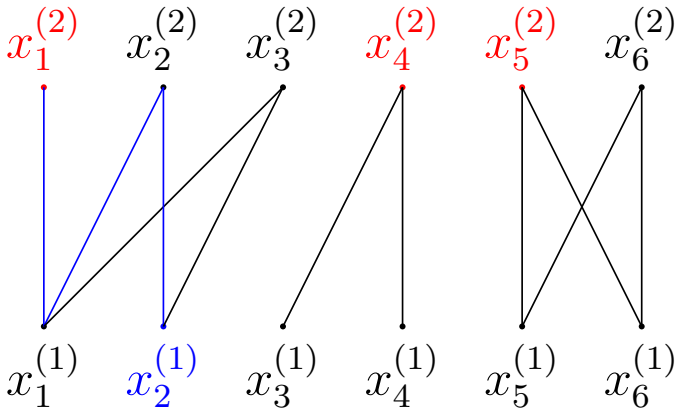

Figure 4-5: Sample graph with representative vertices indicated in red and the path $\overline{P}\left(x_2^{(1)}, r_{1,2}\left(x_2^{(1)}\right)\right)$ indicated in blue. This path corresponds to a word $x_2^{(1)} x_2^{(2)} x_{k_1}^{(3)} \sigma^{l_1} x_1^{(1)} x_2^{(2)} x_{k_2}^{(3)} \sigma^{l_2} x_1^{(1)} x_1^{(2)} x_{k_3}^{(3)} \sigma^{l_3}$, where $k_1, k_2, k_3 \in [N]$ and $l_1, l_2, l_3 \in \{0, 1\}$ are arbitrary.

It remains to show $\varphi_{1,2}^*$ satisfies Properties A1 and A2. Property A1 follows from Equation (4.2.54), which gives

$$\varphi_1\left(\varphi_{1,2}^*(x_i^{(1)} x_j^{(1)})\right) = x_i^{(1)} (x_j^{(1)})^{-1} = x_i^{(1)} x_j^{(1)}. \tag{4.2.57}$$

[14] $r_{1,2}$ does not preserve commutation between elements of G_1 and G_2 (that is, $\left[r_{1,2}(x_i^{(1)}), r_{1,2}(x_j^{(2)})\right]$ may not equal 1) so we cannot extend $r_{1,2}$ to a homomorphism simultaneously mapping G_1 and G_2 into G_2. Instead, there are two separate homormorphisms, one mapping $G_1 \to G_2$ and one mapping $G_2 \to G_2$, and both are denoted by $r_{1,2}$. This somewhat technical issue does not affect the proof.

To prove property A2 we first show that

$$r_{1,2}(\varphi_2(h)) = \varphi_2(\varphi_{1,2}^*(\varphi_1(h))) \tag{4.2.58}$$

for any $h \in H^E$. The claim can be verified by checking the action of the two maps on generators $h_i h_j$ of H^E:

$$\begin{aligned}
\varphi_2(\varphi_{1,2}^*(\varphi_1(h_i h_j))) &= \varphi_2(\varphi_{1,2}^*(\varphi_1(x_{a_i}^{(1)} x_{b_i}^{(2)} x_{c_i}^{(3)} x_{a_j}^{(1)} x_{b_j}^{(2)} x_{c_j}^{(3)} \sigma^{s_i+s_j}))) && (4.2.59)\\
&= \varphi_2(\varphi_{1,2}^*(x_{a_i}^{(1)} x_{a_j}^{(1)})) && (4.2.60)\\
&= r_{1,2}\left(x_{a_i}^{(1)}\right) r_{1,2}\left(x_{a_j}^{(1)}\right) && (4.2.61)\\
&= r_{1,2}\left(x_{b_i}^{(2)}\right) r_{1,2}\left(x_{b_j}^{(2)}\right) && (4.2.62)\\
&= r_{1,2}(\varphi_2(h_i h_j)) && (4.2.63)
\end{aligned}$$

Line (4.2.61) follows from Equation (4.2.55). The key observation comes in line (4.2.62). Because a_i and b_i are both in the clause h_i, they are in the same connected component in the graph $\mathcal{G}_{12}$. Then they have the same representative vertex and

$$r_{1,2}\left(x_{a_i}^{(1)}\right) = r_{1,2}\left(x_{b_i}^{(2)}\right). \tag{4.2.64}$$

Line (4.2.62) follows.

Now any $h \in H$ satisfying $\varphi_2(h) = 1$ must have even length, so $h \in H^E$ and we have

$$\varphi_2(\varphi_{1,2}^*(\varphi_1(h))) = r_{1,2}(\varphi_2(h)) = r_{1,2}(1) = 1. \tag{4.2.65}$$

Using the fact that $r_{1,2}$ is a homomorphism in the last two equalities. This proves Property A2, and completes the proof. □

The next lemma proves that right inverses φ_α^* and $\varphi_{\alpha,\beta}^*$ map within the K subgroup. That is, they map words in $K \cap G_\alpha^E$ to words in $K \cap H^E$.

Lemma 4.2.7. *Let $v \in K \cap G_\alpha^E$ be arbitrary. Then*

$$\varphi_\alpha^*(v) \in K \cap H^E \tag{4.2.66}$$

and

$$\varphi_{\alpha,\beta}^*(v) \in K \cap H^E \tag{4.2.67}$$

for all $\beta \neq \alpha$.

Proof. For notational convenience we prove the result when $\alpha = 1$, $\beta = 2$.

The proof is mechanical: any word $v \in K \cap G_1^E$ can be written

$$v = \prod_i u_i \left[x_{a_{i_1}}^{(1)} x_{a_{i_2}}^{(1)}, x_{a_{i_3}}^{(1)} x_{a_{i_4}}^{(1)} \right] u_i^{-1}. \tag{4.2.68}$$

with $u_i \in G_1$ arbitrary. We pick labels $b_{i_1}, ..., b_{i_4}, c_{i_1}, ..., c_{i_4} \in [N]$ and $s_{i_1}, ..., s_{i_4} \in \{0, 1\}$ so that

$$\varphi_1^* \left(x_{a_{i_j}}^{(1)} \right) = x_{a_{i_j}}^{(1)} x_{b_{i_j}}^{(2)} x_{c_{i_j}}^{(3)} \sigma^{s_{i_j}} \tag{4.2.69}$$

for all $x_{a_{i_j}}$. Then

$$\varphi_1^*(v) = \prod_i \varphi_1^*(u_i) \left[\varphi_1^*(x_{a_{i_1}}^{(1)}) \varphi_1^*(x_{a_{i_2}}^{(1)}), \varphi_1^*(x_{a_{i_3}}^{(1)}) \varphi_1^*(x_{a_{i_4}}^{(1)}) \right] \varphi_1^*(u_i)^{-1} \tag{4.2.70}$$

$$= \prod_i \varphi_1^*(u_i) \left[x_{a_{i_1}}^{(1)} x_{a_{i_2}}^{(1)}, x_{a_{i_3}}^{(1)} x_{a_{i_4}}^{(1)} \right] \left[x_{b_{i_1}}^{(2)} x_{b_{i_2}}^{(2)}, x_{b_{i_3}}^{(2)} x_{b_{i_4}}^{(2)} \right] \left[x_{c_{i_1}}^{(3)} x_{c_{i_2}}^{(3)}, x_{c_{i_3}}^{(3)} x_{c_{i_4}}^{(3)} \right] \varphi_1^*(u_i)^{-1} \in K. \tag{4.2.71}$$

noting that any factors of σ cancel in the commutator.

A similar argument shows $\varphi_{1,2}^*(v) \in K$. To start assume $v \in K \cap G^E$ and write

$$\varphi_{1,2}^*(v) = \prod_i \varphi_{1,2}^*(u_i) \left[\varphi_{1,2}^*(x_{i_1}^{(1)})\varphi_{1,2}^*(x_{i_2}^{(1)}), \varphi_{1,2}^*(x_{i_3}^{(1)})\varphi_{1,2}^*(x_{i_4}^{(1)})\right] \varphi_{1,2}^*(u_i)^{-1} \tag{4.2.72}$$

$$= \prod_i \varphi_{1,2}^*(u_i) \left(\prod_{\alpha=1}^{3} \left[\varphi_\alpha\left(\varphi_{1,2}^*(x_{i_1}^{(1)})\varphi_{1,2}^*(x_{i_2}^{(1)})\right), \varphi_\alpha\left(\varphi_{1,2}^*(x_{i_3}^{(1)})\varphi_{1,2}^*(x_{i_4}^{(1)})\right)\right]\right) \varphi_{1,2}^*(u_i)^{-1} \tag{4.2.73}$$

We can show this word is in K by noting the words $\varphi_\alpha\left(\varphi_{1,2}^*(x_{i_1}^{(1)})\varphi_{1,2}^*(x_{i_2}^{(1)})\right)$ and $\varphi_\alpha\left(\varphi_{1,2}^*(x_{i_3}^{(1)})\varphi_{1,2}^*(x_{i_4}^{(1)})\right)$ are even-length products of clauses in H for any α, then repeatedly applying to commutator identities

$$[x, yz] = [x, y]\, y^{-1}\, [x, z]\, y \tag{4.2.74}$$

$$\text{and } [xy, z] = y^{-1}\, [x, z]\, y\, [y, z] \tag{4.2.75}$$

to show those words are in K. The full argument is given in an appendix (Lemma 4.3.1). □

An important consequence of Lemma 4.2.7 is the following corollary.

Corollary 4.2.8. *Let $v \in K \cap G_\alpha^E$ be arbitrary and $\alpha \neq \beta$. Then*

$$\varphi_\sigma(\varphi_\alpha^*(v)) = \varphi_\sigma(\varphi_{\alpha,\beta}^*(v)) = 1. \tag{4.2.76}$$

Proof. By Lemma 4.3.4, $\varphi_\sigma(k) = 1$ for all $k \in K$. Then, by Lemma 4.2.7 $\varphi_\alpha^*(v) \in K$. Hence

$$\varphi_\sigma\left(\varphi_\alpha^*(v)\right) = 1 \tag{4.2.77}$$

The proof for $\varphi_\sigma\left(\varphi_{\alpha,\beta}^*(v)\right)$ is identical. □

Clearing the G_1 and G_2 subgroups

The next lemma makes critical use of right inverses φ_α^* and $\varphi_{\alpha,\beta}^*$. It should be be thought of as a "pre-processing" step, that puts words in a convenient form to prove Theorem 4.1.6.

Lemma 4.2.9. *If there exists a word $w \in H^E$ satisfying $w = \sigma \pmod K$, then there exists a word w' in H^E satisfying:*

1. $w' = \sigma \pmod K$

2. $\varphi_1(w') = \varphi_2(w') = 1$.

Proof. We construct w' by right multiplying w by $\varphi_1^*(\varphi_1(w^{-1}))$ to clear the G_1 subgroup elements, then multiplying by $\varphi_{2,1}^*\left(\varphi_2\left(\left(w\varphi_1^*(\varphi_1(w^{-1}))\right)^{-1}\right)\right)$ to clear the G_2 subgroup. In math:

$$w' = w\varphi_1^*\left(\varphi_1\left(w^{-1}\right)\right) \cdot \varphi_{2,1}^*\left(\varphi_2\left(\left(w\varphi_1^*\left(\varphi_1\left(w^{-1}\right)\right)\right)^{-1}\right)\right). \tag{4.2.78}$$

First, we show that $\varphi_{2,1}^*\left(\varphi_2\left(\left(w\varphi_1^*(\varphi_1(w^{-1}))\right)^{-1}\right)\right)$ is well defied, and that $w' = \sigma \pmod K$. By assumption, $w = \sigma \pmod K$. Equivalently, $w = k\sigma$, for some $k \in K$. Then

$$\varphi_1(w) = \varphi_1(k\sigma) = \varphi_1(k) \in K \cap G_1^E \tag{4.2.79}$$

since φ_1 maps words in K to words inside K and words in H^E to words in G_1^E. φ_1 is a homomorphism, so we also have $\varphi_1(w^{-1}) \in K \cap G_1^E$. Then, by Lemma 4.2.7,

$$\varphi_1^*\left(\varphi_1(w^{-1})\right) \in K \cap H^E. \tag{4.2.80}$$

A similar argument shows $\varphi_2(w) \in K \cap G_2^E$. From this, and equation 4.2.80 it follows that

$$\varphi_2\left(\left(w\varphi_1^*\left(\varphi_1\left(w^{-1}\right)\right)\right)^{-1}\right) \in K \cap G_2^E \tag{4.2.81}$$

Then, by Lemma 4.2.7

$$\varphi_{2,1}^*\left(\varphi_2\left(\left(w\varphi_1^*\left(\varphi_1\left(w^{-1}\right)\right)\right)^{-1}\right)\right) \in K \cap H^E. \tag{4.2.82}$$

Putting this all together gives

$$w \cdot \varphi_1^* \left(\varphi_1 \left(w^{-1}\right)\right) \cdot \varphi_{2,1}^* \left(\varphi_2 \left(\left(w \varphi_1^* \left(\varphi_1 \left(w^{-1}\right)\right)\right)^{-1}\right)\right) = w \cdot 1 \cdot 1 \pmod{K} \tag{4.2.83}$$

$$= \sigma \pmod{K}, \tag{4.2.84}$$

as desired.

To show $\varphi_1(w') = \varphi_2(w') = 1$, set $h = w\varphi_1^* \left(\varphi_1 \left(w^{-1}\right)\right)$ and note

$$\varphi_1 (h) = \varphi_1 (w) \cdot \varphi_1 \left(\varphi_1^* \left(\varphi_1 \left(w^{-1}\right)\right)\right) \tag{4.2.85}$$

$$= \varphi_1(w) \cdot \varphi_1 \left(w^{-1}\right) \tag{4.2.86}$$

$$= 1. \tag{4.2.87}$$

$w \in H$ by assumption and $\varphi_1^* \left(\varphi_1 \left(w^{-1}\right)\right) \in H$ because $\mathrm{Im}(\varphi_1^*) \in H$. Then $h \in H$ and, by Property A2 of the map $\varphi_{2,1}^*$ and Equation (4.2.87) we have

$$\varphi_1 \left(\varphi_{2,1}^* \left(\varphi_2 (h)\right)\right) = 1. \tag{4.2.88}$$

The maps $\varphi_\alpha, \varphi_\alpha^*$, and $\varphi_{\alpha,\beta}^*$ are all homomorphisms, so we also have

$$\varphi_1 \left(\varphi_{2,1}^* \left(\varphi_2 \left(h^{-1}\right)\right)\right) = 1. \tag{4.2.89}$$

Then we put this all together to see

$$\varphi_1(w') = \varphi_1 \left(h \cdot \varphi_{2,1}^* \left(\varphi_2 \left(h^{-1}\right)\right)\right) \tag{4.2.90}$$

$$= \varphi_1 (h) \cdot \varphi_1 \left(\varphi_{2,1}^* \left(\varphi_2 \left(h^{-1}\right)\right)\right) \tag{4.2.91}$$

$$= 1. \tag{4.2.92}$$

using equations Equations (4.2.87) and (4.2.89) on the last line.

Additionally, property A1 of the map $\varphi_{2,1}^*$ gives

$$\varphi_2(w') = \varphi_2\left(h \cdot \varphi_{2,1}^*\left(\varphi_2\left(h^{-1}\right)\right)\right) \tag{4.2.93}$$

$$= \varphi_2(h) \cdot \varphi_2\left(\varphi_{2,1}^*\left(\varphi_2\left(h^{-1}\right)\right)\right) \tag{4.2.94}$$

$$= \varphi_2(h) \cdot \varphi_2\left(h^{-1}\right) \tag{4.2.95}$$

$$= 1. \tag{4.2.96}$$

Equations (4.2.84), (4.2.92) and (4.2.96) complete the proof. □

Gadgets for word processing

We are now almost ready to prove Theorem 4.1.6. Before we do this, we define two final homomorphisms $f_1, f_2 : G_3^E \to H^E$.[15] The fs are used to put words in an easy-to-cancel form, and are key in the proof of Theorem 4.1.6. They are defined in the following lemma.

Lemma 4.2.10. *For any $\alpha \in \{1,2\}$ and $\beta \neq \alpha \in \{1,2\}$ there exists a homomorphism $f_\alpha : G_3^E \to H^E$ which satisfies*

B1. If $\varphi_\beta\left(\varphi_{3,\beta}^*(v)\right) = 1$ then $\varphi_\beta\left(f_\beta(v)\right) = 1$

B2. $\varphi_\alpha\left(f_\beta(v)\right) = \varphi_\alpha\left(\varphi_{3,\beta}^*(v)\right)$

B3. $\varphi_\beta\left(\varphi_{3,\beta}^*\left(\varphi_3\left(f_\beta(v)\right)\right)\right) = 1.$

B4. $\varphi_\alpha\left(\varphi_{3,\alpha}^*\left(\varphi_3\left(f_\beta(v)\right)\right)\right) = \varphi_\alpha\left(\varphi_{3,\alpha}^*(v)\right)\right)$

for any $v \in G_3^E$.

To get a feel for the significance of Properties B1 to B4, assume existence of a word $w \in G_3^E \cap H$. Note $w \in H$ and $\varphi_2(w) = 1$ by assumption, so $\varphi_{3,2}^*(w) = 1$. Then $\varphi_2(f_2(w)) = \varphi_2(\varphi_{3,2}^*(w)) = 1$ by Property B1 and $\varphi_1(f_2(w)) = \varphi_1(\varphi_{3,2}^*(w))$ by Property B2. Thus,

$$\varphi_1((\varphi_{3,2}^*(w))^{-1} f_2(w)) = \varphi_2((\varphi_{3,2}^*(w))^{-1} f_2(w)) = 1. \tag{4.2.97}$$

[15] We could define analogues of f mapping from any G_α. We only need the maps from G_3, so we give the more specific construction for notational simplicity.

Now, define $w' = w(\varphi^*_{3,2}(w))^{-1} f_2(w)$. Since f_2 and $\varphi^*_{3,2}$ both map into H^E and $w \in H^E$ by assumption we have $w' = w(\varphi^*_{3,2}(w))^{-1} f_2(w) \in H^E$. We have $\varphi_1(w') = \varphi_2(w') = 1$ by our earlier observations and definition of w, and

$$\varphi_3(w') = \varphi_3(w)\varphi_3\left(\varphi^*_{3,2}(w)^{-1}\right)\varphi_3(f_2(w)) \tag{4.2.98}$$

$$= ww^{-1}\varphi_3(f_2(w)) = \varphi_3(f_2(w)). \tag{4.2.99}$$

We conclude that, up to a potential factor of σ, $w \in G_3^E \cap H \implies \varphi_3(f_2(w)) \in G_3^E \cap H$.

Properties B3 and B4 then tell us about the behaviour of this newly constructed word. Of particular importance is property B3, which tells us that, in particular

$$\varphi_2\left(\varphi^*_{3,2}\left(\varphi_3\left(f_2\left(x_i^{(3)}x_j^{(3)}\right)\right)\right)\right) = 1 \tag{4.2.100}$$

so any product of two generators "upgraded" by the map $\varphi_3 \circ f_2$ cancels to the identity in the G_2 subgroup under action by $\varphi^*_{3,2}$. Combining this observation with the intuition given in the proof sketch of Theorem 4.1.6 in Section 4.1.2 is the key to completing the proof. Property B4 is a slightly more technical result that lets us chain together the maps f_1 and f_2 in sequence.

Now we turn to the proof of Lemma 4.2.10. To prepare, we construct "gadget" words which will be used to in the definition of f_α. These words depend on the representative vertices chosen from the connected components of $\mathcal{G}_{13}$ and $\mathcal{G}_{12}$ when constructing the right inverses $\varphi^*_{3,1}$ and $\varphi^*_{3,2}$.

Recall the definition of the function $r_{3,1}$ which maps a vertex $x_i^{(\alpha)}$ in G_3 or G_1 to the representative vertex in the connected component of multigraph $\mathcal{G}_{31}$ containing $x_i^{(\alpha)}$. The vertex $x_i^{(\alpha)}$ can be in either G_1 or G_3. The representative vertex $r_{3,1}(x_i^{(\alpha)})$ is the one chosen when defining the map $\varphi^*_{3,1}$, and so is in G_1. The function $r_{3,2}$ mapping vertices in G_3 or G_2 to vertices in G_2 is defined similarly.

Finally, recall the hypergraph $\mathcal{G}_{123}$ defined in Section 4.2.1. Vertices are identified with elements $x_i^{(\alpha)}$, with $i \in [N]$, $\alpha \in \{1, 2, 3\}$. $\mathcal{G}_{123}$ contains a hyperedge $(x_i^{(1)}, x_j^{(2)}, x_k^{(3)})$ for each clause $x_i^{(1)}x_j^{(2)}x_k^{(3)}\sigma^l \in S$, where l have value 0 or 1. By the arguments of Section 4.2.3, we can assume this hypergraph is connected. Then there exist paths in $\mathcal{G}_{123}$ between any two vertices.

We pick, somewhat arbitrarily, a vertex $x_1^{(\alpha)} \in G_\alpha$, then fix a minimal length path in $\mathcal{G}_{123}$ from each representative vertex $r_{3,\alpha}(x_i^{(3)})$ to $x_1^{(\alpha)}$. Denote this path by $Q_\alpha\left(r_{3,\alpha}(x_i^{(3)})\right)$. Each path corresponds to a sequence of clauses, and hence a word in H. A sample hypergraph $\mathcal{G}_{123}$ is introduced in Figure 4-6, and a sample path is illustrated in Figure 4-8.

Given a sequence of clauses $P = h_{p_1}h_{p_2}...h_{p_s}$ corresponding to a path in $\mathcal{G}_{123}$, define the subsequence of clauses $s_\beta(P)$ to be the sequence including only pairs consisting of adjacent clauses which are connected through the G_β vertices. That is, $s_\beta(P)$ includes only adjacent clauses $h_{p_i}h_{p_{i+1}}$ which satisfy

$$\varphi_\beta(h_{p_i}) = \varphi_\beta(h_{p_{i+1}}). \tag{4.2.101}$$

Note $s_\beta(P)$ is likely not a path, since the pairs of clauses need not be connected to the other pairs. Finally, define words

$$\gamma_1\left(x_i^{(\alpha)}\right) := s_2\left(Q_1\left(r_{3,1}\left(x_i^{(\alpha)}\right)\right)\right) \text{ for } \alpha \in \{1,3\} \tag{4.2.102}$$

and

$$\gamma_2\left(x_i^{(3)}\right) := s_1\left(Q_2\left(r_{3,2}\left(x_i^{(3)}\right)\right)\right) \text{ for } \alpha \in \{2,3\}. \tag{4.2.103}$$

To full sequence of steps involved in the construction of γ_2 is visualized in Figures 4-6 to 4-9.

The following lemma summarizes the important properties of the gadget words $\gamma_2\left(x_i^{(3)}\right)$ and $\gamma_1\left(x_i^{(3)}\right)$.

Lemma 4.2.11. *The words $\gamma_2\left(x_i^{(3)}\right)$, defined as in Equation (4.2.103), satisfy the following properties.*

C1. $\varphi_1\left(\gamma_2\left(x_i^{(3)}\right)\right) = 1$.

C2. $\varphi_2\left(\varphi_{3,2}^*\left(\varphi_3\left(\gamma_2\left(x_i^{(3)}\right)\right)\right)\right) = \varphi_2\left(\varphi_{3,2}^*\left(x_i^{(3)}x_1^{(3)}\right)\right)$.

Words $\gamma_1(x_i^3)$ satisfy similar properties, with the 1 and 2 labels exchanged.

Proof. We show the γ_2 case. The proof in the γ_1 case is identical up to a change of index.

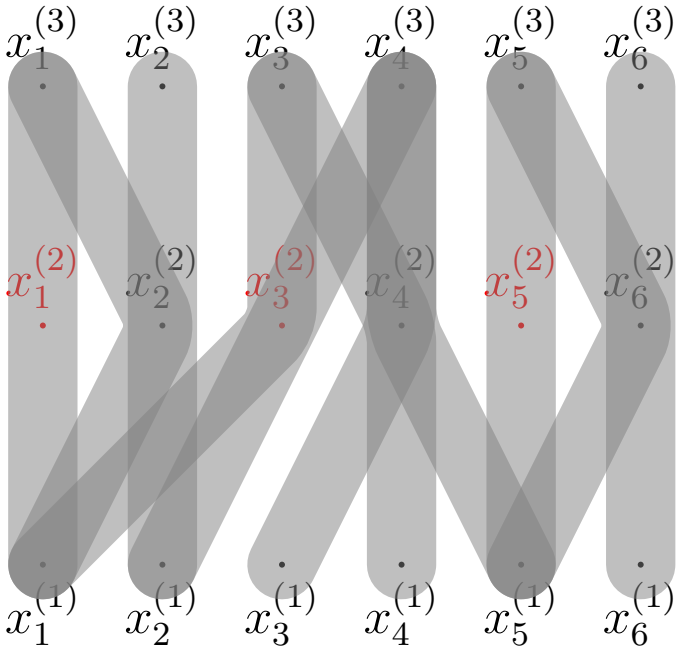

Figure 4-6: Sample hypergraph $\mathcal{G}_{123}$ for a game with alphabet size $N = 6$ and 11 clauses. Representative vertices in the image of the map $r_{3,2}$ are indicated in red. The hypergraph is generated by clause set (σ terms omitted since they don't affect the graph):

$$S = \{x_1^{(1)}x_1^{(2)}x_1^{(3)}, x_1^{(1)}x_2^{(2)}x_1^{(3)}, x_2^{(1)}x_2^{(2)}x_2^{(3)}, x_1^{(1)}x_3^{(2)}x_3^{(3)}, x_2^{(1)}x_3^{(2)}x_4^{(3)}, x_3^{(1)}x_4^{(2)}x_4^{(3)},$$
$$x_4^{(1)}x_4^{(2)}x_3^{(3)}, x_5^{(1)}x_4^{(2)}x_4^{(3)}, x_5^{(1)}x_6^{(2)}x_5^{(3)}, x_5^{(1)}x_5^{(2)}x_5^{(3)}, x_6^{(1)}x_6^{(2)}x_6^{(3)}\}$$

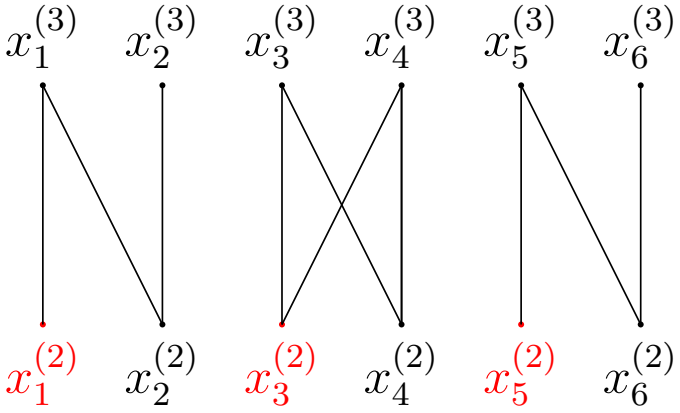

Figure 4-7: Graph $\mathcal{G}_{23}$ corresponding to the same set of clauses as used to generate the hypergraph in Figure 4-6. Representative vertices in the image of the map $r_{3,2}$ are indicated in red.

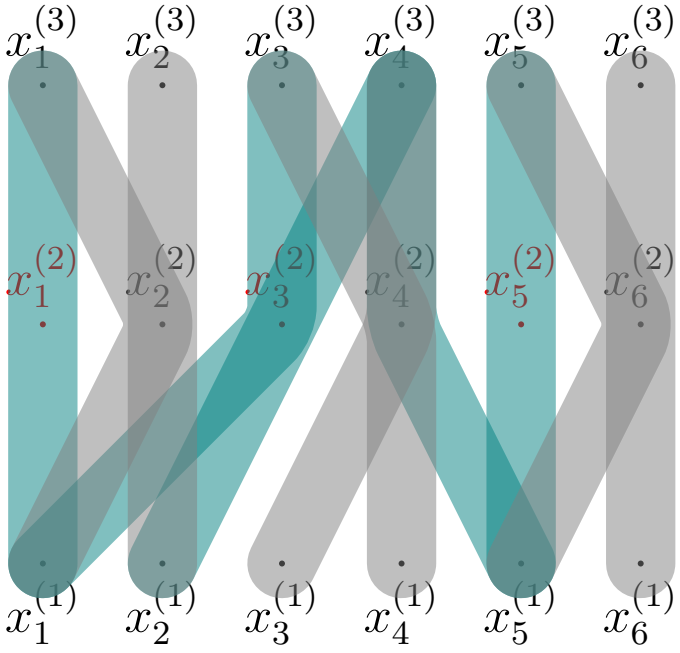

Figure 4-8: Hypergraph repeated from Figure 4-6. A choice of path $Q_2(x_5^{(2)})$ is indicated in teal.

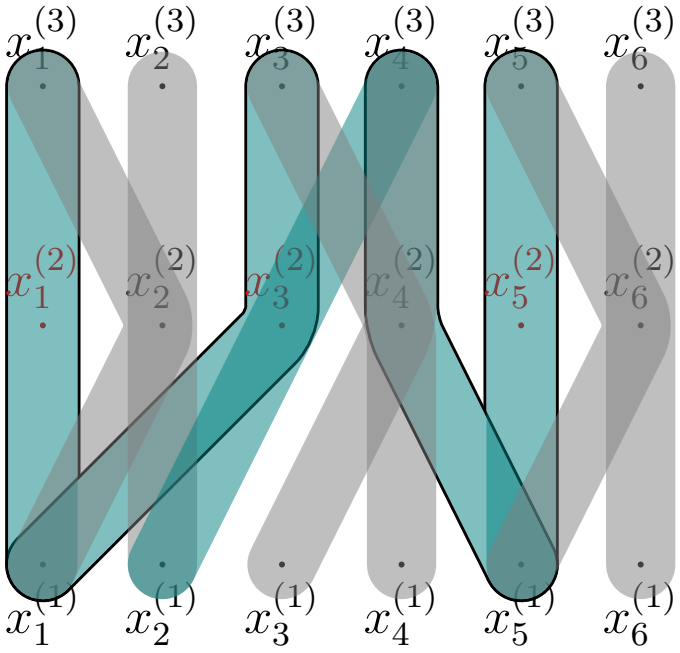

Figure 4-9: Hypergraph repeated from Figure 4-6. The path $Q_2(x_5^{(2)})$ is indicated in teal. The hyperedges making up $\gamma_2(x_5^{(2)})$ are outlined.

To begin the proof, we note the word $Q_2\left(r_{3,2}(x_i^{(3)})\right)$ corresponds to a minimal-length path and so there are never more than two adjacent clauses containing the same element in the G_1 subgroup. (If there were three or more adjacent hyperedges containing the same element in G_1, the middle hyperedges could be deleted and the path would remain connected, contradicting minimality). When defining the sequence/word $\gamma_1(x_i^3)$ all hyperedges in $Q_2\left(r_{3,2}(x_i^{(3)})\right)$ which didn't cancel on the G_1 subgroup were removed when we restricted to the s_1 subsequence, so we can write

$$\gamma_2\left(x_i^{(3)}\right) = h_{y_1}h_{z_1}h_{y_2}h_{z_2}, ..., h_{y_L}h_{z_L} \tag{4.2.104}$$

where $\varphi_1\left(h_{y_j}h_{z_j}\right) = 1$. This shows Property C1.

Next, we prove property C2. We start by numbering all clauses in the path $Q_2\left(r_{3,2}(x_i^{(3)})\right)$, so

$$Q_2\left(r_{3,2}(x_i^{(3)})\right) = h_{p_1}h_{p_2}....h_{p_R}. \tag{4.2.105}$$

Consider two adjacent hyperedges $h_{p_r}h_{p_{r+1}}$ in the path. Since these hyperedges appear in sequence they overlap on at least one vertex.

a) If this vertex is contained in G_1, then this pair of hyperedges is contained in the word $\gamma_2\left(x_i^{(3)}\right)$ and, using the notation of Equation (4.2.104), we have $h_{p_r}h_{p_{r+1}} = h_{y_j}h_{z_j}$ for some $j < L$.

b) Otherwise these hyperedges overlap on a vertex corresponding to a generator of either G_2 or G_3 (equivalently, these hyperedges overlap on a vertex contained in the subgraph $\mathcal{G}_{23}$). In that case $\varphi_3(h_{p_r})$ and $\varphi_3(h_{p_{r+1}})$ are in the same connected component in the graph $\mathcal{G}_{23}$ so $r_{3,2}(\varphi_3(h_{p_r})) = r_{3,2}(\varphi_3(h_{p_{r+1}}))$. Consequently,

$$\varphi_2\left(\varphi_{3,2}^*(\varphi_3(h_{p_r}h_{p_{r+1}}))\right) = r_{3,2}(\varphi_3(h_{p_r})\varphi_3(h_{p_{r+1}})) = 1. \tag{4.2.106}$$

The first equality holds by Equation (4.2.58) and the observation that $r_{3,2}(\varphi_3(h)) = r_{3,2}(\varphi_2(h))$ for any $h \in H$.

Now consider a contiguous string of hyperedges of the form $h_{z_i}, h_{p_{r+1}}, h_{p_{r+2}}..., h_{p_{r+r'}}, h_{y_{i+1}}$ contained in the path (4.2.105). Here h_{z_i} and $h_{y_{i+1}}$ belong to the path $\gamma_2\left(x_i^{(3)}\right)$, but $h_{p_{r+1}}...h_{p_{r+r'}}$ do not. By definition of the subsequence $\gamma_2\left(x_i^{(3)}\right)$, no adjacent hyperedges between $h_{p_{z_i}}$ and $h_{p_{y_{i+1}}}$ overlap on a vertex in the G_1 subspace, else they would be contained in the subsequence $\gamma_2\left(x_i^{(3)}\right)$, a contradiction. Then we apply the observation of the previous paragraph inductively to see

$$\varphi_2\left(\varphi_{3,2}^*(\varphi_3(h_{z_i}))\right) = \varphi_2\left(\varphi_{3,2}^*(\varphi_3(h_{p_{r+1}}))\right) = ... = \varphi_2\left(\varphi_{3,2}^*(\varphi_3(h_{p_{r+r'}}))\right) = \varphi_2\left(\varphi_{3,2}^*(\varphi_3(h_{y_{j+1}}))\right) \tag{4.2.107}$$

This shows

$$\varphi_2\left(\varphi_{3,2}^*(\varphi_3(h_{z_j}h_{y_{j+1}}))\right) = 1. \tag{4.2.108}$$

for any $j < L$. Now we use this observation inductively, and compute

$$\varphi_2\left(\varphi_{3,2}^*\left(\varphi_3\left(\gamma_2\left(x_i^{(3)}\right)\right)\right)\right) = \varphi_2\left(\varphi_{3,2}^*\left(\varphi_3\left(h_{y_1}h_{z_1}h_{y_2}h_{z_2}, ..., h_{y_L}h_{z_L}\right)\right)\right) \tag{4.2.109}$$

$$= \varphi_2\left(\varphi_{3,2}^*\left(\varphi_3\left(h_{y_1}h_{z_L}\right)\right)\right) \tag{4.2.110}$$

$$= \varphi_2\left(\varphi_{3,2}^*\left(x_i^{(3)}x_1^{(3)}\right)\right) \tag{4.2.111}$$

where we used on the last line the fact that $\varphi_3(h_{y_1})$ was in the same connected component in $\mathcal{G}_{23}$ as $x_i^{(3)}$, and $\varphi_3(h_{z_L})$ was in the same connected component as $x_1^{(3)}$, so

$$\varphi_2\left(\varphi_{3,2}^*\left(\varphi_3\left(h_{y_1}h_{z_L}\right)\right)\right) = r_{3,2}\left(\varphi_3(h_{y_1})\right) r_{3,2}\left(\varphi_3(h_{z_L})\right) \tag{4.2.112}$$

$$= r_{3,2}\left(x_i^{(3)}\right) r_{3,2}\left(x_1^{(3)}\right) \tag{4.2.113}$$

$$= \varphi_2\left(\varphi_{3,2}^*\left(x_i^{(3)}x_1^{(3)}\right)\right). \tag{4.2.114}$$

by definition of $r_{3,2}$ and Equation (4.2.61). This proves Property C2. □

Now we use the gadget words $\gamma_1(x_i^{(3)})$ and $\gamma_2(x_i^{(3)})$ to prove Lemma 4.2.10.

Proof (Lemma 4.2.10). Define the homomorphism $f_1 : G_3^E \to H^E$ by its action on the basis

elements

$$f_1(x_i^{(3)}x_j^{(3)}) = \varphi_{3,1}^*\left(x_i^{(3)}\right)\gamma_1\left(x_i^{(3)}\right)\left(\varphi_{3,1}^*\left(x_j^{(3)}\right)\gamma_1\left(x_j^{(3)}\right)\right)^{-1} \tag{4.2.115}$$

$$= \varphi_{3,1}^*\left(x_i^{(3)}\right)\gamma_1\left(x_i^{(3)}\right)\gamma_1\left(x_j^{(3)}\right)^{-1}\varphi_{3,1}^*\left(x_j^{(3)}\right), \tag{4.2.116}$$

with f_2 defined similarly. Both maps are homomorphisms by Lemma 4.2.6. It remains to show they satisfy Properties B1 to B4.

Property B2 follows from Property C1 of the words $\gamma_1(x_i^{(3)})$. Property C1 gives that $\varphi_2\left(\gamma_1(x_i^{(3)})\right) = 1$. Then, checking the action of f_1 on the generators of G_3^E we see

$$\varphi_2\left(f_1(x_i^{(3)}x_j^{(3)})\right) = \varphi_2\left(\varphi_{3,1}^*\left(x_i^{(3)}\right)\gamma_1(x_i^{(3)})\gamma_1(x_j^{(3)})^{-1}\varphi_{3,1}^*\left(x_j^{(3)}\right)\right) \tag{4.2.117}$$

$$= \varphi_2\left(\varphi_{3,1}^*\left(x_i^{(3)}\right)\varphi_{3,1}^*\left(x_j^{(3)}\right)\right) \tag{4.2.118}$$

$$= \varphi_2\left(\varphi_{3,1}^*\left(x_i^{(3)}x_j^{(3)}\right)\right). \tag{4.2.119}$$

The proof of Property B1 is similar to the proof of Property A2 of the map $\varphi_{\alpha,\beta}^*$. Recall the function $r_{3,1}$, defined to map a vertex $x_i^{(\alpha)}$ in G_1 or G_3 to the representative vertex $x_j^{(1)}$ in the connected component of graph G_{13} containing $x_i^{(\alpha)}$. Define the homomorphism $\lambda_1 : G_1^E \to G_1^E$ by extending

$$\lambda_1(x_i^{(1)}x_j^{(1)}) = r_{3,1}\left(x_i^{(1)}\right)\varphi_1\left(\gamma_1\left(x_i^{(1)}\right)\right)\left(r_{3,1}\left(x_j^{(1)}\right)\varphi_1\left(\gamma_1\left(x_j^{(1)}\right)\right)\right)^{-1} \tag{4.2.120}$$

as in Lemma 4.2.5.

Then we claim

$$\lambda_1(\varphi_1(h)) = \varphi_1\left(f_1\left(\varphi_3\left(h\right)\right)\right). \tag{4.2.121}$$

From the proof of Property A2 (see Equations (4.2.55) and (4.2.61)) we have

$$\varphi_1(\varphi_{3,1}^*(x_{c_i}^{(3)})) = r_{3,1}(x_{c_i}^{(3)}). \tag{4.2.122}$$

Then, as in the proof of Property A2, we check that Claim (4.2.121) holds on the generators

of H^E.

$$\varphi_1\left(f_1\left(\varphi_3\left(h_i h_j\right)\right)\right) = \varphi_1\left(f_1\left(\varphi_3\left(x_{a_i}^{(1)} x_{b_i}^{(2)} x_{c_i}^{(3)} x_{a_j}^{(1)} x_{b_j}^{(2)} x_{c_j}^{(3)} \sigma^{s_i+s_j}\right)\right)\right) \tag{4.2.123}$$

$$= \varphi_1\left(f_1\left(x_{c_i}^{(3)} x_{c_j}^{(3)}\right)\right) \tag{4.2.124}$$

$$= \varphi_1\left(\varphi_{3,1}^*\left(x_{c_i}^{(3)}\right)\gamma_1\left(x_{c_i}^{(3)}\right)\left(\varphi_{3,1}^*\left(x_{c_j}^{(3)}\right)\gamma_1\left(x_{c_j}^{(3)}\right)\right)^{-1}\right) \tag{4.2.125}$$

$$= r_{3,1}\left(x_{c_i}^{(3)}\right)\varphi_1\left(\gamma_1\left(x_{c_i}^{(3)}\right)\right)\left(r_{3,1}\left(x_{c_j}^{(3)}\right)\varphi_1\left(\gamma_1\left(x_{c_j}^{(3)}\right)\right)\right)^{-1} \tag{4.2.126}$$

$$= r_{3,1}\left(x_{a_i}^{(1)}\right)\varphi_1\left(\gamma_1\left(x_{a_i}^{(1)}\right)\right)\left(r_{3,1}\left(x_{a_j}^{(1)}\right)\varphi_1\left(\gamma_1\left(x_{a_j}^{(1)}\right)\right)\right)^{-1} \tag{4.2.127}$$

$$= \lambda_1\left(x_{a_i}^{(1)}\right)\lambda_1\left(x_{a_j}^{(1)}\right)^{-1} \tag{4.2.128}$$

$$= \lambda_1(\varphi_1(h_i h_j)) \tag{4.2.129}$$

Where, on line 4.2.127, we used the fact that $x_{a_i}^{(1)}$ and $x_{c_i}^{(3)}$ are both contained in the clause h_i, so the vertices corresponding to $x_{a_i}^{(1)}$ and $x_{c_i}^{(3)}$ are in the same connected component of G_{13} and consequently,

$$r_{3,1}\left(x_{c_i}^{(3)}\right) = r_{3,1}\left(x_{a_i}^{(1)}\right) \tag{4.2.130}$$

and

$$\gamma_1\left(x_{c_i}^{(3)}\right) = \gamma_1\left(x_{a_i}^{(1)}\right). \tag{4.2.131}$$

Since $\lambda_1, \varphi_1, f_1$, and φ_3 are all homomorphisms, this proves the claim.

Then for any $v \in G_3$ satisfying $\varphi_1\left(\varphi_{3,1}^*(v)\right) = 1$ we also have $v = \varphi_3\left(\varphi_{3,1}^*(v)\right)$ and

$$\varphi_1\left(f_1(v)\right) = \varphi_1\left(f_1\left(\varphi_3\left(\varphi_{3,1}^*\left(v\right)\right)\right)\right) \tag{4.2.132}$$

$$= \lambda_1\left(\varphi_1\left(\varphi_{3,1}^*\left(v\right)\right)\right) \tag{4.2.133}$$

$$= \lambda_1(1) = 1 \tag{4.2.134}$$

which proves property B1.

Property B4 follows from Property C1 of the words $\gamma_1(x_i^{(3)})$ and Property A2 of the map

$\varphi^*_{3,2}$. Property C1 gives

$$\varphi_2\left(\gamma_1(x_i^{(3)})\right) = 1 \tag{4.2.135}$$

and then Property A2 gives

$$\varphi_2\left(\varphi^*_{3,2}\left(\varphi_3\left(\gamma_1(x_i^{(3)})\right)\right)\right) = 1. \tag{4.2.136}$$

Thus, the gadgets words inserted by the map f_1 map to the identity under $\varphi_2\left(\varphi^*_{3,2}\left(\varphi_3\right)\right)$ and Property B4 follows. In math, we verify Property B4 by checking the action of the two maps on the generators of G_3^E:

$$\begin{aligned}
&\varphi_2\left(\varphi^*_{3,2}\left(\varphi_3\left(f_1(x_i^{(3)}x_j^{(3)})\right)\right)\right) && (4.2.137)\\
&\quad= \varphi_2\left(\varphi^*_{3,2}\left(\varphi_3\left[\varphi^*_{3,1}\left(x_i^{(3)}\right)\gamma_1\left(x_i^{(3)}\right)\gamma_1\left(x_j^{(3)}\right)^{-1}\varphi^*_{3,1}\left(x_j^{(3)}\right)\right]\right)\right) && (4.2.138)\\
&\quad= \varphi_2\left(\varphi^*_{3,2}\left(\varphi_3\left[\varphi^*_{3,1}\left(x_i^{(3)}\right)\varphi^*_{3,1}\left(x_j^{(3)}\right)\right]\right)\right) && (4.2.139)\\
&\quad= \varphi_2\left(\varphi^*_{3,2}\left(x_i^{(3)}x_j^{(3)}\right)\right). && (4.2.140)
\end{aligned}$$

Where we used Equation (4.2.136) to cancel the two γ_1 terms in line (4.2.138).

Finally, Property B3 follows from Property C2 of the words $\gamma_1(x_i^{(3)})$. If v has even length, we can write

$$v = \prod_i x_{o_i}^{(3)}x_{e_i}^{(3)}. \tag{4.2.141}$$

Then

$$f_1(v) = \prod_i \varphi^*_{3,1}\left(x_{o_i}^{(3)}\right)\gamma_1(x_{o_i}^{(3)})\gamma_1(x_{e_i}^{(3)})^{-1}\varphi^*_{3,1}\left(x_{e_i}^{(3)}\right) \tag{4.2.142}$$

By Property C2 of the words $\gamma_1(x_i^{(3)})$

$$\varphi_1\left(\varphi_{3,1}^*\left(\varphi_3\left(f_1(v)\right)\right)\right) \tag{4.2.143}$$

$$= \prod_i \varphi_1\left(\varphi_{3,1}^*\left(\varphi_3\left(\varphi_{3,1}^*\left(x_{o_i}^{(3)}\right)\right)\right) \cdot \varphi_{3,1}^*\left(\varphi_3\left(\gamma_1(x_{o_i}^{(3)})\right)\right) \cdot \varphi_{3,1}^*\left(\varphi_3\left(\gamma_1(x_{e_i}^{(3)})^{-1}\right)\right) \cdot \varphi_{3,1}^*\left(\varphi_3\left(\varphi_{3,1}^*\left(x_{e_i}^{(3)}\right)\right)\right)\right) \tag{4.2.144}$$

$$= \prod_i \varphi_1\left(\varphi_{3,1}^*\left(x_{o_i}^{(3)}\right) \cdot \varphi_{3,1}^*\left(x_{o_i}^{(3)} x_1^{(3)}\right) \cdot \varphi_{3,1}^*\left(\left(x_{e_i}^{(3)} x_1^{(3)}\right)^{-1}\right) \cdot \varphi_{3,1}^*\left(x_{e_i}^{(3)}\right)\right) \tag{4.2.145}$$

$$= \prod_i \varphi_1\left(\varphi_{3,1}^*\left(x_{o_i}^{(3)} x_{o_i}^{(3)} x_1^{(3)} x_1^{(3)} x_{e_i}^{(3)} x_{e_i}^{(3)}\right)\right) \tag{4.2.146}$$

$$= 1 \tag{4.2.147}$$

□

One nice property of the maps f_1, f_2 is that they map words inside the K subgroup to words inside the K subgroup. We show that in the following lemma.

Lemma 4.2.12. *For any $v \in K \cap G_3^E$ we have*

$$f_1(v), f_2(v) \in K. \tag{4.2.148}$$

Proof. By assumption, we can write

$$v = \prod_i u_i \left[x_{a_{i_1}}^{(3)} x_{a_{i_2}}^{(3)}, x_{a_{i_3}}^{(3)} x_{a_{i_4}}^{(3)}\right] u_i^{-1}. \tag{4.2.149}$$

Then,

$$f_1(v) = \prod_i f_1\left(u_i\right) f_1\left(\left[x_{a_{i_1}}^{(3)} x_{a_{i_2}}^{(3)}, x_{a_{i_3}}^{(3)} x_{a_{i_4}}^{(3)}\right]\right) f_1\left(u_i^{-1}\right) \tag{4.2.150}$$

$$= \prod_i f_1\left(u_i\right) \left[f_1\left(x_{a_{i_1}}^{(3)} x_{a_{i_2}}^{(3)}\right), f_1\left(x_{a_{i_3}}^{(3)} x_{a_{i_4}}^{(3)}\right)\right] f_1\left(u_i^{-1}\right) \tag{4.2.151}$$

We have $f_1\left(x_{a_{i_1}}^{(3)}x_{a_{i_2}}^{(3)}\right), f_1\left(x_{a_{i_3}}^{(3)}x_{a_{i_4}}^{(3)}\right) \in G^E$, so (by Lemma 4.3.1)

$$\left[f_1\left(x_{a_{i_1}}^{(3)}x_{a_{i_2}}^{(3)}\right), f_1\left(x_{a_{i_3}}^{(3)}x_{a_{i_4}}^{(3)}\right)\right] \in K. \tag{4.2.152}$$

But K is normal, so we also have

$$f_1(u_i)\left[f_1\left(x_{a_{i_1}}^{(3)}x_{a_{i_2}}^{(3)}\right), f_1\left(x_{a_{i_3}}^{(3)}x_{a_{i_4}}^{(3)}\right)\right] f_1\left(u_i^{-1}\right) \in K \tag{4.2.153}$$

for all i, hence

$$\prod_i f_1(u_i)\left[f_1\left(x_{a_{i_1}}^{(3)}x_{a_{i_2}}^{(3)}\right), f_1\left(x_{a_{i_3}}^{(3)}x_{a_{i_4}}^{(3)}\right)\right] f_1\left(u_i^{-1}\right) = f_1(v) \in K. \tag{4.2.154}$$

The proof for f_2 is identical. □

As a corollary, we note that the maps f_1, f_2 don't introduce any undesired factors of σ.

Corollary 4.2.13. *For any word $v \in K \cap G_3^E$, we have*

$$\varphi_\sigma(f_1(v)) = \varphi_\sigma(f_2(v)) = 1 \tag{4.2.155}$$

Proof. Similarly to the proof of Corollary 4.2.8, note that $f_1(v) \in K$ by Lemma 4.2.12, so $\varphi_\sigma(f_1(v)) = 1$ by Lemma 4.3.4. The proof for f_2 is similar. □

Final Proof

Finally, we are ready to prove Theorem 4.1.6.

Proof (Theorem 4.1.6). It is immediate that

$$\sigma \in H \implies [\sigma]_K \in H \pmod K. \tag{4.2.156}$$

To see the reverse direction, assume that $[\sigma]_K \in H \pmod K$. Then there exists some $w \in H$ satisfying $w = \sigma \pmod K$. By Lemma 4.2.9, there exists a word $w' \in H$ satisfying $\varphi_1(w') = \varphi_2(w') = 1$ and $w' = \sigma \pmod K$. Note that the last condition implies that $w' = \sigma k$

for some $k \in K$, hence

$$\varphi_3(w') = \varphi_3(\sigma k) = k \in K \cap G_3^E. \tag{4.2.157}$$

We choose words $u_i \in G_3^E$ and indices $a_{i_1}, ..., a_{i_4} \in [N]$ so that

$$\varphi_3(w') = \prod_i u_i \left[x_{a_{i_1}}^{(3)} x_{a_{i_2}}^{(3)}, x_{a_{i_3}}^{(3)} x_{a_{i_4}}^{(3)} \right] u_i^{-1}. \tag{4.2.158}$$

Now we insert gadgets into the word w'. Consider the word

$$w'' = w' \varphi_{3,1}^* \left(\varphi_3(w')\right)^{-1} f_1(\varphi_3\left(w'\right)) \tag{4.2.159}$$

Note that $\varphi_1(w') = 1$, and $w' \in H$. Hence $\varphi_1\left(\varphi_{3,1}^*\left(\varphi_3(w')\right)\right) = 1$ by Property A2 of the map $\varphi_{3,1}^*$. By Property B1 of f_1, we also have $\varphi_1\left(f_1\left(\varphi_3(w')\right)\right) = 1$. Putting this all together,

$$\varphi_1\left(w' \varphi_{3,1}^*\left(\varphi_3(w')\right)^{-1} f_1(\varphi_3\left(w'\right))\right) = \varphi_1\left(w'\right) \varphi_1\left(\varphi_{3,1}^*\left(\varphi_3(w')\right)^{-1}\right) \varphi_1\left(f_1(\varphi_3\left(w'\right))\right) = 1. \tag{4.2.160}$$

By Property B2 of the map f_1 we have

$$\begin{aligned} \varphi_2\left(w' \varphi_{3,1}^*\left(\varphi_3(w')\right)^{-1} f_1(\varphi_3\left(w'\right))\right) &= \varphi_2\left(w'\right) \varphi_2\left(\varphi_{3,1}^*\left(\varphi_3(w')\right)^{-1}\right) \varphi_2\left(f_1(\varphi_3\left(w'\right))\right) && (4.2.161) \\ &= \varphi_2\left(\varphi_{3,1}^*\left(\varphi_3(w')\right)^{-1}\right) \varphi_2\left(\varphi_{3,1}^*(\varphi_3\left(w'\right))\right) && (4.2.162) \\ &= 1 && (4.2.163) \end{aligned}$$

Finally

$$\varphi_3\left(w' \varphi_{3,1}^*\left(\varphi_3(w')\right)^{-1} f_1(\varphi_3\left(w'\right))\right) = \varphi_3\left(f_1(\varphi_3\left(w'\right))\right) \tag{4.2.164}$$

by Property A1 of the map $\varphi_{3,1}^*$. Also note that $\varphi_3\left(f_1(\varphi_3\left(w'\right))\right) \in K$ by Lemma 4.2.12 and the fact that φ_3 maps words in K to words in K (Lemma 4.3.5).

We summarize:

$$\varphi_1(w'') = \varphi_2(w'') = 1, \tag{4.2.165}$$

and

$$\varphi_3(w'') = \varphi_3\left(f_1(\varphi_3\left(w'\right))\right) \in K. \tag{4.2.166}$$

Now we again add gadgets to w'' with the 1 and 2 indices swapped. Recall

$$w'' = w'\varphi_{3,1}^*\left(\varphi_3(w')\right)^{-1} f_1(\varphi_3\left(w'\right)), \tag{4.2.167}$$

then define

$$w''' = w''\varphi_{3,2}^*\left(\varphi_3(w'')\right)^{-1} f_2(\varphi_3\left(w''\right)) \tag{4.2.168}$$

The same arguments as above give

$$\varphi_1(w''') = \varphi_2(w''') = 1 \tag{4.2.169}$$

and

$$\varphi_3(w''') = \varphi_3\left(f_2\left(\varphi_3\left(w''\right)\right)\right) \tag{4.2.170}$$

$$= \varphi_3\left(f_2\left(\varphi_3\left(f_1(\varphi_3\left(w'\right))\right)\right)\right) \in K. \tag{4.2.171}$$

We have, by assumption,

$$\varphi_3(w') = \prod_i u_i \left[x_{a_{i_1}}^{(3)} x_{a_{i_2}}^{(3)}, x_{a_{i_3}}^{(3)} x_{a_{i_4}}^{(3)}\right] u_i^{-1}. \tag{4.2.172}$$

We define a composition of maps $F : G_3 \to G_3$

$$F := \varphi_3 \circ f_2 \circ \varphi_3 \circ f_1. \tag{4.2.173}$$

Then we have

$$\varphi_3(w''') = F\left(\prod_i u_i \left[x^{(3)}_{a_{i_1}} x^{(3)}_{a_{i_2}}, x^{(3)}_{a_{i_3}} x^{(3)}_{a_{i_4}}\right] u_i^{-1}\right) \tag{4.2.174}$$

$$= \prod_i F\left(u_i \left[x^{(3)}_{a_{i_1}} x^{(3)}_{a_{i_2}}, x^{(3)}_{a_{i_3}} x^{(3)}_{a_{i_4}}\right] u_i^{-1}\right) \tag{4.2.175}$$

$$= \prod_i F(u_i)\left[F\left(x^{(3)}_{a_{i_1}} x^{(3)}_{a_{i_2}}\right), F\left(x^{(3)}_{a_{i_3}} x^{(3)}_{a_{i_4}}\right)\right] F\left(u_i^{-1}\right). \tag{4.2.176}$$

where we used the fact that each word $u_i \left[x^{(3)}_{a_{i_1}} x^{(3)}_{a_{i_2}}, x^{(3)}_{a_{i_3}} x^{(3)}_{a_{i_4}}\right] u_i^{-1}$ has even length on the first line, and that each word u_i has even length on the second.

Now

$$\varphi_2\left(\varphi^*_{3,2}\left(F\left(x^{(3)}_j x^{(3)}_k\right)\right)\right) = \varphi_2\left(\varphi^*_{3,2}\left(\varphi_3\left(f_2\left(\varphi_3\left(f_1\left(x^{(3)}_j x^{(3)}_k\right)\right)\right)\right)\right)\right) \tag{4.2.177}$$

$$= 1 \tag{4.2.178}$$

by Property B3 and

$$\varphi_1\left(\varphi^*_{3,1}\left(F\left(x^{(3)}_j x^{(3)}_k\right)\right)\right) = \varphi_1\left(\varphi^*_{3,1}\left(\varphi_3\left(f_2\left(\varphi_3\left(f_1\left(x^{(3)}_j x^{(3)}_k\right)\right)\right)\right)\right)\right) \tag{4.2.179}$$

$$= \varphi_1\left(\varphi^*_{3,1}\left(\varphi_3\left(f_1\left(x^{(3)}_j x^{(3)}_k\right)\right)\right)\right) \tag{4.2.180}$$

$$= 1 \tag{4.2.181}$$

where we used Property B4 and then Property B3 of the maps f_2 and f_1.

Finally, consider the word

$$w'''' = \prod_i \varphi^*_3(F(u_i))\left[\varphi^*_{3,1}\left(F\left(x^{(3)}_{a_{i_1}} x^{(3)}_{a_{i_2}}\right)\right), \varphi^*_{3,2}\left(F\left(x^{(3)}_{a_{i_3}} x^{(3)}_{a_{i_4}}\right)\right)\right] \varphi^*_3\left(F\left(u_i^{-1}\right)\right). \tag{4.2.182}$$

We have

$$\varphi_3(w'''') = \prod_i F(u_i)\left[F\left(x^{(3)}_{a_{i_1}} x^{(3)}_{a_{i_2}}\right), F\left(x^{(3)}_{a_{i_3}} x^{(3)}_{a_{i_4}}\right)\right] F\left(u_i^{-1}\right) \tag{4.2.183}$$

$$= \varphi_3(w''') \tag{4.2.184}$$

Equation (4.2.178) gives

$$\varphi_2(w'''') \tag{4.2.185}$$

$$= \prod_i \varphi_2\left(\varphi_3^*\left(F\left(u_i\right)\right)\right)\left[\varphi_2\left(\varphi_{3,1}^*\left(F\left(x_{a_{i_1}}^{(3)}x_{a_{i_2}}^{(3)}\right)\right)\right), \varphi_2\left(\varphi_{3,2}^*\left(F\left(x_{a_{i_3}}^{(3)}x_{a_{i_4}}^{(3)}\right)\right)\right)\right]\varphi_2\left(\varphi_3^*\left(F\left(u_i^{-1}\right)\right)\right) \tag{4.2.186}$$

$$= \prod_i \varphi_2\left(\varphi_3^*\left(F\left(u_i\right)\right)\right)\left[\varphi_2\left(\varphi_{3,1}^*\left(F\left(x_{a_{i_1}}^{(3)}x_{a_{i_2}}^{(3)}\right)\right)\right), 1\right]\varphi_2\left(\varphi_3^*\left(F\left(u_i\right)\right)\right)^{-1} \tag{4.2.187}$$

$$= 1 \tag{4.2.188}$$

A similar argument using Equation (4.2.181) shows $\varphi_1(w'''') = 1$. Finally, noting that elements in the image of φ_σ commute with each other (an argument similar to the proof of Corollary 4.2.13) shows

$$\varphi_\sigma\left(w''''\right) \tag{4.2.189}$$

$$= \prod_i \varphi_\sigma\left(\varphi_3^*\left(F\left(u_i\right)\right)\right)\varphi_\sigma\left(\left[\varphi_{3,1}^*\left(F\left(x_{a_{i_1}}^{(3)}x_{a_{i_2}}^{(3)}\right)\right), \varphi_{3,2}^*\left(F\left(x_{a_{i_3}}^{(3)}x_{a_{i_4}}^{(3)}\right)\right)\right]\right)\varphi_\sigma\left(\varphi_3^*\left(F\left(u_i^{-1}\right)\right)\right) \tag{4.2.190}$$

$$= \prod_i \varphi_\sigma\left(\varphi_3^*\left(F\left(u_i\right)\right)\right)\left[\varphi_\sigma\left(\varphi_{3,1}^*\left(F\left(x_{a_{i_1}}^{(3)}x_{a_{i_2}}^{(3)}\right)\right)\right), \varphi_\sigma\left(\varphi_{3,2}^*\left(F\left(x_{a_{i_3}}^{(3)}x_{a_{i_4}}^{(3)}\right)\right)\right)\right]\varphi_\sigma\left(\varphi_3^*\left(F\left(u_i^{-1}\right)\right)\right) \tag{4.2.191}$$

$$= \prod_i \varphi_\sigma\left(\varphi_3^*\left(F\left(u_i\right)\right)\right)\varphi_\sigma\left(\varphi_3^*\left(F\left(u_i^{-1}\right)\right)\right) \tag{4.2.192}$$

$$= 1. \tag{4.2.193}$$

To put this all together and complete the proof, consider the word $w'''w''''^{-1}$. Using equations Eqs. (4.2.169) and (4.2.188)

$$\varphi_2(w'''w''''^{-1}) = \varphi_2(w''')\varphi_2(w''''^{-1}) \tag{4.2.194}$$

$$= 1 \tag{4.2.195}$$

with a similar argument giving

$$\varphi_1(w'''w''''^{-1}) = \varphi_1(w''')\varphi_1(w'''')^{-1} \tag{4.2.196}$$
$$= 1. \tag{4.2.197}$$

Equation (4.2.184) gives

$$\varphi_3(w'''w''''^{-1}) = \varphi_3(w''')\varphi_3(w'''')^{-1} \tag{4.2.198}$$
$$= \varphi_3(w''')\varphi_3(w''')^{-1} \tag{4.2.199}$$
$$= 1. \tag{4.2.200}$$

Finally, Equation (4.2.193), Corollary 4.2.13, and Corollary 4.2.8 give

$$\begin{aligned}
\varphi_\sigma(w'''w''''^{-1}) &= \varphi_\sigma(w''') \\
&= \varphi_\sigma(w''\varphi_{3,2}^*\left(\varphi_3(w'')\right)^{-1} f_2(\varphi_3\left(w''\right))) \\
&= \varphi_\sigma(w'') \\
&= \varphi_\sigma\left(w'\varphi_{3,1}^*\left(\varphi_3(w')\right)^{-1} f_1(\varphi_3\left(w'\right))\right) \\
&= \varphi_\sigma\left(w'\right) \\
&= \sigma.
\end{aligned}$$

The proof is complete. □

4.3 Properties of K and its Interactions

Here we prove several small facts used in the proof of Theorem 4.1.6 as well as some which add perspective on K.

4.3.1 Properties of K

Lemma 4.3.1. *Let u, v be two even length words in G_α. Then $[u, v] \in K$.*

Proof. Let $l(u)$ denote the length of u, with $l(v)$ defined similarly. Define $L = l(u) + l(v)$. We prove by induction on L.

When $L = 4$, u and v must both have length 2, hence $[u, v]$ is a generator of K. Then the result is immediate.

Otherwise, we must have that either $l(u)$ or $l(v)$ is greater than 2. For now we assume $l(v) > 2$. Then we can write

$$v = v'v'' \tag{4.3.1}$$

with v' and v'' both even length words. Note that

$$l(v') + l(v'') = l(v) \tag{4.3.2}$$

so v' and v'' both have length less than v. Then we can write

$$[u, v] = [u, v'v''] \tag{4.3.3}$$

$$= [u, v']\, v'^{-1}\, [u, v'']\, v' \tag{4.3.4}$$

where we have used the commutator identity

$$[x, yz] = [x, y]\, y^{-1}\, [x, z]\, y \tag{4.3.5}$$

on the second line. $l(u) + l(v')$ and $l(u) + l(v'')$ are both less than L, so by the induction hypothesis we have $[u, v']$ and $[u, v'']$ are both in K. Since K is normal, that also implies

$$v'^{-1}\, [u, v'']\, v' \in K, \tag{4.3.6}$$

and since K is a group

$$[x, y]\, y^{-1}\, [x, z]\, y \in K. \tag{4.3.7}$$

The proof when $l(u) > 2$ is almost identical, except we use the commutator identity

$$[xy, z] = y^{-1}\,[x, z]\,y\,[y, z] \tag{4.3.8}$$

□

Canonical form for monomials mod K

Consider the game group G is defined for k players and let $\sim_K$ denote the equivalence relation on G defined by modding out by K. In this subsection we shall write down a canonical selection from the equivalence classes. This is not used in the proofs here, but might be in other proofs and it is certainly useful in computer experiments. While G is defined for k players modding out by K acts independently on the variables $x_j^{(\alpha)}$ $j = 1, \ldots, n$ associated with each player α. Thus wlog we can take $k = 1$. Also G contains σ but we shall ignore it, since σ has no impact on the canonical form.

The core observation is the following lemma.

Lemma 4.3.2. *Suppose G is the game group of a 1-XOR quantum qame. Monomials of the form*

$$wabcdq \quad and \quad wcbadq \quad and \quad wadcbq$$

are all equal mod K. Here a, b, c, d are generators of the G and w and q are arbitrary monomials.

For degree 3 or more monomials this immediately implies that interchanging any two even position variables or any two odd position variables in a monomial m produces a monomial $\tilde{m}$ with $m \sim_K \tilde{m}$.

Proof. We first show $abcd \sim_K adcb$ by noting

$$(adcb)^{-1}\,abcd = bcd\;bcd = bc\;dc\;cb\;cd \sim_K 1. \tag{4.3.9}$$

where the last equation is true by definition of K. The proof that the first and third monomials are equivalent goes similarly.

If m has degree 3 write it as abc, then the property just proved for degree 4 gives

$$abc \sim_K abcxx \;\Leftrightarrow\; cbaxx \sim_K cba \tag{4.3.10}$$

as claimed. □

Given an ordering on the generators of G, a canonical form of a monomial m is seen easily from the lemma. We describe it in terms of an algorithm.

Algorithm K, Q

1. Find its even (resp, odd) part, namely the monomial whose entries are the variables in the even (resp odd) locations of m. For example: take $m = zgabcdfzz$, then

   $$even[m] := gbdfz \qquad odd[m] := zacez$$

2. Select a variable, say v, and count how many times, e, it appears in $even[m]$ and o times in $odd[m]$.

   If $o \leq e$, then remove all variables v from the list $odd[m]$ and also remove o of the v's from $even[m]$. If $e \leq o$, then remove all v from the list $even[m]$ and also remove e of the v's from $odd[m]$. The order of removal does not matter. Do this for all variables (not just v) to get $evQ[m]$ and $oddQ[m]$.

   Example revisited: take G to have generators equal to the alphabet $a, \ldots, z$ with each generator having square equal to 1. $e = 1$ and $o = 2$ for the variable z. So $evQ[m] = gbdf$ and $oddQ[m] = zace$.

3. Order both lists. $alph[even] := bdfg$, $alph[odd] := acez$

4. Recombine these words to make one word. $canon[m] := abcdefzg$ □

Application of Lemma 4.3.2 proves the Algorithm succeeds as is formalized by the following.

Proposition 4.3.3. *For monomials of degree ≥ 3, we have that $canon[m]$ is uniquely determined and $m \sim_K canon[m]$. That is, $canon[m]$ is a canonical form for m.*

4.3.2 The interaction of φ_σ and φ_α with K

Lemma 4.3.4. *For any $k \in K$,*

$$\varphi_\sigma(k) = 1. \tag{4.3.11}$$

Proof. We can write

$$k = \prod_i u_i \left[x_{a_i}^{\alpha_i} x_{b_i}^{\alpha_i}, x_{c_i}^{\alpha_i} x_{d_i}^{\alpha_i}\right] u_i^{-1} \tag{4.3.12}$$

Then,

$$\varphi_\sigma(k) = \varphi_\sigma\left(\prod_i u_i \left[x_{a_i}^{\alpha_i} x_{b_i}^{\alpha_i}, x_{c_i}^{\alpha_i} x_{d_i}^{\alpha_i}\right] u_i^{-1}\right) \tag{4.3.13}$$

$$= \prod_i \varphi_\sigma(u_i) \left[\varphi_\sigma\left(x_{a_i}^{\alpha_i} x_{b_i}^{\alpha_i}\right), \varphi_\sigma\left(x_{c_i}^{\alpha_i} x_{d_i}^{\alpha_i}\right)\right] \varphi_\sigma\left(u_i^{-1}\right) \tag{4.3.14}$$

$$= \prod_i \varphi_\sigma(u_i)\varphi_\sigma\left(u_i^{-1}\right) \tag{4.3.15}$$

$$= 1 \tag{4.3.16}$$

where we used that $\mathrm{Im}(\varphi_\sigma) = \{\sigma, 1\}$ is a commutative group to show the commutator terms were the identity. □

Lemma 4.3.5. *For any $k \in K$, and $\alpha \in \{1, 2, 3\}$:*

$$\varphi_\alpha(k) \in K. \tag{4.3.17}$$

Proof. Define the set C to be all commutators of pairs, that is

$$C = \left\{\left[x_i^\alpha x_j^\alpha, x_k^\alpha x_l^\alpha\right] : i, j, k, l \in [n], \alpha \in [3]\right\}. \tag{4.3.18}$$

Recall that K was defied to be the normal closure of C in G^E, that is:

$$K = \langle C \rangle^{G^E}. \tag{4.3.19}$$

We first show that

$$\varphi_\alpha(c) \in C \tag{4.3.20}$$

for all $c \in C$. To see this, note

$$\varphi_\alpha\left(\left[x_i^{(\beta)}x_j^{(\beta),}, x_k^{(\beta)}x_l^{(\beta)}\right]\right) = \left[x_i^{(\alpha)}x_j^{(\alpha),}, x_k^{(\alpha)}x_l^{(\alpha)}\right] \in K \tag{4.3.21}$$

for $\alpha = \beta$, and

$$\varphi_\alpha\left(\left[x_i^{(\beta)}x_j^{(\beta),}, x_k^{(\beta)}x_l^{(\beta)}\right]\right) = 1 \in K. \tag{4.3.22}$$

for $\alpha \neq \beta$.

Then, since φ_α is a homomorphism mapping $G^E \to G_3^E$, and $\varphi_\alpha(C) \subset C$, we have

$$\varphi_\alpha : \langle C \rangle^{G^E} \hookrightarrow \langle C \rangle^{G_3^E} \subset K. \tag{4.3.23}$$

The result follows. □

4.3.3 Equivalence between a PREF and $\sigma \in H \pmod K$

In [66] an object called a *parity refutation* was defined. A (paraphrased) version of that definition using the language of Section 4.1.1 is repeated here. First, we define a *parity preserving permutation*.

Definition 4.3.6. *A parity preserving permutation of a sequence of generators (written here as a product)*

$$x_{a_1}^{(1)}x_{a_2}^{(1)}...x_{a_{l_1}}^{(1)}x_{b_1}^{(2)}...x_{b_{l_2}}^{(2)}x_{c_1}^{(3)}...x_{c_{l_3}}^{(3)}\sigma^s \tag{4.3.24}$$

is a permutation P which satisfies

$$P(x_{a_i}^{(1)}) = x_{a_j}^{(1)} \tag{4.3.25}$$

with $i = j \pmod 2$*, similar restrictions for* $P(x^{(2)}_{b_{i'}})$ *and* $P(x^{(3)}_{c_{i''}})$ *and the condition* $P(\sigma) = \sigma$.

An equivalent definition of parity preserving permutations which will be useful to use later are permutations P which can be decomposed into products of transpositions of the form $\pi^{(\alpha)}_{j,j+2}$ with $\alpha \in [3]$ and

$$\pi^{(\alpha)}_{j,j+2}\left(x^{(\alpha)}_{a_1}x^{(\alpha)}_{a_2}...x^{(\alpha)}_{a_j}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_{j+2}}...x^{(\alpha)}_{a_l}\right) = x^{(\alpha)}_{a_1}x^{(\alpha)}_{a_2}...x^{(\alpha)}_{a_{j+2}}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_j}...x^{(\alpha)}_{a_l} \tag{4.3.26}$$

Parity preserving permutations can be used to define an equivalence relation on the words $g \in G$

Definition 4.3.7. *Two words* $g_1, g_2 \in G$ *are* parity permutation equivalent*, written* $g_1 \sim_p g_2$*, if there is a sequence of generators*

$$x^{(1)}_{a_1}x^{(1)}_{a_2}...x^{(1)}_{a_{l_1}}x^{(2)}_{b_1}...x^{(2)}_{b_{l_2}}x^{(3)}_{c_1}...x^{(3)}_{c_{l_3}}\sigma^s = g_1 \tag{4.3.27}$$

and a parity preserving permutation P *acting on that sequence of generators satisfying*

$$P(x^{(1)}_{a_1}x^{(1)}_{a_2}...x^{(1)}_{a_{l_1}}x^{(2)}_{b_1}...x^{(2)}_{b_{l_2}}x^{(3)}_{c_1}...x^{(3)}_{c_{l_3}}\sigma^s) = g_2 \tag{4.3.28}$$

Routine calculation (given in [66]) shows $\sim_p$ is an equivalence relation on elements of G. Finally, we define a *parity refutation* (PREF).

Definition 4.3.8. *A sequence of clauses* $h_{r_1}, h_{r_2}, ..., h_{r_l}$ *is called a* parity refutation *if* $h_{r_1}h_{r_2}...h_{r_l} \sim_p \sigma$.

Existence of a parity refutation is exactly equivalent to a word $\sigma \in H \pmod K$, as we show in the following theorem. (Actually, a stronger statement is true: the equivalence relation $\sim_p$ is exactly the same as the equivalence relation on G induced by modding out by K. Small modifications to the proof below give that result.)

Theorem 4.3.9. *A sequence of clauses* $h_{r_1}h_{r_2}...h_{r_l}$ *is a parity refutation iff the word* $h_{r_1}h_{r_2}...h_{r_l} \in$

H obtained by multiplying the clauses together satisfies

$$h_{r_1}h_{r_2}...h_{r_l} = \sigma \pmod{K} \tag{4.3.29}$$

Proof. Both directions of the proof are nontrivial. We first show that if a sequence of clauses $h_{r_1}h_{r_2}...h_{r_l}$ forms a parity refutation then $h_{r_1}h_{r_2}...h_{r_l} = \sigma \pmod{K}$. Recall that any parity preserving permutation P can be decomposed into transpositions of the form $\pi^{(\alpha)}_{j,j+2}$, where

$$\pi^{(\alpha)}_{j,j+2}\left(x^{(\alpha)}_{a_1}x^{(\alpha)}_{a_2}...x^{(\alpha)}_{a_j}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_{j+2}}...x^{(\alpha)}_{a_l}\right) = x^{(\alpha)}_{a_1}x^{(\alpha)}_{a_2}...x^{(\alpha)}_{a_{j+2}}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_j}...x^{(\alpha)}_{a_l} \tag{4.3.30}$$

But we also have

$$K \ni \left[x^{(\alpha)}_{a_{j+2}}x^{(\alpha)}_{a_{j+1}}, x^{(\alpha)}_{a_j}x^{(\alpha)}_{a_{j+1}}\right] = x^{(\alpha)}_{a_{j+2}}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_j}x^{(\alpha)}_{a_{j+2}}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_j} \tag{4.3.31}$$

hence

$$x^{(\alpha)}_{a_j}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_{j+2}} = x^{(\alpha)}_{a_j}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_{j+2}}x^{(\alpha)}_{a_{j+2}}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_j}x^{(\alpha)}_{a_{j+2}}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_j} \pmod{K} \tag{4.3.32}$$

$$= x^{(\alpha)}_{a_{j+2}}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_j} \pmod{K}. \tag{4.3.33}$$

As a consequence, we also have

$$x^{(\alpha)}_{a_1}x^{(\alpha)}_{a_2}...x^{(\alpha)}_{a_j}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_{j+2}}...x^{(\alpha)}_{a_l} = x^{(\alpha)}_{a_1}x^{(\alpha)}_{a_2}...x^{(\alpha)}_{a_{j+2}}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_j}...x^{(\alpha)}_{a_l} \pmod{K} \tag{4.3.34}$$

$$= \pi^{(\alpha)}_{j,j+2}\left(x^{(\alpha)}_{a_1}x^{(\alpha)}_{a_2}...x^{(\alpha)}_{a_j}x^{(\alpha)}_{a_{j+1}}x^{(\alpha)}_{a_{j+2}}...x^{(\alpha)}_{a_l}\right) \pmod{K}. \tag{4.3.35}$$

Since the word $x^{(\alpha)}_{a_1}x^{(\alpha)}_{a_2}...x^{(\alpha)}_{a_l}$ was arbitrary and we could decompose P into products of transpositions of the form $\pi^{(\alpha)}_{j,j+2}$ we conclude

$$h_{r_1}h_{r_2}...h_{r_l} = x^{(1)}_{a_{r_1}}x^{(1)}_{a_{r_2}}...x^{(3)}_{c_{r_l}}\sigma^{s_{r_1}+s_{r_2}+...s_{r_l}} \tag{4.3.36}$$

$$= P(x^{(1)}_{a_{r_1}}x^{(1)}_{a_{r_2}}...x^{(3)}_{c_{r_l}}\sigma^{s_{r_1}+s_{r_2}+...s_{r_l}}) \pmod{K} \tag{4.3.37}$$

$$= \sigma \pmod{K} \tag{4.3.38}$$

Where line (4.3.37) follows from equation (4.3.35) and line (4.3.38) follows from the definition of a parity refutation. This completes the proof in one direction.

It remains to show that if $h_{r_1}h_{r_2}...h_{r_l} = \sigma \pmod K$ we also have $h_{r_1}h_{r_2}...h_{r_l} \sim_p \sigma$. Our first step is to note that the equivalence relation $\sim_p$ respects multiplication by construction – that is we have $g_1 \sim_p g_2$ and $g_3 \sim_p g_4$ implies $g_1 g_2 \sim_p g_3 g_4$. We next note that for any set of generators $x_i^{(\alpha)}, x_j^{(\alpha)}, x_s^{(\alpha)}, x_t^{(\alpha)}$ and word $w \in G$ we have

$$w\left[x_i^{(\alpha)}x_j^{(\alpha)}, x_s^{(\alpha)}x_t^{(\alpha)}\right]w^{-1} = wx_i^{(\alpha)}x_j^{(\alpha)}x_s^{(\alpha)}x_t^{(\alpha)}\left(x_i^{(\alpha)}x_j^{(\alpha)}\right)^{-1}\left(x_s^{(\alpha)}x_t^{(\alpha)}\right)^{-1}w^{-1} \tag{4.3.39}$$

$$\sim_p ww^{-1}x_i^{(\alpha)}x_j^{(\alpha)}x_s^{(\alpha)}x_t^{(\alpha)}\left(x_i^{(\alpha)}x_j^{(\alpha)}\right)^{-1}\left(x_s^{(\alpha)}x_t^{(\alpha)}\right)^{-1} \tag{4.3.40}$$

$$\sim_p ww^{-1}x_i^{(\alpha)}x_j^{(\alpha)}x_s^{(\alpha)}x_t^{(\alpha)}\left(x_s^{(\alpha)}x_t^{(\alpha)}\right)^{-1}\left(x_i^{(\alpha)}x_j^{(\alpha)}\right)^{-1} = 1 \tag{4.3.41}$$

since the permutations moving w^{-1} to the other side of $\left[x_i^{(\alpha)}x_j^{(\alpha)}, x_s^{(\alpha)}x_t^{(\alpha)}\right]$ and swapping $\left(x_i^{(\alpha)}x_j^{(\alpha)}\right)^{-1}$ and $\left(x_s^{(\alpha)}x_t^{(\alpha)}\right)^{-1}$ are both parity preserving permutations. It follows that for any $k \in K$, $k \sim_p 1$. Then, if $h_{r_1}h_{r_2}...h_{r_l} = \sigma \pmod K$ we must also have $h_{r_1}h_{r_2}...h_{r_l}k = \sigma$ for some $k \in K$, and hence

$$h_{r_1}...h_{r_l} = h_{r_1}...h_{r_l}kk^{-1} \sim_p \sigma(1) = \sigma \tag{4.3.42}$$

where we used that $\sim_p$ respected multiplication $h_{r_1}...h_{r_l}k = \sigma$ and $k^{-1} \sim_p 1$ to obtain the equivalence. This completes the proof.

4.3.4 MERP as a mod K strategy

Recall from Definition 4.1.3 the MERP strategies are a nice class of finite dimensional strategies which generalize the GHZ strategy. Here we give a direct proof that MERP strategies are annihilated by the K relations.

Theorem 4.3.10. *The MERP strategy observables respect the mod K relations. That is,*

$$\left[X_i^{(\alpha)}X_{i'}^{(\alpha)}, X_j^{(\alpha)}X_{j'}^{(\alpha)}\right] = 1 \tag{4.3.43}$$

for all α, i, i', j, j' if the $X_i^{(\alpha)}$ are MERP strategy observables as defined above.

Proof. The proof is computational, with some tricks about Pauli matrices. Let all the $X_i^{(\alpha)}$ be MERP strategy observables and note, for all indices

$$\left[X_i^{(\alpha)}X_{i'}^{(\alpha)}, X_j^{(\alpha)}X_{j'}^{(\alpha)}\right] = I^{\otimes((\alpha-1)} \otimes \left[M(\theta_i^{(\alpha)})M(\theta_{i'}^{(\alpha)}), M(\theta_j^{(\alpha)}M(\theta_{j'})^{(\alpha)}))\right] \otimes I^{\otimes(k-\alpha)} \tag{4.3.44}$$

by the tensor product structure. Now, the Pauli matrices anti-commute, so

$$\sigma_x\sigma_z = -\sigma_z\sigma_x \tag{4.3.45}$$

and

$$\sigma_x \exp(i\theta\sigma_z) = \exp(-i\theta\sigma_z)\sigma_x \qquad \exp(i\theta\sigma_z)\sigma_x = \sigma_x \exp(-i\theta\sigma_z) \tag{4.3.46}$$

where the later equalities can be shown by the Taylor series expansion of $\exp(i\theta\sigma_z)$. This lets us write our MERP strategy observables in a slightly simplier form, since

$$M(\theta_i^{(\alpha)}) = \exp\left(i\theta_i^{(\alpha)}\sigma_z\right)\sigma_x \exp\left(-i\theta_i^{(\alpha)}\sigma_z\right) \tag{4.3.47}$$

$$= \exp\left(2i\theta_i^{(\alpha)}\sigma_z\right)\sigma_x \tag{4.3.48}$$

As a more more significant application of Equation (4.3.46) we can show MERP strategy observables switch the sign on $\theta_i^{(\alpha)}$ when they commute since

$$M(\theta_i^{(\alpha)})M(\theta_j^{(\alpha)}) = \exp\left(2i\theta_i^{(\alpha)}\sigma_z\right)\sigma_x \exp\left(2i\theta_j^{(\alpha)}\sigma_z\right)\sigma_x \tag{4.3.49}$$

$$= \sigma_x \exp\left(-2i\theta_i^{(\alpha)}\sigma_z\right) \exp\left(2i\theta_j^{(\alpha)}\sigma_z\right)\sigma_x \tag{4.3.50}$$

$$= \sigma_x \exp\left(2i\theta_j^{(\alpha)}\sigma_z\right) \exp\left(-2i\theta_i^{(\alpha)}\sigma_z\right)\sigma_x \tag{4.3.51}$$

$$= \exp\left(-2i\theta_j^{(\alpha)}\sigma_z\right)\sigma_x \exp\left(-2i\theta_i^{(\alpha)}\sigma_z\right)\sigma_x \tag{4.3.52}$$

$$= M(-\theta_j^{(\alpha)})M(-\theta_i^{(\alpha)}) \tag{4.3.53}$$

using Equation (4.3.46) on the second line. Now, repeatedly applying Equation (4.3.53) gives

$$M(\theta_i^{(\alpha)})M(\theta_{i'}^{(\alpha)})M(\theta_j^{(\alpha)})M(\theta_{j'}^{(\alpha)}) = M(\theta_j^{(\alpha)})M(-\theta_i^{(\alpha)})M(-\theta_{i'}^{(\alpha)})M(\theta_{j'}^{(\alpha)}) \tag{4.3.54}$$

$$= M(\theta_j^{(\alpha)})M(\theta_{j'}^{(\alpha)})M(\theta_i^{(\alpha)})M(\theta_{i'}^{(\alpha)}) \tag{4.3.55}$$

Hence

$$\left[M(\theta_i^{(\alpha)})M(\theta_{i'}^{(\alpha)}), M(\theta_j^{(\alpha)})M(\theta_{j'}^{(\alpha)})\right] = 1 \tag{4.3.56}$$

and the result follows. □

□

4.4 Subgroup Membership

Theorem 4.4.1. *The subgroup membership problem is solvable in polynomial time for any finitely generated abelian group.*[16]

Proof. It reduces to linear algebra over the integers. We can write all the relations in the group G and generators of the subgroup $\tilde{G}$ as products of generators of G, raised to some power. When we multiply generators or apply a relation we just add or subtract the multiplicities of the relevant generators. So the subgroup membership problem just asks if a given vector (corresponding to the group element) is in the span of the vectors corresponding to the relations and subgroup generators. □

4.5 Chapter Summary

This chapter completely characterizes 3XOR games with perfect commuting operator strategies. First, an alternate view of PREFs as "the subgroup membership problem mod K" is developed (Theorem 4.1.2, with equivalence to the PREF condition discussed in Section 4.3.3).

[16] Stronger versions of this statement are also true. In particular, the subgroup membership problem is solvable for any finitely generated metabelian group[55] (meaning commutators of commutators vanish) or finitely generated nilpotent group[42].

Then, in the most involved algebraic argument of this thesis the "subgroup membership mod K" problem is shown to be necessary and sufficient for 3 player XOR games (Sketch following Theorem 4.1.6, full proof in Section 4.2.4). Combining this result with the MERP-PREF duality discussed in Chapter 3 shows MERP strategies are optimal for perfect 3 player XOR games.

Chapter 5

Specific Families of Games and Random Games

In this chapter we use the techniques developed in previous chapters (particularly Chapter 3) to construct families of games with interesting properties and to study randomly generated XOR games. The first family of games we construct, Capped GHZ (), is a family where ncSoS takes $\exp(n)$ levels and $\exp(\exp(n))$ time to detect that $\omega^* < 1$ (Theorem 5.1.2), in contrast to our algorithm which runs in polynomial time. The second, Asymptotically Perfect Difference (APD), is an explicit, deterministic family of k-XOR games with $\omega^* = 1$ and classical value $\omega \to 1/2$ in the limit of large k (Theorem 5.1.3).

For random instances of games, we show the existence of an unsatisfiable (i.e. $\omega^* < 1$) phase as in the classical case (Theorem 5.1.4). We also relate our methods to the ncSoS hierarchy. For random instances, we show that in the unsatisfiable phase, a superlinear number of levels of ncSoS is necessary to certify that $\omega^* < 1$ (Theorem 5.1.5).

This chapter uses the notation developed in Chapter 3 to describe and analyze XOR games.

5.1 Results

Theorem 5.1.1. *There exists a 6-player XOR game G with alphabet size 3 and 6 clauses, for which $\omega^*(G) = 1$ but the algorithm of Theorem 3.2.1 cannot detect this.*

Proof. Section 5.2.1. □

Theorem 5.1.2. *There exists a family of 3-XOR games with $\omega^* < 1$ but for which the minimum refutation length scales exponentially in the number of clauses m and alphabet size n. For these games exponentially many levels of ncSoS are needed to witness that $\omega^* < 1$.*

Proof. Section 5.2.2. □

Theorem 5.1.3. *There exists a family of k-XOR games, parametrized by K, for which $\omega^*(G(K)) = 1$ and the classical value is bounded by*

$$\frac{1}{2} \leq \omega(G(K)) \leq \frac{1}{2} + \sqrt{\frac{K+1}{2^{K+1}}} \leq \frac{1}{2} + \sqrt{\frac{\log k}{k}}. \tag{5.1.1}$$

Proof. Section 5.2.3. □

Theorem 5.1.4. *For every k, there exists a constant C_k^{unsat} depending only on k such that a random k-XOR game G with $m \geq C_k^{unsat} n$ clauses has value $\omega^*(G) < 1$ with probability $1 - o(1)$.*

Proof. Section 5.3.2. □

Theorem 5.1.5. *For any constant C, the minimum length refutation of a random 3-XOR game with $m = Cn$ queries on an alphabet of size n has length at least*

$$\frac{en\log(n)}{8C^2\log(\log(n))} - o\left(\frac{n\log(n)}{\log(\log(n))}\right) \tag{5.1.2}$$

with probability $1 - o(1)$ (as $n \to \infty$). Hence, either $\omega^* = 1$ or $\Omega(n\log(n)/\log(\log(n)))$ levels of the ncSoS hierarchy are needed to witness that $\omega^* < 1$ for such games.

Note that we can choose $C \geq C_3^{\text{unsat}}$ (with C_3^{unsat} from Theorem 5.1.4) such that for large enough n, typical random instances will have $\omega^* < 1$ but ncSoS will require $\Omega(n\log(n)/\log(\log(n)))$ levels to detect this.

Proof. Section 5.3.3. □

5.2 Specific Games

In this section we use the machinery of the previous sections to construct some games with interesting properties.

The first is a simple game, the 123 Game, which illustrates conditions under which the PREF condition can be fooled. It is a relatively small (6 player, 6 query) non-symmetric game which does contain a PREF, but still does not contain any refutations. We show this by giving an explicit value 1 strategy for the 123 Game.

The second is a family of games, called Capped GHZ (), which are designed to be hard instances for the ncSoS algorithm. In particular, the game on n variables (denoted CG_n) is a symmetric game with value strictly less than 1, meaning the decision algorithm of Section 3.4.3 can show the game has value < 1 in poly time, but with a minimum refutation of length at least exponential in n. This shows a doubly exponential improvement in the runtime of our decision algorithm as compared to the ncSoS algorithm, and an exponential improvement over the previous best known ncSoS lower bounds for this problem [30][1]. This game construction is based primarily on the theorems of Section 3.4, which outline the relationship between refutations and ncSoS runtime, as well as our decision algorithm.

Finally, we construct a family of games with commuting operater value 1 and a low classical value. These games are called Asymptotically Perfect Difference (APD) games, and are parameterized by K. The classical value of the K-th APD game (APD_K) approaches $1/2$, which is the lowest possible, in the limit of large K. The existence of such a family was posed as an open question in [8]. The construction of these games is based primarily on the difference between the linear equations defining MERP value 1 and classical value 1, which is discussed in Section 3.5.3.

These games are summarized in the following table, with a full discussion of each in the subsequent sections.

[1] In fact, to our knowledge, our results are the first exponential degree lower bound for the ncSoS hierarchy applied to *any* problem.

Game	n	m	k	ω^*	ω	minimum refutation length
123 Game	3	6	6	$\mathbf{1}$	$5/6$	–
CG_n	n	$3n-1$	3	$< 1 - 1/\exp(n)$	$1 - 1/m$	$\mathbf{2^{n+1} - 2}$
APD_K	2	2^K	$2^K - 1$	$\mathbf{1}$	$\mathbf{1/2 + \sqrt{K/2^K}}$	–

Table 5.1: Overview of the games constructed in this section. Quantities of note are denoted in bold.

5.2.1 123 Game

We begin with a discussion of the intuition behind the 123 game, then follow with an explicit value 1 strategy. It is instructive to begin by analyzing the "Small 123 Game".

Definition 5.2.1. *Define the* ***Small 123 Game*** *to be the $k = 3$ player game with $n = 3$ and $m = 6$ clauses*

$$G_{123}^{small} := \left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \end{bmatrix}, \begin{bmatrix} 3 \\ 3 \\ 3 \\ -1 \end{bmatrix}, \begin{bmatrix} 2 \\ 3 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \\ 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 3 \\ 1 \\ 2 \\ 1 \end{bmatrix} \right\}. \tag{5.2.1}$$

In this form, it is clear the Small 123 Game has $\omega^*(G_{123}^{\text{small}}) < 1$, since placing its clauses in the order presented forms a refutation.

The game matrix A has a one-dimensional left nullspace (corresponding to the space of candidate PREF specifications z satisfying $A^T z = 0$):

$$z \propto \begin{bmatrix} 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix}^T. \tag{5.2.2}$$

Any odd multiples of this basis vector produce a PREF specification z.

We now add players to this game while preserving this PREF specification, until we exclude all refutations formed by permutations of a single copy of each of the clauses. To preserve the PREF specification, for each question j given to a new player, we ensure j is given to the player once in an even clause (2, 4, or 6) and once in an odd clause (1, 3, or 5).

We must add three players to exclude all possible reorderings of the length-6 refutation given by the clauses of the Small 123 Game, and in doing so end up with the "123 Game" (clauses reordered to expose the game structure):

Definition 5.2.2. *Define the **123 Game** by the following set of clauses:*

$$G_{123} := \left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 3 \\ 3 \\ 3 \\ 3 \\ 3 \\ 3 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \\ 2 \\ 3 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 3 \\ 1 \\ 3 \\ 1 \\ 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 3 \\ 1 \\ 2 \\ 2 \\ 3 \\ 1 \\ 1 \end{bmatrix} \right\}. \tag{5.2.3}$$

The 123-game has been constructed to make it difficult to reorder valid PREF specifications into refutations (for instance, it can be shown that no permutation of the valid length-6 PREF specifications

$$\pm \begin{bmatrix} 1 & 1 & 1 & -1 & -1 & -1 \end{bmatrix}^T$$

corresponds to a valid refutation).

In Section 3.4.3 we demonstrated that symmetric games have a refutation whenever they contain a PREF by construction of all required shift gadgets. In the (non-symmetric) 123 Game, there are no obvious shift gadgets present. This structure gives some intuition for why one would expect this game to have value 1 even though it has a PR. In the next section we prove that this intuition is correct; the 123-Game does in fact have value 1.

Value 1 Strategy

We define a simple strategy: measure in the Z basis if sent a 1, X if sent a 2, and Y if sent a 3.[2] If each player plays the 123 Game uses this strategy, it results in the following set of query observables:

[2]This is motivated by the observation that the 123 game provably does not have a refutation if we assume the measurements for different questions anticommute. We plan on addressing this intuition formally in an upcoming paper.

$$\mathcal{Q}_{123} := \left\{ \begin{bmatrix} Z \\ Z \\ Z \\ Z \\ Z \\ Z \end{bmatrix}, \begin{bmatrix} X \\ X \\ X \\ X \\ X \\ X \end{bmatrix}, \begin{bmatrix} Y \\ Y \\ Y \\ Y \\ Y \\ Y \end{bmatrix}, \begin{bmatrix} Z \\ X \\ Y \\ Z \\ X \\ Y \end{bmatrix}, \begin{bmatrix} X \\ Y \\ Z \\ Y \\ Z \\ X \end{bmatrix}, \begin{bmatrix} Y \\ Z \\ X \\ X \\ Y \\ Z \end{bmatrix} \right\}. \tag{5.2.4}$$

We also define a state on which these measurement can be made.[3]

$$|\psi_{123}\rangle := \frac{1}{\sqrt{8}}\Big(\Big[\,|000000\rangle + |111111\rangle\,\Big] - \Big[\,|100100\rangle + |001010\rangle + |010001\rangle + |011011\rangle + |110101\rangle + |101110\rangle\,\Big]\Big) \tag{5.2.5}$$

Theorem 5.2.3. *The strategy observables in $\mathcal{Q}_{123}$ measured on the state $|\psi_{123}\rangle$ win the 123 Game with probability 1. (The 123 Game has value 1.)*

Proof. For every string in $|\psi_{123}\rangle$, its compliment is also in $|\psi_{123}\rangle$ with the same sign. Additionally, every string in $|\psi_{123}\rangle$ has even Hamming weight. Overall, we may then conclude

$$XXXXXX\,|\psi_{123}\rangle = ZZZZZZ\,|\psi_{123}\rangle = |\psi_{123}\rangle \tag{5.2.6}$$

and hence

$$YYYYYY\,|\psi_{123}\rangle = (i)^6 XXXXXX\Big[ZZZZZZ\,|\psi_{123}\rangle\Big] = -\,|\psi_{123}\rangle\,. \tag{5.2.7}$$

It remains to check the outcomes for the last 3 queries. Explicit calculation gives

$$ZXYZXY\,|000000\rangle = (-1)\,|0\rangle\,|1\rangle\,(-i)\,|1\rangle\,(-1)\,|0\rangle\,|1\rangle\,(-i)\,|1\rangle = -\,|011011\rangle\,. \tag{5.2.8}$$

[3]This state was found through simple trial and error.

as well as

$$ZXYZXY\left|111111\right\rangle = \left|1\right\rangle\left|0\right\rangle i\left|0\right\rangle\left|1\right\rangle\left|0\right\rangle i\left|0\right\rangle = -\left|100100\right\rangle \tag{5.2.9}$$

Similarly, we can check

$$ZXYZXY\left|001010\right\rangle = (-1)\left|0\right\rangle\left|1\right\rangle i\left|0\right\rangle(-1)\left|0\right\rangle\left|0\right\rangle(-i)\left|1\right\rangle = \left|010001\right\rangle. \tag{5.2.10}$$

and

$$ZXYZXY\left|110101\right\rangle = \left|1\right\rangle\left|0\right\rangle(-i)\left|1\right\rangle\left|1\right\rangle\left|1\right\rangle i\left|0\right\rangle = \left|101110\right\rangle. \tag{5.2.11}$$

Putting this all together we see

$$ZXYZXY\left|\psi_{123}\right\rangle = \left|\psi_{123}\right\rangle, \tag{5.2.12}$$

with similar (permuted) arguments holding for $XYZYZX$ and $YZXXYZ$. □

5.2.2 Capped GHZ () Games

We begin by considering a family of symmetric games with commuting operater value < 1. The key property of this family is that to detect that $\omega^* < 1$ requires an exponentially high level in the ncSoS hierarchy, whereas the algorithm presented in Section 3.4.3 can do so in polynomial time.

Definition 5.2.4. *Define the n-th order **Capped GHZ game** as the 3-XOR game with alphabet size n and $m = 3n-1$ clauses defined by*

$$\mathrm{CG}_n := \left\{ \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 2 \\ +1 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \\ 2 \\ +1 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \\ 1 \\ +1 \end{bmatrix}, \ldots, \begin{bmatrix} (n-1) \\ n \\ n \\ +1 \end{bmatrix}, \begin{bmatrix} n \\ (n-1) \\ n \\ +1 \end{bmatrix}, \begin{bmatrix} n \\ n \\ (n-1) \\ +1 \end{bmatrix}, \begin{bmatrix} n \\ n \\ n \\ +1 \end{bmatrix} \right\}. \tag{5.2.13}$$

We claim $\omega^*(\mathrm{CG}_n) < 1$, and that it requires level at least $2^{n+1} - 2$ in the ncSoS hierarchy to detect this fact. Define the i-th triple of CG_n to be the clause set

$$A_i := \left\{ \begin{bmatrix} i \\ (i+1) \\ (i+1) \\ +1 \end{bmatrix}, \begin{bmatrix} (i+1) \\ i \\ (i+1) \\ +1 \end{bmatrix}, \begin{bmatrix} (i+1) \\ (i+1) \\ i \\ +1 \end{bmatrix} \right\}. \tag{5.2.14}$$

The clauses

$$\begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \end{bmatrix} \text{ and } \begin{bmatrix} n \\ n \\ n \\ +1 \end{bmatrix} \tag{5.2.15}$$

are called the *caps* (upper and lower) of the game and, for notational convenience, are referred to by A_0 and A_n. Our first claim shows that any refutation for CG_n must include both the upper and lower caps.

Lemma 5.2.5. *Let $\mathcal{E}$, $\mathcal{O}$ be minimal multiplicity equivalent multisets of queries taken from CG_n, so $\mathcal{E} \sim \mathcal{O}$ and no clause appears in both $\mathcal{E}$ and $\mathcal{O}$. If $\mathcal{E} \uplus \mathcal{O}$ contains some $x \in A_j$ with $j \notin \{0, n\}$, then $\mathcal{E} \uplus \mathcal{O}$ also contains clauses drawn from A_{j-1} and A_{j+1}.*

Proof. Without loss of generality, we assume

$$x = \begin{bmatrix} j \\ (j+1) \\ (j+1) \\ +1 \end{bmatrix} \in \mathcal{E}. \tag{5.2.16}$$

We then proceed by contradiction. If no clause from A_{j-1} is contained in $\mathcal{O}$ then the multiplicity of letter j for wire 1 in $\mathcal{O}$ cannot match $\mathcal{E}$, and the contradiction is immediate.

To prove the second claim, assume x occurs in $\mathcal{E}$ with multiplicity λ, and no terms from A_{j+1} are contained in $\mathcal{O}$. Then, in order to match the $(j+1)$ multiplicity on the 2nd and

3rd wires, clauses

$$y_1 = \begin{bmatrix} (j+1) \\ j \\ (j+1) \\ +1 \end{bmatrix} \text{ and } y_2 = \begin{bmatrix} (j+1) \\ (j+1) \\ j \\ +1 \end{bmatrix} \tag{5.2.17}$$

must both occur in $\mathcal{O}$ with multiplicity λ. Then we find $(j+1)$ occurs on the first wire of $\mathcal{E}$ with multiplicity 0, and on the first wire of $\mathcal{O}$ with multiplicity 2λ. Then $\mathcal{E}$ and $\mathcal{O}$ cannot be multiplicity equivalent, and this contradiction proves our result. □

A bound on the minimum length refutations for CG_n follows in a straightforward manner from Lemma 5.2.5.

Theorem 5.2.6. *The minimal length refutation for* CG_n *has length at least* $2^{n+1} - 2$.

Proof. We show the minimal sized multiplicity equivalent multisets $\mathcal{E}$ and $\mathcal{O}$ formed by elements of CG_n have size at least $2^{n+1} - 2$. By Lemma 5.2.5 the lower cap A_0 of CG_n is contained in either $\mathcal{E}$ or $\mathcal{O}$. Without loss of generality, assume it is contained in $\mathcal{E}$.

Then $\mathcal{E}$ contains letter 1 on every wire, and by minimality we know $A_0 \cap \mathcal{O} = \emptyset$. Since $\mathcal{E}$ and $\mathcal{O}$ are multiplicity equivalent multisets, we conclude $A_1 \subseteq \mathcal{O}$. But then $\mathcal{O}$ has two 2s on each wire, and by minimality $A_1 \cap \mathcal{E} = \emptyset$. So we conclude $(A_2)^2 \in \mathcal{E}$, where the notation A_2^2 denotes the multiset containing two copies of each element of A_2, and containment of one multiset in another implies containment of each element with at least it's multiplicity. Continuing in this vein, we see (assuming even n for the assignment of A_n below, though this does not affect the counting):

$$A_0 \uplus (A_2)^2 \uplus (A_4)^8 \ldots (A_n)^{2^{n-1}} \subseteq \mathcal{E} \text{ and } A_1 \uplus (A_3)^4 \uplus (A_5)^{16} \ldots (A_{n-1})^{2^{n-2}} \subseteq \mathcal{O}. \tag{5.2.18}$$

The total number of clauses contained in $\mathcal{E} \uplus \mathcal{O}$ is then given by

$$1 + 3(2^0) + 3(2^1) + \ldots 3(2^{n-2}) + 2^{n-1} = 1 + 3(2^{n-1} - 1) + 2^{n-1} \tag{5.2.19}$$

$$= 2^{n+1} - 2. \tag{5.2.20}$$

Any refutation gives rise to even and odd multiplicity equivalent multisets $\mathcal{E}$ and $\mathcal{O}$, and the above demonstrates that their combined size must be $\geq 2^{n+1} - 2$, proving the lower bound on refutation length. □

Theorem 5.2.6 shows that there exists a pseudodistribution on the clauses of CG_n which appears to have value 1 to a level exponential in the ncSoS hierarchy (proving Theorem 5.1.2). The minimal length multisets constructed in the proof of Theorem 5.2.6 are in fact multiplicity equivalent and the parity bits multiply to -1 (there is exactly one copy of A_0, which is the only question with $s_i = -1$) meaning CG_n contains a PREF. Since CG_n is a symmetric game, these two properties are sufficient to ensure a refutation exists (Section 3.4.3) giving $\omega^*(\mathrm{CG}_n) < 1$.

5.2.3 Asymptotically Perfect Difference (APD) Games

We next construct a family of k-XOR games, parameterized by $K \in \mathbb{N}$, with $k = 2^K - 1$, $m = 2^K$ clauses, and asymptotically perfect difference: each game in the family is a noPREF game, meaning

$$\omega^*(\mathrm{APD}_K) = 1, \tag{5.2.21}$$

while

$$\omega(\mathrm{APD}_K) \sim \frac{1}{2} + \sqrt{\frac{K}{2^K}} \sim \frac{1}{2} + \sqrt{\frac{\log k}{k}} \tag{5.2.22}$$

indicating that the difference is asymptotically as large as possible,

$$\lim_{K\to\infty} 2\left(\omega^*(\mathrm{APD}_K) - \omega(\mathrm{APD}_K)\right) = 1. \tag{5.2.23}$$

Definition 5.2.7. *Define the* ***Asymptotically Perfect Difference*** *family of XOR games parameterized by a scale $K \in \mathbb{N}$ as the set of games with alphabet size $n = 2$, $k = 2^K - 1$ players, and $m = 2^K$ clauses:*

$$\mathrm{APD}_K := \left\{ \begin{bmatrix} q_i \\ s_i \end{bmatrix} : q_i^{(\alpha)} = B_{(K)}^{\alpha,i} + 1 \right\}. \tag{5.2.24}$$

The s_i are defined to adversarially minimize $\omega(\mathrm{APD}_K)$ and the matrix $B_{(K)} \in \{0,1\}^{2^K \times 2^K}$ is recursively defined by

$$B_{(0)} = \begin{bmatrix} 1 \end{bmatrix} \tag{5.2.25}$$

$$B_{(K+1)} = \begin{bmatrix} \bar{B}_{(K)} & B_{(K)} \\ B_{(K)} & B_{(K)} \end{bmatrix} \tag{5.2.26}$$

with $\bar{B}$ produced by switching $0 \leftrightarrow 1$ for all entries of B. Equivalently, $\bar{B} = J - B$, with J the all-ones matrix.

Note that by the game definition, the $m \times kn = (2^K) \times (2 * (2^K - 1))$ game matrix $A_{(K)}$ for APD_K consists of the first $2^K - 1$ columns of $B_{(K)}$ interleaved with the first $2^K - 1$ columns of $\bar{B}_{(K)}$:

$$A_{(K)} = \begin{bmatrix} B^{\cdot,1}_{(K)} & \bar{B}^{\cdot,1}_{(K)} & B^{\cdot,2}_{(K)} & \bar{B}^{\cdot,2}_{(K)} & \dots & B^{\cdot,2^K-1}_{(K)} & \bar{B}^{\cdot,2^K-1}_{(K)} \end{bmatrix}. \tag{5.2.27}$$

The pairs of columns in $A_{(K)}$ corresponding to the two possible outputs from each player are complementary, making $A_{(K)}$ a valid game matrix.

Note that APD_2 is exactly the GHZ game, so the APD family is a particular many-player generalization of GHZ:

$$B_{(2)} = \begin{bmatrix} \bar{B}_{(1)} & B_{(1)} \\ B_{(1)} & B_{(1)} \end{bmatrix} \tag{5.2.28}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \tag{5.2.29}$$

$$\implies A_{(2)} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}. \tag{5.2.30}$$

Exchanging columns $3 \Leftrightarrow 6$ and $4 \Leftrightarrow 5$, corresponding to a relabeling of players and inputs,

gives A_{GHZ} as defined in (3.3.3). The choice of parity bits in GHZ is known to minimize the classical value, exactly matching the definition of APD_2.

We now prove our claims about the commuting operater and classical values of APD games.

Commuting Operater Value

Lemma 5.2.8. *For all K, $B_{(K)}$ has trivial kernel.*

Proof. We proceed by induction.

1. **Base case:** $B_{(0)} = \begin{bmatrix} 1 \end{bmatrix}$ has trivial kernel by inspection.

2. **Induction step:** Assume $B_{(K)}$ has trivial kernel, i.e. $B_{(K)}x = 0 \implies x = 0$. We now demonstrate that $B_{(K+1)}$ has trivial kernel by contradiction.

   Assume to the contrary that $B_{(K+1)}x = 0$ for $x \neq 0$. We can expand the blocks of this equation:

   $$B_{(K+1)}x = \begin{bmatrix} \bar{B}_{(K)} & B_{(K)} \\ B_{(K)} & B_{(K)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0. \tag{5.2.31}$$

   By the bottom block,

   $$B_{(K)}x_1 + B_{(K)}x_2 = 0 \tag{5.2.32}$$

   $$B_{(K)}(x_1 + x_2) = 0 \tag{5.2.33}$$

   $$\implies x_2 = -x_1. \qquad \text{(Induction hypothesis)} \tag{5.2.34}$$

Using this relation in the top block, we have

$$0 = \bar{B}_{(K)}x_1 - B_{(K)}x_1 \tag{5.2.35}$$

$$= \left(J - B_{(K)} - B_{(K)}\right) x_1 \tag{5.2.36}$$

$$2B_{(K)}x_1 = Jx_1 \tag{5.2.37}$$

$$2B_{(K)}x_1 = \begin{bmatrix} \sum_i x_1^i \\ \sum_i x_1^i \\ \dots \end{bmatrix}. \tag{5.2.38}$$

Noting that the bottom row of $B_{(K)}$ is always the all-ones vector by the definition, we can consider the bottom element of (5.2.38)

$$2\sum_i x_i = \sum_i x_i \tag{5.2.39}$$

$$\implies \sum_i x_i = 0. \tag{5.2.40}$$

This means $Jx_1 = 0$, which together with (5.2.37) gives:

$$B_{(K)}x_1 = 0. \tag{5.2.41}$$

By the induction hypothesis, this must mean $x_1 = 0 = x_2$, contradicting $x \neq 0$.

□

Theorem 5.2.9. *For all K, APD_K is a noPREF game, and thus has a MERP strategy with value 1 and $\omega^*(\mathrm{APD}_K) = 1$. The same holds for any choice of $\hat{s}$.*

Proof. First, we demonstrate that $(A_{(K)})^T$ has trivial kernel.

We have from Lemma 5.2.8 that $B_{(K)}$ has trivial kernel, and thus its rank is $m = 2^K$. $A_{(K)}$ includes all columns of $B_{(K)}$ except the last, the all-ones vector. $A_{(K)}$ also includes columns of $\bar{B}_{(K)}$. Adding a column of $B_{(K)}$ to the corresponding column of $\bar{B}_{(K)}$ produces the all-ones vector, so it must be in the column-span of $A_{(K)}$ as well. Finally, this means the

column span of $A_{(K)}$ includes the column span of $B_{(K)}$ and so the rank must be m. By the rank-nullity theorem, matrix $(A_{(K)})^T$ has trivial kernel.

The PR constraints are unsatisfiable, so APD_K is a noPREF game. By Theorem 3.3.30, APD_K has a MERP strategy with value 1 and $\omega^*(\text{APD}_K) = 1$. □

Classical Value

We extend the motivating classical results presented in Section 3.5.3 to analyze the classical value of the APD family. Corollary 3.5.4 demonstrates that the set of outputs achievable by a deterministic classical strategy is given exactly by $\mathcal{Y}_2 := \text{im}_{\mathbb{F}_2}(A)$. Recalling that $\sigma_2 = \dim \mathcal{Y}_2$, we see that when $\sigma_2 \ll m$, the set of deterministically achievable outputs is much smaller than the total space of possible parity bit vectors, and so we should be able to find a vector $\hat{s} \in \mathbb{F}_2^m$ with large Hamming distance from all outputs in $\mathcal{Y}_2$. In this section the probabilistic method is used to formalize this argument.

Theorem 5.2.10. *Let A be an XOR game matrix, for which $\sigma_2 \leq \delta m$. Then there exists a parity bit vector $\hat{s} \in \mathbb{F}_2^m$ for which the game $G \sim (A, \hat{s})$ has value at most*

$$\frac{1}{2} + \sqrt{\frac{\delta}{2}} \tag{5.2.42}$$

Proof. This argument is a close variant of the usual Hamming bound on error-correcting codes. Let S denote the set of $\hat{s}$ within distance $m(1/2 - \epsilon)$ of some point in $\mathcal{Y}_2$. Using the fact that $|\mathcal{Y}_2| = 2^{\sigma_2} \leq 2^{\delta m}$ we have

$$|S| \leq 2^{\delta m} \sum_{k \leq m\left(\frac{1}{2} - \epsilon\right)} \binom{m}{k}. \tag{5.2.43}$$

We bound the sum over binomial coefficients with the Chernoff bound to obtain

$$|S| \leq 2^{\delta m} 2^{m(1-2\epsilon^2)}. \tag{5.2.44}$$

Then for any $\varepsilon > \sqrt{\delta/2}$ there exists a $\hat{s}$ with distance $\geq m(1/2 - \varepsilon)$ from any point in $\mathcal{Y}_2$. This corresponds to value $1/2 + \varepsilon$. □

We now consider the specific case of the APD game and demonstrate the asymptotic limit of the classical value.

Lemma 5.2.11. *Given $K \in \mathbb{N}$, the APD game* APD_K *has* $\sigma_2(\mathrm{APD}_K) = K + 1$.

Proof. Recall that σ_2 is the dimension of $\mathcal{Y}_2$, the image of $A_{(K)}$ viewed as a map taking $\mathbb{F}_2^{kn} \to \mathbb{F}_2^m$. Equivalently, $\mathcal{Y}_2$ is the column span of $A_{(K)}$ taken over $\mathbb{F}_2$, and for this proof we use this view. By the same argument as Theorem 5.2.9, the column span of $A_{(K)}$ is identical to the column span of $B_{(K)}$. We prove this Lemma by induction over the $B_{(K)}$:

1. **Base case:** $B_{(0)} = \begin{bmatrix} 1 \end{bmatrix}$ giving $\sigma_2 = 1$ by inspection.

2. **Induction step:** Assume $\sigma_2(\mathrm{APD}_K) = K + 1$, meaning the dimension of the column span of $B_{(K)}$ over $\mathbb{F}_2$ is $K + 1$. We can write $B_{(K+1)}$ in block format:

$$B_{(K+1)} = \begin{bmatrix} \bar{B}_{(K)} & B_{(K)} \\ B_{(K)} & B_{(K)} \end{bmatrix} = \begin{bmatrix} (J - B_{(K)}) & B_{(K)} \\ B_{(K)} & B_{(K)} \end{bmatrix} = \begin{bmatrix} (J + B_{(K)}) & B_{(K)} \\ B_{(K)} & B_{(K)} \end{bmatrix} \text{ (over } \mathbb{F}_2\text{)} \tag{5.2.45}$$

   All columns in the right block of (5.2.45) take the form $\begin{bmatrix} x & x \end{bmatrix}^T$, so their span is

$$\mathsf{S} := \left\{ \begin{bmatrix} r & r \end{bmatrix}^T : r \in \mathcal{Y}_2(\mathrm{APD}_K) \right\}. \tag{5.2.46}$$

   On the other hand, all columns in the left block take the form $\begin{bmatrix} 1 \oplus x & x \end{bmatrix}^T = \begin{bmatrix} 1 & 0 \end{bmatrix}^T + \begin{bmatrix} x & x \end{bmatrix}^T$. The form of the right block span guarantees $\begin{bmatrix} 1 & 0 \end{bmatrix}^T$ is linearly independent from the right columns. Thus the total column span is

$$\mathcal{Y}_2(APD_{K+1}) = \mathsf{S} \cup \left(\mathsf{S} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) \tag{5.2.47}$$

   and $\sigma_2(APD_{K+1}) = \sigma_2(\mathrm{APD}_K) + 1 = (K + 1) + 1$, completing the induction step.

□

Theorem 5.1.3. The APD family has classical value

$$\frac{1}{2} \leq \omega(\mathrm{APD}_K) \leq \frac{1}{2} + \sqrt{\frac{K+1}{2^{K+1}}} \leq \frac{1}{2} + \sqrt{\frac{\log k}{k}}. \tag{5.2.48}$$

Proof. The lower bound of $\frac{1}{2}$ applies to all XOR games since a random assignment will satisfy half the clauses in expectation.

For the first upper bound, note that for APD family, $m = 2^K$ and from Lemma 5.2.11, $\sigma_2 = K + 1$. Then Theorem 5.2.10 yields the bound

$$\omega(\mathrm{APD}_K) \leq \frac{1}{2} + \sqrt{\frac{\sigma_2}{2m}} = \frac{1}{2} + \sqrt{\frac{K+1}{2^{K+1}}} \leq \frac{1}{2} + \sqrt{\frac{K}{2^K}}. \tag{5.2.49}$$

The last bound in the theorem statement is obtained by noting that $K = 2^k$. □

Finally, we conclude by mentioning that even though the APD construction may require an exponential time to choose the adversarial s_i, one can achieve the same asymptotic difference with high probability by choosing the s_i uniformly at random. This is implicit in the proof of Theorem 5.2.10, which implies that a randomly chosen $\hat{s}$ has value $\geq 1/2 + \varepsilon$ with probability $\leq 2^{(\delta - 2\varepsilon^2)m}$. Note as well that $\omega^* = 1$ for any choice of s_i, according to Theorem 5.2.9.

5.3 Random Games

The previous sections give a complete characterization of symmetric games with commuting operator value 1. However, as demonstrated by the final example of the previous Section (5.2.1), non-symmetric games remain, in general, hard to characterize. One area where we can make some progress is in understanding the value of randomly generated XOR games. We will work in a model, specified in Definition 3.3.5, where each clause is sampled uniformly with replacement from the set of all possible clauses.

The classical value of random CSPs[4] in this model has been intensely studied for several predicates including XOR, and it is useful to summarize the classical results. While determin-

[4] As noted in Section 3.1, CSPs and games are closely related. Classically, the difference between a CSP and the associated symmetric game is that each player in a game may play according to a different assignment of the variables; thus, the value of a CSP is always less than or equal to the classical value of the associated symmetric game.

ing the exact classical value of a random k-XOR instance for $k \geq 3$ remains hard, union bound arguments can give probabilistic bounds on the classical value of random k-XOR instances, in terms of the number of variables n and the number of clauses m. Combining these with second moment-type arguments and combinatorial analysis has revealed the existence of SAT and UNSAT phases for random instances in the limit of large n, which are separated by a sharp threshold in m [51]. For $k = 3$, this threshold occurs at $m/n \approx 0.92$ [21]. When m/n is below the threshold, a random 3-XOR instance has value 1 with probability approaching 1 as $n \to \infty$, while when m/n is above the threshold, a random instance has value 1 with probability approaching 0, and in fact, in the UNSAT phase, it is known that the value is close to 1/2. In addition to the true value, one can study the performance of the SoS algorithm on random instances. A key result in this direction is that of Grigoriev [27], who showed the existence of a region in the UNSAT phase with classical value close to 1/2, but for which the classical SoS algorithm reports a classical value of 1 until a high level in the SoS hierarchy. In the language we have developed thus far, he showed this by proving that random XOR games with appropriately chosen m and n do not admit any short-length classical refutations. One can interpret this result as showing the existence of a phase which is both UNSAT and computationally intractable.

The goal of this section is to prove a quantum analogue of these results. We are limited in one important sense: classically, the existence of an UNSAT phase with value close to 1/2 is shown via a union bound over the set of possible classical strategies, but this tool is no longer available to us for quantum strategies. Using our refutation-based technology, the best we can say is that the commuting operater value of a game is bounded a small distance away from 1 (see Section 3.4.1). At the same time, the quantum case presents us with an opportunity to go beyond what is possible classically: while the classical SoS algorithm has a natural upper bound at level kn, no such bound exists for the ncSoS algorithm. We could thus potentially improve on Grigoriev's result to prove a superexponential lower bound on the runtime of ncSoS.

We work subject to these considerations. In one direction, we know that for any G, $\omega^*(G) \geq \omega(G)$, and so we immediately get an entangled SAT phase for 3-XOR games with $m \lessapprox 0.92n$. In the other direction we show the existence of an entangled UNSAT phase:

specifically, we show that there exists a constant C_k depending only on the number of players k such that random games with more than $C_k n$ queries have commuting operater value < 1 with high probability. For 3-XOR games we find $C_3 \lessapprox 4$. Our bounds on the entangled SAT and UNSAT phases are only a constant factor apart, leaving open the possibility of a sharp threshold behavior as in the classical case.

Further, in analogy with Grigoriev's results, we also show that random XOR games with $m = O(n)$ queries have, w.h.p., no refutation with length less than $\Omega\left(n \log(n)/\log(\log(n))\right)$. By Lemma 3.4.4, this implies ncSoS takes superexponential time to show these games have value < 1.

5.3.1 SAT Phase

To start, we will show how the existence of a SAT phase for k-XOR viewed as a CSP implies the existence of such a phase for k-player XOR games. This is a simple consequence of the connection between games and CSPs.

Lemma 5.3.1. *For every k-XOR game G with m clauses and n variables, there exists a corresponding k-XOR CSP instance Φ_G with the same number of clauses and variables, such that if $\mathrm{val}(\Phi_G) = 1$, then $\omega(G) = 1$. Moreover, when G is chosen at random according to the distribution in Definition 3.3.5, the induced definition over Φ_G is the one generated by uniformly sampling m clauses over n variables with replacement.*

Proof. For each clause $(q_{i_1}, \ldots, q_{i_k}, s_i) \in G$, create a clause $x_{i_1} x_{i_2} \ldots x_{i_k} = s_i$ in Φ_G, where the variables x_i are taken over $\{\pm 1\}$. (We allow Φ_G to contain repeated clauses.) It is clear that Φ_G has the same number of clauses and variables as in G, and that if G is random then Φ_G is distributed as in the lemma statement.

If Φ has value 1, let x be a satisfying assignment for Φ. Then the classical strategy where all players play according to x is a strategy for G achieving value 1. Hence $\mathrm{val}(\Phi_G) = 1$ implies that $\omega(G) = 1$. □

Corollary 5.3.2. *For every k, there exists a constant B_k such that for any $b < B_k$, a random k-XOR game with $m = bn$ clauses will have $\omega(G) = \omega^*(G) = 1$ with probability approaching 1 as $n \to \infty$.*

Proof. The analogous statement for k-XOR CSP instances is proved in Theorem 16 of [51]. Let B_k be the threshold appearing in that theorem. By Lemma 5.3.1, if we sample a random k-XOR game G with $m = bn$ clauses, then the associated CSP instance Φ_G will be a random CSP instance with bn clauses, and thus have value 1 with probability approaching 1. Hence, $\omega(G) = \omega^*(G) = 1$ with probability approaching 1 as well. □

For $k = 3$, the constant B_k can be computed to be ≈ 0.92 [21].

5.3.2 UNSAT Phase

Since we are considering general random XOR games, we cannot appeal to the shift gadgets available by symmetry. Instead we use probabilistic analysis to show that such gadgets exist with high probability, given enough clauses. Below we give the analysis for the specific case of random 3-XOR games. The analysis for general k proceeds identically, with different constants depending on the number of players.

Lemma 5.3.3. *Let G be a randomly generated 3-XOR game defined by the set M of queries and associated parity bits, with $|M| = m \geq 3.3n$. Then with probability $1 - o(1)$, there exists a set $N_{3,1} \subseteq [n]$ with $|N_{3,1}| > 0.95n$ such that for all $a, b \in N_{3,1}$, G contains a shift gadget*

$$S^{3\to 1}(ab).$$

Proof. Consider a bipartite graph between two sets of n vertices. Label one set of vertices by $([n], 3)$, and the other by $([n], 2)$. Add an edge between $(j, 3)$ and $(j', 2)$ iff there exists a query

$$\begin{bmatrix} r \\ j' \\ j \end{bmatrix} \in M$$

where $r \in [n]$ is arbitrary. Label the edge by the index of the query corresponding to it. Our key observation is that that $S^{3\to 1}(ab)$ can be constructed from the queries corresponding to a walk from $(a, 3)$ to $(b, 3)$ in the graph.

Because queries are randomly generated, edges in this graph are randomly generated as well. So our graph is a $G_{n,n,m}$ Erdös-Rényi random bipartite graph. A technical result (Lemma 5.3.4) gives that this graph is at least as connected as $\hat{G}_{n,n,p}$ – a random bipartite graph in which each edge is present independently with probability $p = m/n^2 - \epsilon/n = (3.3 - \epsilon)/n$, where ϵ is an arbitrary small constant.

Finally, applying a Galton-Watson style argument to this random graph shows [37, Theorem 9] that with probability $1 - o(1)$ it contains a "giant component" that touches at least γn vertices of $([n], 3)$, where γ is the unique solution in the interval $(0, 1]$ to the equation

$$\gamma + \exp\left(pn(\exp(-pn\gamma) - 1)\right) = 1 \implies \gamma > 0.95.$$

□

Lemma 5.3.4 (Relating random graph models)**.** *Let $G \sim G_{N,N,m}$ with $m = CN$. Further let $\hat{G} \sim \hat{G}_{N,N,p}$ with $p = (C - \epsilon)/N$, for arbitrary small constant ϵ. For any value Z, if $\hat{G}$ contains a connected component of size Z with probability $1 - o(1)$ then G also contains a connected component of size Z with probability $1 - o(1)$.*

Proof. We couple the distributions used to generate $\hat{G}$ and G. In particular, a graph G can be generated by choosing a graph $\hat{G}$, then randomly adding or removing edges until the graph has exactly m edges. As long as we only add edges, this process will only increase the size of the largest connected component in the graph. Letting $E(\hat{G})$ be the set of edges of a graph $\hat{G}$, we find

$$\mathsf{E}|E(\hat{G})| = N^2 p = (C - \epsilon)N$$

and so

$$\mathsf{p}|E(\hat{G})| > m \leq \exp\left(-\epsilon^2 N/3\right) = o(1)$$

by a Chernoff bound. □

Lemma 5.3.3 tells us that, given a large enough number of queries, most variables can

be shuffled in exactly the manner described in Section 3.4.2. If we consider only queries involving these variables, we should then be able to construct refutations from PREFs using exactly the techniques described in the later half of that section. In fact, we only need to restrict to those variables on $k-2$ of the wires, since cancellations on the first two wires are automatic (see, in particular, the proof of Lemma 3.4.22).

If a large enough number of queries remain one would expect that they admit a PR with high probability. This fact is proved below.

Lemma 5.3.5. *For any k-XOR game G with m queries, alphabet size n and*

$$m - kn = \delta > 0, \tag{5.3.1}$$

if the parity bits for G are picked randomly then G has a PREF with probability at least $1 - 2^{-\delta}$.

Proof. By definition, a PREF specification is any vector $z \in \mathbb{Z}^m$ satisfying

$$A^T z = 0 \text{ and} \tag{5.3.2}$$

$$\hat{s}^T z = 1. \tag{5.3.3}$$

When $m > kn$, the matrix A^T has rank $\leq kn$. By the rank-nullity theorem, the kernel of A^T has dimension $\geq m - kn$, and so the are at least δ linearly independent vectors z satisfying (5.3.2). If the parity bits are chosen randomly, each of these vectors z satisfy (5.3.3) with independent probability $1/2$, and the result follows. □

Finally, we use our lemmas to prove the specific $k = 3$ case of the random game threshold.

Theorem 5.1.4. Let G be a random 3-XOR game with $m = \lceil 3.3n \rceil$ clauses on an alphabet of size n. Then, with probability $1 - o(1)$, G has value < 1.

Proof. Let $N_{3,1}$ be defined as in Lemma 5.3.3, and extend this definition to $N_{3,2}$ analogously. Define

$$N_3 := N_{3,1} \cap N_{3,2}. \tag{5.3.4}$$

Let γ be defined as in Lemma 5.3.3. A union bound then gives that the expected size of N_3 is bounded below by

$$(1 - 2(1 - \gamma)) > 0.9n.$$

Finally we let M be the set of queries for G, then define

$$M' := \{(q^{(1)}, q^{(2)}, q^{(3)}) \in M : q^{(3)} \in N_3\}.$$

If N_3 were independent of M', we could conclude

$$\mathsf{E}|M'| = m\frac{|N_3|}{n} > 3.01n \tag{5.3.5}$$

and then, by concentration,

$$\mathsf{p}|M'| < 3.009 \lesssim \exp(-n) = o(1). \tag{5.3.6}$$

M and N_3 are not independent, but a technical lemma (Lemma 5.3.6) shows their correlation can only increase the size of M', hence (5.3.6) remains valid.

Now consider a game G' consisting of only the clauses of G with queries in M'. M' has been constructed such that G' has shuffle gadgets for any wire of a pair of queries drawn from M'. Furthermore $|M'| - 3n \geq 0.009n$ with high probability, so by Lemma 3.4.21 and Lemma 5.3.5, we can then conclude G' contains a complete refutation with probability $1 - o(1)$. Since G' contains a subset of the clauses of G, this also means G contains a complete refutation with probability $1 - o(1)$. □

Lemma 5.3.6. *Let G be a random 3-XOR game on m clauses, and let N_3 and M' be defined as in the proof of Theorem 5.1.4. If there exists some constant δ for which*

$$\mathsf{E}|N_3| \geq \delta n$$

with probability $1 - o(1)$*, then we have, for any* $\epsilon > 0$ *that*

$$\mathsf{E}|M'| \geq (\delta - \epsilon)m$$

with probability $1 - o(1)$ *as well.*

Proof. We first move from the random game G to the random game $\hat{G}$, in which the total number of clauses isn't fixed, but rather every possible clause appears in the game with probability[5][6]

$$\frac{m - \epsilon_1}{2n^3}. \tag{5.3.7}$$

We also define the variables $\hat{N}_3$, $\hat{M}$ and $\hat{M}'$, which depend on $\hat{G}$ in exactly the same way the unhatted variables depends on G. By an argument identical to the one used in the proof of Lemma 5.3.4, lower bounds on the size of $\hat{M}'$ will carry over to lower bounds on the size of M' for G with high probability.

The techniques used to bound the size of N_3 work equally well on $\hat{N}_3$, and so

$$|\hat{N}_3| \geq (\delta - \epsilon_1 - \epsilon_2)n \tag{5.3.8}$$

with probability $1 - o(1)$.

Now we let A be some arbitrary subset of $[n]$ of size $\lfloor(\delta - \epsilon_1 - \epsilon_2)n\rfloor$, and define

$$\hat{M}(A) = \{(q^{(1)}, q^{(2)}, q^{(3)}) \in \hat{M} : q^{(3)} \in A\}.$$

Since A is arbitrary, it is immediate that

$$\mathsf{E}|\hat{M}(A)| = (\delta - \epsilon_1 - \epsilon_2)m \tag{5.3.9}$$

[5]Note the factor of 2 in the denominator comes from the choice of parity bit.
[6]Here and below we use ϵ_i to indicate arbitrary small constants.

and (by concentration)

$$\mathsf{p}|\hat{M}(A)| < (\delta - \epsilon_1 - \epsilon_2 - \epsilon_3)m = o(1). \tag{5.3.10}$$

Finally, we define the indicator random variables I_q to take on value 1 if $q \in \hat{M}$, and 0 otherwise. Our key observation is that

$$\mathsf{E}I_q \mid A \subseteq N_3 = \mathsf{E}I_q \left(\frac{\mathsf{p}A \subseteq N_3 \mid I_q = 1}{\mathsf{p}A \subseteq N_3} \right) \tag{5.3.11}$$

$$\geq \mathsf{E}I_q \tag{5.3.12}$$

and this remains true even after conditioning on the outcomes of other I_q's.

The indicator for the event

$$\left\{ |\hat{M}(A)| < (\delta - \epsilon_1 - \epsilon_2 - \epsilon_3)m \right\} \tag{5.3.13}$$

is a decreasing function of the $I_q's$, and so we can conclude

$$\mathsf{p}|\hat{M}(A)| \leq (\delta - \epsilon_1 - \epsilon_2 - \epsilon_3)m \mid A \subseteq N_3 \leq \mathsf{p}|\hat{M}(A)| < (\delta - \epsilon_1 - \epsilon_2 - \epsilon_3)m \tag{5.3.14}$$

Putting this all together, we find

$$\mathsf{p}|\hat{M'}| \leq (\delta - \epsilon_1 - \epsilon_2 - \epsilon_3)m \leq \mathsf{p}|\hat{M}(A)| < (\delta - \epsilon_1 - \epsilon_2 - \epsilon_3)m \mid A \subseteq N_3 \tag{5.3.15}$$

$$\leq \mathsf{p}|\hat{M}(A)| < (\delta - \epsilon_1 - \epsilon_2 - \epsilon_3)m \tag{5.3.16}$$

$$= o(1) \tag{5.3.17}$$

(where the first line follows from definition of M'). Since $|\hat{M'}|$ is a lower bound for $|M'|$ with high probability, we can set $\epsilon = \epsilon_1 + \epsilon_2 + \epsilon_3$ and conclude the result.

□

5.3.3 Lower Bound on Refutation Length (Sketch)

In this section we sketch the proof of the following theorem, which gives a lower bound that holds with high probability for the length of refutations of random 3-XOR games. Aside from the immediate implications of the theorem, this result is also significant because its proof uses a counting technique not found elsewhere in the paper.

Theorem 5.1.5. For any constant C, the minimum length refutation of a random 3-XOR game with $m = Cn$ queries on an alphabet of size n has length at least

$$\frac{en\log(n)}{8C^2\log(\log(n))} - o\left(\frac{n\log(n)}{\log(\log(n))}\right) \tag{5.3.18}$$

with probability $1 - o(1)$ (as $n \to \infty$). Hence, either $\omega^* = 1$ or $\Omega(n\log(n)/\log(\log(n)))$ levels of the ncSoS hierarchy are needed to witness that $\omega^* < 1$ for such games.

This significance of this result is twofold. Firstly, it gives a lower bound on refutation lengths which matches the length of refutations constructed using the methods of Section 3.4.3 to a factor of $O(\log(\log(n)))$. This suggests that the algorithm described in Section 3.4.3 is a near-optimal method for constructing refutations for symmetric XOR games.[7] Secondly, combining Theorem 5.1.5 with Lemma 3.4.4 show that an ncSoS proof that a random 3-XOR game has value < 1 requires going to level $\Omega(n\log(n)/\log(\log(n)))$ in the ncSoS hierarchy. This results in a runtime which is superexponential in n, and longer than the worst possible case for classical (commuting) SoS applied to XOR games (or boolean CSPs in general).

Theorem 5.1.5 is proved using a careful application of the first moment method. The full analysis is somewhat involved, and so we spend some time discussing the key ideas required for the proof. The proof hinges on enumerating possible refutations in a somewhat non-intuitive way. Rather than building up a refutation of length ℓ query by query, we will instead write down all possible sequences of ℓ queries, and consider all the ways those queries could cancel to form a refutation. The key definition required to make this counting work is that of a *cancellation pattern*.

[7] Strictly speaking, this conclusion is motivated only for 3-XOR games. That being said, for larger k, Theorem 5.1.5 still gives a lower bound which is tight to a factor of $C_k\log(\log(n))$, and it is reasonable to expect that, with additional work, this lower bound could be tightened further.

Definition 5.3.7. *A length ℓ* ***one wire cancellation pattern*** *is a partition of $[\ell]$ into $\ell/2$ pairs of the form $\{(a_1, b_1), \ldots, (a_{\ell/2}, b_{\ell/2})\}$ with*

$$a_i < b_i \text{ and } a_i < a_j \implies b_i > b_j \tag{5.3.19}$$

all $i, j \in [\ell/2]$ (no cancellation patterns exist for odd ℓ). When discussing k-XOR games, a length ℓ ***cancellation pattern*** *refers to an ordered list containing k one wire cancellation patterns.*

Definition 5.3.8. *Given a length ℓ cancellation pattern, the* ***locations*** *of that cancellation pattern are elements of $[\ell]$, corresponding to the positions at which queries can appear in the cancellation. The* ***sites*** *of the cancellation pattern are specified by coordinates $(\alpha, i) \in [k] \otimes [\ell]$, and represent the places where individual questions appear. Site (α_1, i_1) is said to* ***cancel*** *site (α_2, i_2) iff $\alpha_1 = \alpha_2$ and the pair (i_1, i_2) is contained in the α_1-th cancellation pattern. In this case, the pair of sites $((\alpha_1, i_1), (\alpha_2, i_2))$ is referred to as a* ***cancellation****.*

Definition 5.3.9. *Using matrix notation to specify individual letters in a word, a cancellation pattern is* ***valid*** *on a word W iff*

$$w_{\alpha_1, i_1} = w_{\alpha_2, i_2} \tag{5.3.20}$$

for all sites (α_1, i_1) and (α_2, i_2) which cancel one another.

By definition, a word cancels to the identity iff there exists at least one cancellation pattern which is valid on the word. It is also straightforward to give a combinatorial bound on the number of possible length ℓ cancellation patterns.

Claim 5.3.10. *The number of possible cancellation patterns on a single wire with ℓ locations is given by the $\ell/2$-th Catalan number, denoted by $\mathcal{C}_{\ell/2}$. The number of possible cancellation patterns on a length ℓ word formed from k-XOR queries is then given by*

$$\left(\mathcal{C}_{\ell/2}\right)^k \leq 2^{k\ell}. \tag{5.3.21}$$

Proof. Direct from the definition of Catalan numbers, and standard bounds on their size. See [62] for an extensive discussion. □

To illustrate the benefit of working in terms of cancellation patterns, we prove a simple theorem, regarding the existence of a restricted class of refutations.

Theorem 5.3.11. *Let $m \in o(n^{k/2})$. Then, as $n \to \infty$, a random k-XOR game with m queries on an alphabet of size n will contain a refutation in which every query is used at most once with probability at most $o(1)$.*

Proof. We apply the first moment method. There are $\ell!\binom{m}{\ell}$ ways of creating a word of length ℓ from the queries, and at most $2^{k\ell}$ cancellation patterns on the word. Since queries are all independent and randomly chosen, each cancellation pattern on a length ℓ word is valid with probability $(1/n)^{k\ell/2}$. Then the probability of a valid cancellation of any length is given by (using $(m)(m-1)...(m-2r) \leq m^{2r}$)

$$\sum_{r=1}^{m/2} \left[(2r)! \binom{m}{2r} (\mathcal{C}_r)^k (1/n)^{kr} \right] \leq \sum_{r=1}^{m/2} \left[2^k \left(m/n^{k/2} \right) \right]^{2r} \in o(1). \tag{5.3.22}$$

□

We now take a small detour, and use techniques similar to the one above to reprove a result of Grigoriev [27]. This is done to illustrate the power of these techniques, but also for completeness, as we will use Grigoriev's result in our proof of Theorem 5.1.5.

Theorem 5.3.12 (Originally proved in [27])**.** *Let G be a random 3-XOR game on the set of queries M, with $|M| = m = Cn$ and alphabet size n. Define a classical refutation to be a subset of queries $T \subseteq M$ such that*

$$|\{q \in T \mid q^{(\alpha)} = j\}| = 2mallj \in [n], \alpha \in \{1,2,3\} \tag{5.3.23}$$

(if written as a word, T would contain each $j \in [n]$ an even number of times on each wire). Then, with probability $1 - o(1)$ as $n \to \infty$ the shortest classical refutation contained in m has length at least en/C^2.

Proof. We again use the first moment method, paralleling the argument used in the proof of Theorem 5.3.11. We find $\binom{Cn}{\ell}$ ways of choosing ℓ queries from M, and $((\ell-1)!!)^3$ ways of pairing up letters on all rows once ℓ queries have been chosen (if ℓ is even). As before, each pair of letters is equivalent independently with probability $(1/n)$ and so by the union bound the probability of a classical refutation of length less than ℓ is bounded by

$$\sum_{r=1}^{\ell/2}\left[\binom{Cn}{2r}((2r-1)!!)^3(1/n)^{3r}\right] \leq \sum_{r=1}^{\ell/2}\left[(Cn)^{2r}\left(2^r r!\right)(1/n)^{3r}\right] \tag{5.3.24}$$

$$= \sum_{r=1}^{\ell/2}\left[r!\left(2C^2/n\right)^r\right] \tag{5.3.25}$$

$$\leq \sum_{r=1}^{\ell/2}\left[e\sqrt{r}\left(2C^2 r/(en)\right)^r\right]. \tag{5.3.26}$$

Noting this sum is $o(1)$ provided $\ell C^2/en < 1$ completes the proof. □

Returning to the informal proof of Theorem 5.1.5, the natural approach is to try to generalize the proof of Theorem 5.3.11 by allowing repeated queries and repeating the union bound analysis. Unfortunately, when queries are repeated not all cancellations are valid independent of one another, which makes it dramatically more difficult to compute the probability of a given cancellation pattern being valid. To accommodate this, we require additional terminology for discussing the different types of cancellations that can occur when a cancellation interacts with a word containing repeated queries. This is introduced below, along with a brief discussion of how these cancellations are accounted for in the full proof.

Definition 5.3.13. *Given a cancellation pattern on a word made up of queries from a random k-XOR game, define:*

- *The set of* ***independent cancellations*** *to be a maximal set of cancellations so that each cancellation is valid independent of all others in the set with probability* $1/n$.
- *The set of* ***dependent cancellations*** *to be the set of cancellations which are valid with probability 1 if all independent cancellations are valid.*
- *The set of* ***self cancellations*** *to be the set of all cancellations which are valid with*

probability 1 independent of all other cancellations (these occur when a query is canceled with itself).

*A **full cancellation pattern** is a cancellation pattern where cancellations are specified to be independent, dependent or self ahead of time, and this full cancellation pattern is valid on a word iff the sets defined above are compatible with the way the cancellations are labeled ahead of time.*

Note there is some freedom in which cancellations are chosen as dependent vs. independent. This ambiguity allows us to simplify the full proof, and is left in intentionally.[8] Semi-formally, we can now give the proof of Theorem 5.1.5 as follows:

Proof (semi-formal). Our goal is to show that, under the conditions of Theorem 5.1.5, any cancellation pattern on a word consisting of a small number of queries is valid with vanishing probability. We restrict our attention to minimum length refutations: refutations for which no subset of queries can be removed while leaving a valid refutation.

We then attempt a union bound argument in which we identify the various ways queries can interact with cancellation patterns in the refutation. We begin by segmenting the queries in the refutation into maximal strings of queries connected via dependent or self cancellations. We call these phrases. By definition, the phrases themselves must be connected by independent cancellations.

We can bound the number of ways of building a phrase of length k. The first query in a phrase can be a picked arbitrarily from a set of size m. After that, a query connected to a known query by a self-cancellation is fixed exactly, and concentration inequalities can be used to show that a query connected to a fixed query via a dependent cancellation is drawn from a set of size at most $m\log(n)/n$.[9] Then the ways of choosing queries such that they form the given phrase is bounded by

$$m\left(\frac{m\log(n)}{n}\right)^{k-1}. \tag{5.3.27}$$

[8] Of course, it could also be removed by fixing a convention for the cancellations which are labeled independent (i.e. choosing the lexicographically minimal set).

[9] Proved in Lemma 5.3.16

We next place some restrictions on the number and type of phrases that can occur in a refutation. By minimality, each phrase must contain at least one site involved in an independent cancellation (otherwise the phrase is "redundant"); then by parity each phrase must contain two. We also get a bound on the number of queries appearing an odd number of times. Removing all queries that occur an even number of times, and leaving only one copy of each query that occurs an odd number will produce a classical refutation. Theorem 5.3.12 then tells us that with probability $1 - o(1)$, any valid refutation must have en/C^2 queries which occur an odd number of times.

We then use a result from the technical proof: for p phrases and s sites with independent cancellations,

$$s \geq 2p + en/4C^2. \tag{5.3.28}$$

Using (5.3.27) to bound the number of ways each phrase occurs, and a factor of $1/\sqrt{n}$ per site in an independent cancellation (making $1/n$ per independent cancellation) we find that any full length-ℓ cancellation pattern is valid on some word of ℓ queries with probability at most

$$m^p \left(\frac{m\log(n)}{n}\right)^{\ell-p} \left(\frac{1}{n}\right)^{s/2} \leq m^p \left(\frac{m\log(n)}{n}\right)^{\ell-p} \left(\frac{1}{n}\right)^{p+en/8C^2} \tag{5.3.29}$$

$$\leq \left(\frac{1}{\log(n)}\right)^p \left(\frac{1}{n}\right)^{en/8C^2} (C\log(n))^\ell \tag{5.3.30}$$

$$\leq \left(\frac{1}{n}\right)^{en/8C^2} (C\log(n))^\ell . \tag{5.3.31}$$

Adding in a union bound over all possible length ℓ full cancellation patterns, we find the probability of a valid length ℓ cancellation pattern existing is at most

$$\mathcal{C}^3_{\ell/2} 3^{3\ell/2} \left(\frac{1}{n}\right)^{en/8C^2} (C\log(n))^\ell \leq 12^{3\ell/2} \left(\frac{1}{n}\right)^{en/8C^2} (C\log(n))^\ell \tag{5.3.32}$$

$$\leq \left(\frac{1}{n}\right)^{en/8C^2} (42C\log(n))^\ell . \tag{5.3.33}$$

Setting $m/n = C$ and following through the geometric series we find the probability of a

refutation of length less than or equal to ℓ existing is at most

$$\frac{42C\log(n)^{\ell+1}}{n^{en/8C^2}} + o(1) \tag{5.3.34}$$

where the $o(1)$ term comes from the use of results 5.3.12 and 5.3.16 in our proof. If follows that the total probability of refutation is $o(1)$ unless

$$\ell \geq \frac{en\log(n)}{8C^2\log(42C\log(n))} - 1 \tag{5.3.35}$$

completing the proof of Theorem 5.1.5. ≈□

While the proof above was hopefully convincing, it wasn't completely formal. A more careful proof that clearly discusses the various events the union bound is constructed over is given below.

5.3.4 Lower Bound on Refutation Length (Full Proof)

For the most part, the key ideas used in the proof of Theorem 5.1.5 are well covered in Section 5.3.3. The remaining details are primarily technical, but somewhat involved. We begin by formalizing the definition of a *phrase*, used informally above.

Definition 5.3.14. *Consider a full cancellation pattern consisting of dependent, self and independent cancellations. Let G be a graph with vertices corresponding to dependent or self cancellations in the cancellation pattern. Add an edge between vertices if the corresponding cancellations overlap at some location. The sets of cancellations corresponding to connected components in this graph are called* ***phrases****.*

Our analysis will require language specific to the ways in which queries and phrases can occur in a refutation. That language is introduced below.

Definition 5.3.15. *Given a refutation, we define the following sets:*

- *L_r is the set of locations located at the leftmost point in some phrase. We call queries at these locations* ***roots****.*

- *L_c is the set of all locations in phrases which are not the leftmost point of a phrase. Queries at these locations are called **constrained queries**.*
- *P is the set of all phrases in the cancellation pattern.*
- *$P_r \subseteq P$ is the set of all phrases for which every location in the phrase contains only self or dependent cancellations. Phrases in P_r are called **redundant phrases**.*
- *S is the set of all sites in independent cancellations.*

*Redundant phrases are so named because removing all queries contained in them still leaves a valid refutation. For this reason **minimal length refutations** are defined to be refutations that do not contain any redundant phrases.*

We now prove a few basic properties about the structure of refutations constructed from random queries.

Lemma 5.3.16. *Let $m = Cn$ for some constant C. Then, with probability $1 - o(1)$, all refutations for a random 3-XOR game on m queries with n variables will satisfy*

1. *The refutation contains at least en/C^2 distinct queries occurring an odd number of times.*
2. *The cancellations can be labeled so that $|S| \geq 2|P| + en/4C^2$.*
3. *For all queries q_i: the refutation implies that q_i cancels with at most $C\log(n)$ other queries on each wire.*

Proof. We prove 1 by appealing to [27]. Note we can obtain a classical refutation from a quantum refutation by taking a single copy of each query repeated an odd number of times. Then, we know from [27] (alternately Theorem 5.3.12) that there are en/C^2 distinct queries repeated an odd number of times in the quantum refutation.

To prove 2, we first note that every distinct query occurring an odd number of times must be involved in at least three independent cancellations (one per wire) across all locations where it appears, resulting in a total count of $3en/C^2$ cancellations. We will show that we can

relabel independent and dependent cancellations such that at least 1/4 of these independent cancellations are all contained in at most $en/4C^2$ phrases.

To do so, we begin by making a list of all queries occurring an odd number of times in our refutation, and consider a cancellation pattern on which only the self-cancellations have been fixed. We refer to a phrase induced by these self cancellations as a *subphrase*. We now extract a query from the list, pick an odd length subphrase involving that query (this subphrase may have length one), and mark three non-self cancellations coming from that subphrase as independent (one per wire). Next, we remove from our list any queries connected to this subphrase by the newly labeled independent cancellations. Removing the connected queries from the list ensures that any non-self cancellations involving elements remaining on our list will be independent from the cancellations we have labeled so far. We then repeat this process until we have exhausted all queries on our list, and then label the remaining cancellations in any valid manner.

Over this process, we remove at most three additional queries from the list for every subphrase we identify, so when we have exhausted all queries on this list (but before we have labeled any dependent cancellations), we will have at $en/4C^2$ subphrases containing at least $3en/4C^2$ independent cancellations. Each of these subphrases is contained in a phrase (since all locations are connected via self-cancellations) and labeling the remaining cancellations cannot change the cancellations already labeled as independent, so we have found at least $3en/4C^2$ independent cancellations contained in at most $en/4C^2$ phrases.

From here the proof of 2 is straightforward: by minimality, each phrase contains must contain at least one independent cancellation, and hence by parity each phrase must contain two. Furthermore, we have already identified a special set of at most $en/4C^2$ phrases which contain at least $3en/4C^2$ independent cancellations. Letting p_1 be the number of phrases identified so far, and p_2 be the number of phrases not contained in the set already identified, we see

$$|S| \geq 2p_2 + \frac{3en}{4C^2} \geq 2(p_2 + p_1) + \frac{en}{4C^2} = 2|P| + \frac{en}{4C^2} \tag{5.3.36}$$

as desired.

Finally, 3 follows from concentration of measure. We define $y(j)$ to be the random variable counting the number of queries with letter j on the top wire, so

$$y(j) = \left|\{i : q_i^{(1)} = j\}\right|. \tag{5.3.37}$$

It is then clear that

$$\mathsf{E}y(j) = m/n = C. \tag{5.3.38}$$

By a Chernoff bound

$$\mathsf{p}y(j) \geq C\log(n) \leq e^{-(\log^2(n)-1)C/3n} \tag{5.3.39}$$

$$\leq 1/n^{C\log(n)/3} = o(1) \tag{5.3.40}$$

and so a union bound argument gives the result for large n. □

The proof of Theorem 5.1.5 will follow from our observations in Lemma 5.3.16 and first moment arguments. To make clear the analysis, we first present a simple algorithm for generating minimal length refutations with length ℓ from a random set of queries G.

Algorithm 5.3.17 (Refutation generator)**.**

input: A set of m queries, with $m = Cn$, and parameter ℓ

output: A minimal refutation of length ℓ, or *failure*

1. Initialize ℓ locations where queries might be placed.
2. Randomly generate a cancellation pattern consisting of self, dependent and independent cancellations on the ℓ locations. Identify the phrases in this cancellation pattern.
3. If there is any redundant phrase, return *failure: not minimal.*
4. Randomly map queries to locations.
   (a) If the independent cancellations require there to be more than $C\log(n)$ queries which agree on any wire, or if the cancellation pattern would imply $|S| \leq 2|P| +$

$en/4C^2$, return *failure: improbable cancellation.*

(b) If self-cancellations occur between non-identical queries, or dependent cancellations are not implied by independent cancellations, return *failure: improper labeling.*

5. If any independent cancellations occur between queries which disagree on the wire where the cancellation is occurring, return *failure: invalid cancellation.*

6. Otherwise, return *success* along with the cancellation pattern and query mapping.

We prove Theorem 5.1.5 by proving two basic facts about Algorithm 5.3.17. Firstly, we show that, with high probability[10], there exists a random seed for which Algorithm 5.3.17 finds a refutation provided one exists. Secondly, we show the expected number of paths on which Algorithm 5.3.17 returns success is small unless ℓ is sufficiently large. We will prove these claims separately.

Theorem 5.3.18 (Correctness)**.** *Algorithm 5.3.17 only returns success when it finds a valid minimum length refutation. Furthermore, when the input queries are randomly selected, with probability* $1 - o(1)$ *the algorithm has a positive probability of finding all valid length* ℓ *refutations.*

Proof. The first claim is clear from inspection of the algorithm. The second follows from Lemma 5.3.16 and further inspection. In particular, the only refutations not found by the algorithm are those which require greater than $C\log(n)$ queries to agree on a wire, or those with a cancellation pattern for which

$$|S| \leq 2|P| + en/4C^2. \tag{5.3.41}$$

Lemma 5.3.16 tells us that these cases occur with probability $o(1)$ for randomly chosen queries. □

Theorem 5.3.19. *Given a randomly chosen set of* $m = Cn$ *queries as input, the expected number of minimal length* ℓ *refutations which can be found by Algorithm 5.3.17 is upper*

[10]It should be stressed that this with high probability refers to the randomness associated with choosing the queries provided as input to the algorithm, not the randomness associated with the algorithm's run.

bounded by

$$(1/n)^{e/2C^2}(42C\log(n))^{\ell}. \tag{5.3.42}$$

In particular, we expect to find no refutations until

$$\ell = \Omega(n\log(n)/\log(\log(n))) \tag{5.3.43}$$

Proof. We give an overcounting of the number of possible paths Algorithm 5.3.17 can take. We first note that a path can be completely specified by a choice of cancellation pattern and mapping of queries to locations.

Using our rough bound on the Catalan numbers, there are at most $\mathcal{C}^3_{\ell/2} \leq 4^{3\ell/2}$ different ways of pairing up all sites for cancellations. Since each cancellation can be one of three types, we find a total of

$$3^{3\ell/2}4^{3\ell/2} \leq 42^{\ell} \tag{5.3.44}$$

possible cancellation patterns.

We next give a rough (over)counting of the number of ways queries can be mapped to locations such that the cancellation pattern is not rejected in step 4 of the algorithm.

In particular, we allow arbitrary queries to be mapped to locations in L_r. After this mapping, we note all remaining locations are in L_c. Assuming the cancellation pattern was not rejected in step 4a, a location connected to a fixed query by a self cancellation can only have a single query mapped to it, and a location connected to a fixed query by a dependent cancellation can have at most $C\log(n)$ queries mapped. In total then, we find

$$m^{|L_r|}\left(C\log(n)\right)^{|L_c|} = m^{|P|}\left(C\log(n)\right)^{|L_c|} \tag{5.3.45}$$

possible mappings from queries to locations.

Finally, we bound the probability that our given query assignment doesn't fail in step 5 of the algorithm. Noting independent cancellations are, by definition, independent we find

the probability of failure is given by

$$\left(\frac{1}{n}\right)^{|S|/2}. \tag{5.3.46}$$

Since our cancellation pattern doesn't contain any redundant phrases, and was not rejected as improbable by the algorithm we also have

$$|S| \geq 2|P| + en/4C^2. \tag{5.3.47}$$

The overall expected number of successes for a given cancellation pattern can then be bounded by:

$$\begin{aligned} m^{|P|}(C\log(n))^{|L_c|}\left(\frac{1}{n}\right)^{|P|+en/8C^2} &= m^{|P|}\left(\frac{m\log(n)}{n}\right)^{\ell-|P|}\left(\frac{1}{n}\right)^{|P|+en/8C^2} && (5.3.48)\\ &\leq \left(\frac{1}{\log(n)}\right)^{|P|}\left(\frac{1}{n}\right)^{en/8C^2}(C\log(n))^{\ell} && (5.3.49)\\ &\leq \left(\frac{1}{n}\right)^{en/8C^2}(C\log(n))^{\ell} && (5.3.50) \end{aligned}$$

resulting in an overall bound on the expected number of successes for any length ℓ of

$$\left(\frac{1}{n}\right)^{en/8C^2}(42C\log(n))^{\ell}. \tag{5.3.51}$$

Summing the geometric series, the expected total number of refutations of length less than ℓ can then be bounded above by

$$\left(\frac{1}{n}\right)^{en/8C^2}\frac{(42C\log(n))^{\ell+1}-1}{(42C\log(n))-1}. \tag{5.3.52}$$

We see this is $o(1)$[11] unless

$$\ell \geq \frac{en\log(n)}{8C^2\log(\log(n))} - o\left(\frac{n\log(n)}{\log\log(n)}\right), \tag{5.3.53}$$

and the desired result follows from Markov's inequality. □

To close this section, we note Theorem 5.1.5 is immediate from Theorems 5.3.18 and 5.3.19.

[11] As a word of caution: it should be noted the $(en\log(n)) / \left(8C^2\log(\log(n))\right)$ only dominates when C is taken to be a constant with respect to n. When C scales with n the above analysis will still work, but requires more care in computing the final bound.

Chapter 6

Conclusion and Open Questions

There are two points of view one can take when summarizing the results in this thesis. Firstly, this can be understood as a thesis about XOR games. From this point of view, the key results of this thesis are contained in Chapters 3 and 4 and the random games section of Chapter 5, where symmetric, 3-player, and random XOR games are analyzed. The subgroup membership (and relatedly, the noPREF) characterizations of XOR games with perfect commuting operator value play a key role in all these arguments, as do MERP strategies, which emerge as optimal strategies for a surprisingly large class of games (though not all games – see 5.2.1). This point of view also leads to a natural class of open questions: while the techniques in this thesis work for a large class of games they also fail on others. What can we say about strategies for those games? Is there some generalization of MERP or the noPREF condition that works for 4 player or, even better, k player XOR Games? Given recent results about the complexity of optimal strategy for some nonlocal games [36], it is natural to ask how rich the class of optimal strategies for XOR games can get.

A second, complimentary view of this thesis is as a thesis that lays out a blueprint for an algebraic study of nonlocal games. A similar blueprint has already been laid out, and used to great effect, in the study of synchronous games [49, 34]. This thesis begins the process of generalizing this blueprint to non-synchronous games. From this point of view Chapter 2 is foundational. Also important is the view of the noPREF condition as subgroup membership mod K discussed in Chapter 4, and the MERP-PREF duality discussed in Sections 3.3.5 and 3.5.3. From this point of view a foundational question remains open:

we have an example where a simplified algebraic certificate (subgroup membership mod K, or noPREF) is necessary and sufficient to guarantee the existence of perfect commuting operator strategies, and consequently a simple class of tensor product strategies (MERP) can be shown to be optimal. Yet all parts of this example, including MERP strategies, the noPREF condition, and MERP-PREF duality, were constructed in a very ad-hoc manner. Is there a more general mathematical principle that could have guided us to this example? And can similar techniques be applied to other nonlocal games? Answer these questions may lead to progress on the XOR questions discussed above, and/or advance the study of nonlocal games generally.

A final set of open questions concern potential applications of the games and strategies developed in this thesis. One example of this is short depth circuits, where measurements similar to MERP measurements have been used to prove a short depth circuit separation [67]. More generally, XOR games have played a key role in results in both foundational physics [52] and theoretical computer science [53]. It is likely that XOR games, and nonlocal games generally will be at the center of many advances yet to come.

Bibliography

[1] Jonathan Barrett, Daniel Collins, Lucien Hardy, Adrian Kent, and Sandu Popescu. Quantum nonlocality, bell inequalities, and the memory loophole. *Physical Review A*, 66(4):042111, 2002. [p. 25]

[2] Jonathan Barrett, Lucien Hardy, and Adrian Kent. No signaling and quantum key distribution. *Physical review letters*, 95(1):010503, 2005. [p. 70]

[3] Dave Bayer and Persi Diaconis. Trailing the dovetail shuffle to its lair. *The Annals of Applied Probability*, 2(2):294–313, 1992. [p. 104]

[4] John S Bell. On the einstein podolsky rosen paradox. *Physics Physique Fizika*, 1(3):195, 1964. [pp. 17, 24]

[5] Sergey Bravyi, David Gosset, and Robert König. Quantum advantage with shallow circuits. *Science*, 362(6412):308–311, 2018. [p. 18]

[6] Jop Briet, Harry Buhrman, Troy Lee, and Thomas Vidick. Multiplayer xor games and quantum communication complexity with clique-wise entanglement. *arXiv preprint arXiv:0911.4007*, 2009. [p. 128]

[7] Jop Briët and Thomas Vidick. Explicit lower and upper bounds on the entangled value of multiplayer xor games. *Communications in Mathematical Physics*, 321(1):181–207, 2013. [p. 127]

[8] Jop Briët and Thomas Vidick. Explicit lower and upper bounds on the entangled value of multiplayer XOR games. *Communications in Mathematical Physics*, 321(1):181–207, 2013. [p. 193]

[9] Jaka Cimpric, Bill Helton, Scott McCullough, and Christopher Nelson. A non-commutative real nullstellensatz corresponds to a non-commutative real ideal; algorithms. *arXiv preprint arXiv:1105.4150*, 2011. [pp. 54, 55, 56, 57]

[10] Jakob Cimprič, J William Helton, Igor Klep, Scott McCullough, and Christopher Nelson. On real one-sided ideals in a free algebra. *Journal of Pure and Applied Algebra*, 218(2):269–284, 2014. [p. 54]

[11] Boris S Cirel'son. Quantum generalizations of Bell's inequality. *Letters in Mathematical Physics*, 4(2):93–100, 1980. [p. 71]

[12] John F Clauser, Michael A Horne, Abner Shimony, and Richard A Holt. Proposed experiment to test local hidden-variable theories. *Physical review letters*, 23(15):880, 1969. [pp. 24, 33]

[13] Richard Cleve, Peter Hoyer, Benjamin Toner, and John Watrous. Consequences and limits of nonlocal strategies. In *CCC '04*, pages 236–249, 2004. [p. 70]

[14] Richard Cleve, Li Liu, and William Slofstra. Perfect commuting-operator strategies for linear system games. *Journal of Mathematical Physics*, 58(1):012202, 2017. [pp. 137, 140, 142]

[15] Richard Cleve and Rajat Mittal. Characterization of binary constraint system games. In *International Colloquium on Automata, Languages, and Programming*, pages 320–331. Springer, 2014. [pp. 80, 94]

[16] Andrea Coladangelo and Jalex Stark. Unconditional separation of finite and infinite-dimensional quantum correlations. *arXiv preprint arXiv:1804.05116*, 2018. [p. 24]

[17] Roger Colbeck. *Quantum And Relativistic Protocols For Secure Multi-Party Computation*. PhD thesis, University of Cambridge, 2006. [p. 70]

[18] BIG Bell Test Collaboration et al. Challenging local realism with human choices. *Nature*, 557(7704):212–216, 2018. [p. 18]

[19] Andrew C Doherty, Yeong-Cherng Liang, Ben Toner, and Stephanie Wehner. The quantum moment problem and bounds on entangled multi-prover games. In *2008 23rd Annual IEEE Conference on Computational Complexity*, pages 199–210. IEEE, 2008. [pp. 31, 69]

[20] Andrew C. Doherty, Yeong-Cherng Liang, Ben Toner, and Stephanie Wehner. The quantum moment problem and bounds on entangled multi-prover games. In *CCC '08*, pages 199–210, 2008. [pp. 71, 73, 90]

[21] Olivier Dubois and Jacques Mandler. The 3-XORSAT threshold. *Comptes Rendus Mathématique*, 335(11):963–966, 2002. [pp. 72, 207, 209]

[22] Artur K Ekert. Quantum cryptography based on Bell's theorem. *Physical review letters*, 67(6):661, 1991. [p. 70]

[23] Andrew S Friedman, Alan H Guth, Michael JW Hall, David I Kaiser, and Jason Gallicchio. Relaxed bell inequalities with arbitrary measurement dependence for each observer. *Physical Review A*, 99(1):012121, 2019. [p. 18]

[24] Jason Gallicchio, Andrew S Friedman, and David I Kaiser. Testing bell's inequality with cosmic photons: Closing the setting-independence loophole. *Physical review letters*, 112(11):110405, 2014. [p. 18]

[25] Jingliang Gao. Quantum union bounds for sequential projective measurements. *Physical Review A*, 92(5):052331, 2015. [p. 95]

[26] Daniel M Greenberger, Michael A Horne, Abner Shimony, and Anton Zeilinger. Bell's theorem without inequalities. *American Journal of Physics*, 58(12):1131–1143, 1990. [pp. 75, 113]

[27] Dima Grigoriev. Linear lower bound on degrees of Positivstellensatz calculus proofs for the parity. *Theoretical Computer Science*, 259:613–622, 2001. [pp. 81, 89, 91, 92, 207, 217, 222]

[28] Michael JW Hall. Local deterministic model of singlet state correlations based on relaxing measurement independence. *Physical review letters*, 105(25):250404, 2010. [p. 18]

[29] Johannes Handsteiner, Andrew S Friedman, Dominik Rauch, Jason Gallicchio, Bo Liu, Hannes Hosp, Johannes Kofler, David Bricher, Matthias Fink, Calvin Leung, et al. Cosmic bell test: measurement settings from milky way stars. *Physical review letters*, 118(6):060401, 2017. [p. 18]

[30] Aram Harrow, Anand Natarajan, and Xiaodi Wu. Limitations of semidefinite programs for separable states and entangled games. 2016. [p. 193]

[31] Johan Håstad. Some optimal inapproximability results. *Journal of the ACM (JACM)*, 48(4):798–859, 2001. [p. 78]

[32] J Helton and Scott McCullough. A positivstellensatz for non-commutative polynomials. *Transactions of the American Mathematical Society*, 356(9):3721–3737, 2004. [pp. 31, 69]

[33] J William Helton, Scott McCullough, and Mihai Putinar. Strong majorization in a free *-algebra. *Mathematische Zeitschrift*, 255(3):579–596, 2007. [p. 53]

[34] William Helton, Kyle P Meyer, Vern I Paulsen, and Matthew Satriano. Algebras, synchronous games and chromatic numbers of graphs. *arXiv preprint arXiv:1703.00960*, 2017. [pp. 143, 229]

[35] Zhengfeng Ji. Classical verification of quantum proofs. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing*, pages 885–898. ACM, 2016. [pp. 70, 71]

[36] Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen. Mip*= re. *arXiv preprint arXiv:2001.04383*, 2020. [pp. 23, 24, 35, 229]

[37] Tony Johansson. The giant component of the random bipartite graph. Master's thesis, Chalmers University of Technology, 2012. [p. 210]

[38] David I Kaiser. Tackling loopholes in experimental tests of bell's inequality. *arXiv preprint arXiv:2011.09296*, 2020. [p. 18]

[39] Julia Kempe, Oded Regev, and Ben Toner. Unique games with entangled provers are easy. *SIAM Journal on Computing*, 39(7):3207–3229, 2010. [p. 72]

[40] Calvin Leung, Amy Brown, Hien Nguyen, Andrew S Friedman, David I Kaiser, and Jason Gallicchio. Astronomical random numbers for quantum foundations experiments. *Physical Review A*, 97(4):042120, 2018. [p. 18]

[41] Ming-Han Li, Cheng Wu, Yanbao Zhang, Wen-Zhao Liu, Bing Bai, Yang Liu, Weijun Zhang, Qi Zhao, Hao Li, Zhen Wang, et al. Test of local realism into the past without detection and locality loopholes. *Physical review letters*, 121(8):080404, 2018. [p. 18]

[42] Markus Lohrey. The rational subset membership problem for groups: a survey. In *Groups St Andrews*, volume 422, pages 368–389, 2013. [p. 188]

[43] Klaus Madlener and Birgit Reinert. String rewriting and gröbner bases—a general approach to monoid and group rings. In *Symbolic rewriting techniques*, pages 127–180. Springer, 1998. [pp. 65, 66]

[44] Ralph McKenzie and Richard J Thompson. An elementary construction of unsolvable word problems in group theory. In *Studies in Logic and the Foundations of Mathematics*, volume 71, pages 457–478. Elsevier, 1973. [p. 36]

[45] KA Mikhailova. The occurrence problem for direct products of groups. *Matematicheskii Sbornik*, 112(2):241–251, 1966. [pp. 131, 143]

[46] Miguel Navascués, Stefano Pironio, and Antonio Acín. A convergent hierarchy of semidefinite programs characterizing the set of quantum correlations. *New J. Phys.*, 10(7):073013, 2008. [pp. 31, 32, 34, 71, 73, 90, 94]

[47] Miguel Navascués, Stefano Pironio, and Antonio Acín. A convergent hierarchy of semidefinite programs characterizing the set of quantum correlations. *New Journal of Physics*, 10(7):073013, 2008. [p. 129]

[48] Petr Sergeevich Novikov. Algorithmic unsolvability of the word problem in group theory. *Journal of Symbolic Logic*, 23(1), 1958. [p. 36]

[49] Vern I Paulsen, Simone Severini, Daniel Stahlke, Ivan G Todorov, and Andreas Winter. Estimating quantum chromatic numbers. *Journal of Functional Analysis*, 270(6):2188–2222, 2016. [p. 229]

[50] David Pérez-García, Michael M Wolf, Carlos Palazuelos, Ignacio Villanueva, and Marius Junge. Unbounded violation of tripartite bell inequalities. *Communications in Mathematical Physics*, 279(2):455–486, 2008. [pp. 127, 128, 136]

[51] Boris Pittel and Gregory B Sorkin. The satisfiability threshold for k-xorsat. *Combinatorics, Probability and Computing*, 25(2):236–268, 2016. [pp. 207, 209]

[52] Dominik Rauch, Johannes Handsteiner, Armin Hochrainer, Jason Gallicchio, Andrew S Friedman, Calvin Leung, Bo Liu, Lukas Bulla, Sebastian Ecker, Fabian Steinlechner, et al. Cosmic bell test using random measurement settings from high-redshift quasars. *Physical Review Letters*, 121(8):080403, 2018. [pp. 18, 230]

[53] Ben W Reichardt, Falk Unger, and Umesh Vazirani. Classical command of quantum systems. *Nature*, 496(7446):456–460, 2013. [pp. 18, 230]

[54] Ben W Reichardt, Falk Unger, and Umesh Vazirani. Classical command of quantum systems. *Nature*, 496(7446):456–460, 2013. [p. 70]

[55] NS Romanovskii. Some algorithmic problems for solvable groups. *Algebra and Logic*, 13(1):13–16, 1974. [p. 188]

[56] Yurii Savchuk and Konrad Schmüdgen. Unbounded induced representations of *-algebras. *Algebras and Representation Theory*, 16(2):309–376, 2013. [pp. 59, 63]

[57] Volkher B Scholz and Reinhard F Werner. Tsirelson's problem. *arXiv preprint arXiv:0812.4305*, 2008. [pp. 22, 23]

[58] Alexander Schrijver. *Theory of Linear and Integer Programming*. Wiley, Chichester, 1986. [pp. 112, 120]

[59] William Slofstra. Tsirelson's problem and an embedding theorem for groups arising from non-local games, 2016. [pp. 71, 72]

[60] William Slofstra. The set of quantum correlations is not closed. In *Forum of Mathematics, Pi*, volume 7. Cambridge University Press, 2019. [pp. 20, 22, 24]

[61] William Slofstra. Tsirelson's problem and an embedding theorem for groups arising from non-local games. *Journal of the American Mathematical Society*, 33(1):1–56, 2020. [p. 144]

[62] R. P. Stanley. *Enumerative Combinatorics, vol. 2*. Cambridge University Press, 1999. Exercise 6.36 and references therein. [p. 217]

[63] Boris S Tsirel'son. Quantum analogues of the Bell inequalities. The case of two spatially separated domains. *Journal of Mathematical Sciences*, 36(4):557–570, 1987. [pp. 29, 34, 71]

[64] Umesh Vazirani and Thomas Vidick. Fully device-independent quantum key distribution. *Physical review letters*, 113(14):140501, 2014. [p. 70]

[65] Thomas Vidick. Three-player entangled XOR games are NP-hard to approximate. In *Proceedings of the 2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, FOCS '13, pages 766–775. IEEE Computer Society, 2013. [pp. 71, 72]

[66] Adam Bene Watts, Aram W Harrow, Gurtej Kanwar, and Anand Natarajan. Algorithms, bounds, and strategies for entangled xor games. *arXiv preprint arXiv:1801.00821*, 2018. [pp. 129, 130, 133, 183, 184]

[67] Adam Bene Watts, Robin Kothari, Luke Schaeffer, and Avishay Tal. Exponential separation between shallow quantum circuits and unbounded fan-in shallow classical circuits. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 515–526, 2019. [pp. 18, 230]