2012

Nontraditional Positional Games: New methods and boards for playing Tic-Tac-Toe

Mary Jennifer Riegel
The University of Montana

Nontraditional Positional Games: New methods and boards for playing Tic-Tac-Toe

By

Mary Jennifer Riegel

B.A., Whitman College, Walla Walla, WA, 2006
M.A., The University of Montana, Missoula, MT, 2008

Dissertation

presented in partial fulfillment of the requirements
for the degree of

Doctorate of Philosophy
in Mathematics

The University of Montana
Missoula, MT

May 2012

Approved by:

Sandy Ross, Associate Dean of the Graduate School
Graduate School

Dr. Jennifer McNulty, Chair
Mathematical Sciences

Dr. Mark Kayll
Mathematical Sciences

Dr. George McRae
Mathematical Sciences

Dr. Nikolaus Vonessen
Mathematical Sciences

Dr. Michael Rosulek
Computer Science

Riegel, Mary J., Doctorate of Philosophy, May 2012 Mathematics

Nontraditional Positional Games: New methods and boards for playing Tic-Tac-Toe

Committee Chair: Jennifer McNulty, Ph.D.

In this dissertation we explore variations on Tic-Tac-Toe. We consider positional games played using a new type of move called a hop. A hop involves two parts: move and replace. In a hop the positions occupied by both players will change: one will move a piece to a new position and one will gain a piece in play. We play hop-positional games on the traditional Tic-Tac-Toe board, on the finite planes $AG(2,q)$ and $PG(2,q)$ as well as on a new class of boards which we call nested boards. A nested board is created by replacing the points of one board with copies of a second board. We also consider the traditional positional game played on nested boards where players alternately occupy open positions. We prove that the second player has a drawing strategy playing the hop-positional game on $AG(2,q)$ for $q \geq 5$ as well as on $PG(2,q)$ for $q \geq 3$. Moreover we provide an explicit strategy for the second player involving weight functions. For four classes of nested boards we provide a strategy and thresholds for the second player to force a draw playing a traditional positional game as well as the new hop-positional game. For example we show that the second player has a drawing strategy playing on the nested board $[AG(2,q_1) : PG(2,q_2)]$ for all $q_2 \geq 7$. Other bounds are also considered for this and other classes of nested boards.

Acknowledgments

I would like to thank the people who helped me along the way to completing this dissertation. To my advisor Dr. Jenny McNulty; thank you for your contributions to this work. To the Department of Mathematical Science, Faculty, Students, and Staff; I would not have finished had it not been for your support.

Most importantly, thank you to my family, immediate and extended. This process has been a roller coaster of emotions. I can't begin to thank you for you encouragement, patience, and support. Thank you for believing even when I did not.

Notation

$(E, \mathcal{W})$	The board with point set E and winning sets $\mathcal{W}$
$AG(2, q)$	The affine plane of order q
$PG(2, q)$	The projective plane of order q
$\sigma_i = [\mathcal{X}_i, \{O_1, \ldots, O_{i-1}\}, \alpha]$	The game state prior to Olivia's i^{th} turn
$\mathcal{X}_i$	The i positions occupied by Xavier at game state σ_i
$\mathcal{O}_i = \{O_1, O_2, \ldots, O_{i-1}, \alpha\}$	The i positions occupied by Olivia at game state σ_i
α	The position of the newest O on the board
$\sigma'_i = [\mathcal{X}_i \cup \alpha, \{O_1, \ldots, O_{i-1}, O_i\}]$	The game state after Olivia's i^{th} turn
$\mathcal{X}'_i = \mathcal{X}_i \cup \{\alpha\}$	The $i + 1$ positions occupied by Xavier at game state σ'_i
$\mathcal{O}'_i = \{O_1, O_2, \ldots, O_{i-1}, O_i\}$	The i positions occupied by Olivia at game state σ'_i
σ_∞	The game state at the end of the game, that is a game state in which the board is full or in which one of the players has claimed a wining line
$w_\alpha(\sigma_i)$	The α-weight of the game state σ_i
$w_\alpha(q \mid \sigma_i)$	The α-weight of the point q at game state σ_i
$w_\alpha(p, q \mid \sigma_i)$	The α-weight of the pair of points $\{p, q\}$ at game state σ_i
$w_\beta(\sigma)$	The β-weight of the game state σ
$w_\beta(q \mid \sigma)$	The β-weight of the point q at game state σ
$w_\beta(p, q \mid \sigma)$	The β-weight of the pair of points $\{p, q\}$ at game state σ
$[\mathcal{M}_1 : \mathcal{M}_2]$	The nested board where $\mathcal{M}_1 = (E_1, \mathcal{W}_1)$ is the outer component board and $\mathcal{M}_2 = (E_2, \mathcal{W}_2)$ is the inner component board
$[\mathcal{W}_1 : \mathcal{W}_2]$	The collection of all of the winning sets of the nested board $[\mathcal{M}_1 : \mathcal{M}_2]$
$W \times (W_i)_{i=1}^{\lvert W \rvert}$	A winning set of the nested board $[\mathcal{M}_1 : \mathcal{M}_2]$, where $W \in \mathcal{W}_1$ and $W_i \in \mathcal{W}_2$ for $i = 1, \ldots, \lvert W \rvert$

Contents

List of Figures

Chapter 1

Introduction

In this dissertation we will be exploring new variations on positional games. We consider positional games played using a new type of move called a hop. A hop involves two parts: move and replace. In a hop the positions occupied by both players will change: one will move a piece to a new position and one will gain a piece in play. We play hop-positional games on the traditional Tic-Tac-Toe board, on the finite planes $AG(2,q)$ and $PG(2,q)$, as well as on a new class of boards which we call nested boards. A nested board is created by replacing the points of one board with copies of a second board. We also consider the traditional positional game played on nested boards where players alternately occupy open positions.

Before we begin considering these variations, we briefly provide an introduction to the material. In the first section we will introduce positional game theory and some of its vocabulary. This will be followed with a brief discussion of some of the more common variations one might see in exploring positional games in Section 2. Section 3 contains a quick review of basic facts about affine and projective planes over finite fields. In Section 4 we introduce the main variation we will consider throughout this dissertation. An outline of the major results of the thesis will be presented in Section 5.

1.1 Positional Game Theory

A combinatorial game is a specific type of game played by two players who alternate turns. Combinatorial games have many common characteristics. Well known examples of combinatorial games include NIM, Tic-Tac-Toe, and chess. The two players in any combinatorial game have perfect information about the state of the game; that is, they have knowledge of the board and all of the pieces in play at all times. The game Battleship™ is an example of a game between two players which is not a combinatorial game as neither player plays with perfect information about the board. A move in a combinatorial game is the action of a player on his or her turn, and is defined by the rules of the game. In this type of game the rules may allow the different players to make different moves. Such games are referred to as *biased* or *partizan* games. An example of a partizan game is Domineering where one player places dominoes on the board vertically and the other player places them horizontally. Games where both players have the same options on each turn are called *unbiased* games. The game NIM is an example of an unbiased combinatorial game. In NIM, players alternately remove any number of counters from a single pile attempting to be the last to remove a game piece. The players in a combinatorial game are also not allowed to move repeatedly through a sequence of identical positions. In chess this restriction is referred to as the threefold repetition rule which states that a player can claim a draw if the same position occurs three times during a game. Without this rule, chess would not be a combinatorial game. Finally, no element of chance is involved in playing a combinatorial game. Poker for example is not a combinatorial game as the dealing of cards involves chance. Poker also violates the condition on perfect information. In this dissertation, we explore only finite combinatorial games, namely games that terminate in a finite number of moves. The games we consider will also be unbiased.

Combinatorial games come in two flavors: NIM-like games and Tic-Tac-Toe-like games. NIM-like games are played until some player has no moves remaining. The last player to play is declared the winner under normal play. One defining feature of NIM-like games is that they break down into smaller and simpler subgames over the course of play. Another important characteristic of NIM-like games is that someone always wins; that is, a draw is not an allowed outcome. NIM-like games are discussed in great detail in the books *Winning Ways for your Mathematical Plays* by Berlekamp, Conway, and Guy [8] and *Lessons in Play* by Albert, Nowakowski and Wolfe [12].

Tic-Tac-Toe-like games are combinatorial games which cannot be subdivided into smaller games over the course of play. These games do not necessarily end because someone cannot play, but rather because a certain configuration or position has been achieved. For example, a game of Tic-Tac-Toe ends either because someone claimed three positions in a row (a winning configuration for that player) or because the board filled and neither player won (a drawn configuration). Tic-Tac-Toe-like games are also referred to as positional games, and are treated in-depth in the book *Combinatorial Games: Tic-Tac-Toe Theory* by Beck [2]. Tic-Tac-Toe, checkers, and chess are all combinatorial games which fall into the sub-classification positional games. All of the games we consider in this dissertation are positional games.

Formally, a positional game requires the following: a finite set E, the *points* or *positions* of the board, and some arbitrary family $\mathcal{W}$ of subsets of E called the *winning sets*, which are defined prior to the start of the game. The *game board* is the set of points E along with the collection $\mathcal{W}$; it will be denoted $(E, \mathcal{W})$. The two players, the first and second player, alternately occupy previously unoccupied or *open* points of the board. For clarity of language, we will assume the game is played by the two players Xavier and Olivia. Xavier will always play first; he will alternate turns with Olivia. The board, along with the sets of positions $\mathcal{X}$ and $\mathcal{O}$ occupied by Xavier and Olivia respectively at any point during the game is referred to as a game *state*. In any figure, unless otherwise indicated, positions occupied by Xavier will be marked by an X and those occupied by Olivia by an O. The first player to occupy a winning set $A \in \mathcal{W}$ wins. If the board is full (all positions are occupied) and neither player occupies a winning set, then the game ends in a *draw*. We refer to such a game state as a *drawn state*. A board $(E, \mathcal{W})$ is said to *admit a drawn state* if there exists a partition of E into $\mathcal{X}$ and $\mathcal{O}$ so that every winning set $A \in \mathcal{W}$ has at least one point in $\mathcal{X}$ and at least one point in $\mathcal{O}$. If a board admits a drawn state, that does not mean that a game on that board will always end in a draw, rather it allows for the possibility of a draw. On the other hand, if a board does not admit a drawn state, then the game must end with some player winning.

In combinatorial game theory, a *strategy* is an algorithm which selects the next move of a player based on the current state of the game. An *optimal strategy* for a player is one that produces a move for every game state guaranteed to result in the best possible outcome for that player. In some cases an optimal strategy is a winning strategy. A winning strategy for Xavier is a strategy which results in a win for him given any sequence of moves by Olivia. A strategy does not lay out every move in a game in advance, rather it selects the next move based on the sequence of moves leading up to the current game state.

In combinatorial game theory, an initial question that is asked when studying a specific game is: which player, if any, has an advantage? In answering this question we determine what is called the outcome class of the game. In terms of game strategies, we want to determine what optimal strategies exist. We say a player *plays optimally* if he/she uses his/her optimal strategy without error. We should note that the existence of some optimal strategies precludes the existence of others. For example, if Xavier has a winning strategy, then playing optimally he is guaranteed a win. This means that Olivia can have neither a winning nor a drawing strategy. Similarly if Olivia has a winning strategy and plays optimally, then Xavier can neither win, nor force a draw. If, on the other hand, neither player has a winning strategy, then both players must have drawing strategies. We should also note that just because Xavier has a winning strategy does not mean that the first player will always win. If the first player either makes a mistake or fails to implement the winning strategy it is possible for a game to end in a draw or with the second player winning, however in this case the first player will not have played optimally. We use the expression *first player win* to describe a game in which the first player has a winning strategy which under optimal play will result in the first player winning.

Given a positional game played optimally by both players, there are three conceivable outcomes of the game: the first player has a winning strategy; the second player has a winning strategy; or both players have a drawing strategy. However, it turns out that the second player cannot actually have a winning strategy. That is, despite the second player occasionally winning at Tic-Tac-Toe, no strategy exists which can guarantee such an outcome against a perfect first player. This is formalized in the following theorem, which is proved using a *strategy stealing* argument (see e.g., [2]).

Theorem 1.1.1. *In a positional game, Xavier can always force at least a draw; that is, either Xavier has a winning strategy or both players have a drawing strategy.*

Proof. Suppose that Olivia has a winning strategy S. We will obtain a contradiction by demonstrating how Xavier can win by 'stealing' S. Let Xavier first make an arbitrary move. Ignoring this game piece, Xavier will pretend to be the second player for the rest of the game by implementing Olivia's strategy S. If on his turn Xavier is told by S to make a move still available, then he should do so. If the move was taken by him as his arbitrary first move, then he should make another arbitrary move. As any such arbitrary move by Xavier results in one more position with an X, such a move can only benefit Xavier and thus does not change the effectiveness of the strategy S. It follows that Xavier can win by following the 'stolen' strategy S. This contradicts that S was a winning strategy for Olivia playing second as both players cannot win. Therefore, Olivia doers not have a winning strategy, and Xavier can at least force a draw. □

We should note that, in a positional game, Xavier will always have an available first move unless the board has no open positions. However, in the case that the board has no open positions Olivia cannot have a winning strategy S, and we would say the game ends in a draw. This is the only situation in which Xavier would not be able to make an opening move, as our positional games have the same rules of play for both Xavier and Olivia. That is, there are no boards for positional games in which Xavier has no opening move, but Olivia does. Given the second player cannot have a winning strategy, the best Olivia can do is have a drawing strategy which would prevent Xavier from having a winning strategy. Thus, determining the outcome class of a positional game is reduced to answering the question "Does Xavier have a winning strategy?" Again, if he does we refer to the game as a first player win game. If Xavier does not have a winning strategy, then both players have a drawing strategy. As this is the best possible outcome for Olivia we refer to such games as second player draw games.

1.2 Variations on Tic-Tac-Toe

While games like checkers and chess are prototypical examples of positional games, much of the research into the field actually involves variations on Tic-Tac-Toe due to the complexity of these other games. Tic-Tac-Toe is and example of a *static* game. Checkers and chess, on the other hand, are examples of *dynamic* games in which pieces can be relocated or removed from the board. The relative simplicity of Tic-Tac-Toe makes it an ideal candidate for mathematical inquiry. In this section we briefly explore some variations on Tic-Tac-Toe before we move on to the specific variations which are the focus of this dissertation. In all of the variations discussed in this section the two players alternately place pieces in open positions, that is play the static or traditional positional game of Tic-Tac-Toe.

As stated above, the goal in a Tic-Tac-Toe-like game is to occupy a winning set first. A win under these rules is also called a *strong win*. In general determining how to create a strong win is hard. For example, the $4 \times 4 \times 4$ Tic-Tac-Toe game is known to be a first player win, but the winning strategy is extremely complicated, requiring a computer assisted proof. The $5 \times 5 \times 5$ game is expected to be a draw, but the 3^{125} step backtracking algorithm needed to prove it is computationally intractable to attempt.

The first, and most common variation to consider is that of the Maker-Breaker game, which removes the requirement to win first. Instead, in this version the first player, Maker, attempts to complete a winning set while the second player, Breaker, attempts to prevent it. In the strong win game both players must 'build' and 'block' at the same time. In the Maker-Breaker game, the two jobs are separate, which makes the analysis somewhat easier. In Maker-Breaker games unlike strong games, someone always wins: maker if he completes a winning set and breaker if he does not. For some games, changing to the Maker-Breaker version does not help at all. For example, in the game Hex each player is attempting to create a connected path from one side of a board to the opposite side. The only way to block your opponent from winning is to win yourself as no Hex board admits a draw. Thus the strategy for each player is still to win, even in the Maker-Breaker case. However, there are some instances where playing under Maker-Breaker rules does aid in analysis. For example one can check that when playing Tic-Tac-Toe under Maker-Breaker rules, by not having to block the second player, the first player can actually win. The ability to win the Maker-Breaker game but not the original game is one reason Maker-Breaker wins are also called *weak wins*.

The Shannon Switching game is an example of a Maker-Breaker game. The game is played on the edges of a graph with Maker's goal being to create a path connecting two distinguished vertices of the graph. Breaker can prevent a win by completing a cut separating the vertices. In "A Solution of the Shannon Switching Game", Lehman [11] proved that Maker has a winning strategy when playing on the edges of a complete graph attempting to complete a spanning tree (thus creating a path between any two vertices). He also proves necessary conditions on a general graph for Breaker to have a winning strategy in the Shannon Switching Game.

Having made the switch to Maker-Breaker games, may authors are able to make progress in determining game outcomes on various boards using a modified probabilistic method. For example consider a Tic-Tac-Toe game played on the d-dimensional integer lattice where a winning sets is a collection of m consecutive points on a line. In the paper "Potential-Based Strategies for Tic-Tac-Toe on the Integer Lattice with Numerous Directions" Kruczek and Sundberg [10] identified a bound in the size m of a winning set for Breaker to win if the allowed directions of winning sets is limited. The arguments in the paper rely on a modification of the probabilistic method first developed by Erdös and Selfridge [9] and later modified by Beck [2].

Other variations of Tic-Tac-Toe played on graphs involve changing the winning set structure. For example, in the paper "A Sharp Threshold for the Hamilton Cycle MakerBreaker Game," Hefetz et. al. explore a Maker-Breaker game played on random graphs. The idea is that given a random graph $G(n, p)$ with n vertices where each edge is independently included with probability p, two players play on the edges of the graph with Maker attempting to complete a Hamilton Cycle. The results of this paper rely on earlier results which show that for p sufficiently large with respect to n, a random graph $G(n, p)$ is almost surely Hamiltonian. The main result from this paper is as follows.

Theorem 1.2.1 ([7] Theorem 1)**.** *There exists a constant $l > 0$ such that the Hamilton cycle game on $G(n, \frac{\log n + (\log\log n)^l}{n})$ is almost surely a Maker win.*

We should note that the strategies developed when using the probabilistic method are deterministic. The probabilistic nature is the existence of the board, not the strategy. Once such a board exists the strategy will determine the outcome of the game.

Other variations include the biased game in which the two player are allowed to place a different number of pieces on the board. For example a game might be played where Maker is allowed to place 1 piece and Breaker q pieces. The goal of this type of analysis is to determine the threshold value of q at which the game switches from Maker win to Breaker win. For example, in the paper "Biased Positional Games on Matroids" Bednarska and Pikhurko [3] play such a game where Makers goal is to claim a circuit of the matroid. In the paper they prove threshold values for q based on the rank structure of the given matroid. For example, if one plays on the cycle matroid of the complete graph K_n, then Maker wins the game for all $q < \lceil n/2 \rceil - 1$. ([3] Corollary 10). They show in fact that this bound holds for either the Maker-Breaker or the Breaker-Maker game where in each case Maker places 1 and Breaker q pieces.

One final variation is the Chooser-Picker game. This game functions in the same ways as the classic compromise of one person cuts and the other chooses. The Picker selects two positions and then the Chooser selects which one to keep with the other going to the Picker. In the Chooser-Picker version the Chooser is playing as the Maker while in the Picker-Chooser, the Chooser plays as the breaker. Such games are considered in the paper "On Chooser Picker positional games" where Csernenszky, Mándity, and Pluhár [1] [1] prove among other things that when playing so that Maker is trying to complete a base of a matroid, Picker wins the Picker-Chooser game if and only if there exist two disjoint bases. ([1] Theorem 2). For more variations on Tic-Tac-Toe we recommend Beck's book [2] on the subject.

The variation we will introduce in Section 1.4 is different from these variations in that the game played using the rule is an example of a dynamic game like checker or chess. We will be examining the strong win version of the game where both players attempt to win first. We should not that the game could be modified to be played as a Maker-Breaker game, but no such analysis will be included in this dissertation.

1.3 Finite Planes AG(2, q) and PG(2, q)

The next section introduces a move variation we will use to play a positional game, both on $AG(2,q)$ and $PG(2,q)$ in Chapter 3 and on a new class of boards constructed from them in Chapter 5. We review here relevant properties of the geometry of these planes, which can be found in any text on finite geometries or matroids, e.g., [13–15].

An *affine plane* is an ordered pair (P,L) where P is a set of *points* and L is a collection of subsets of P called *lines* such that each of the following conditions holds.

(A1) Two distinct points are on exactly one line.

(A2) If l is a line and x a point not on l, there is exactly one line through x which does not intersect l.

(A3) There exist four points, no three of which are collinear.

A *projective plane* is an ordered pair (P,L) where P is a set of *points* and L is a collection of subsets of P called *lines* such that each of the following conditions holds.

(P1) Any two distinct points are on exactly one line.

(P2) Any two distinct lines intersect at a unique point.

(P3) There exist four points, no three of which are collinear.

In particular, we will be concerned with special finite affine and projective planes denoted $AG(2, q)$ and $PG(2, q)$ respectively. These particular finite planes can be viewed in terms of vector spaces using the following constructions. For a prime power q, let $\mathbb{F}_q$ be the finite field of order q, and consider the $\mathbb{F}_q$-vector space $\mathbb{F}_q^2$ of dimension 2. The affine plane $AG(2, q)$ has as its point and line sets the points and lines respectively of $\mathbb{F}_q^2$. We should note that as the vector space $\mathbb{F}_q^2$ satisfies the axioms of an affine plane listed above, the specific planes $AG(2, q)$ are in fact affine planes. We leave too the reader to verify that the structure of $AG(2, q)$ satisfies the axioms (A1), (A3), and (A3).

Given this construction, we have in the affine plane $AG(2, q)$ of order q:

1. there are q^2 points and $q^2 + q$ lines;
2. every line contains q points;
3. every point is contained in $q + 1$ lines; and
4. there are $q + 1$ parallel classes each containing q lines.

To get the projective plane of order q we consider the vector space $\mathbb{F}_q^3$, and let the points of $PG(2, q)$ correspond to the lines through the origin in $\mathbb{F}_q^3$ (vector subspaces of dimension 1). The lines of $PG(2, q)$ correspond to the planes through the origin in $\mathbb{F}_q^3$ (vector subspaces of dimension 2). One can verify that (P1), (P2) and (P3) hold for $PG(2, q)$.

Using what we know about vector spaces, we may establish some properties of the finite projective plane $PG(2, q)$:

1. the plane contains $q^2 + q + 1$ points and $q^2 + q + 1$ lines;
2. every line contains $q + 1$ points; and
3. every point is contained in $q + 1$ lines.

The following theorem, found, e.g., in *Matroid Theory* by Welsh [14], relates affine and projective planes via a simple construction.

Theorem 1.3.1. *(a) If one line and all of the points on it are removed from a projective plane, then the remaining incidence structure is an affine plane. (b) Given any affine plane, there exists a projective plane which determines it by the construction of (a).*

Given this theorem, we see that the affine plane $AG(2,q)$ exists if and only if the projective plane $PG(2,q)$ exists. In particular, as finite fields exist for all prime powers q, we know that both the affine and projective planes exist for all prime powers. For non prime-powers, there are only partial results about the existence or non-existence of these planes. For example, in 1949 Bruck and Ryser [4] proved:

Theorem 1.3.2. *If $q \equiv 1 \pmod 4$ or $q \equiv 2 \pmod 4$ and if q is not the sum of two squares, then there is no projective plane of order q.*

This theorem says in particular that no projective plane of order 6 exists as 6 is not the sum of two squares. It follows that no affine plane of order 6 exists by the connection between the two types of planes. We will use this fact in Chapter 3. On the other hand, this result does not prove the non-existence of the plane of order 10, as $10 = 1 + 9$ is the sum of two perfect squares. The proof by Lam, Theil, and Swiercz [6] of the non-existence of such a plane uses a backtracking search by computer.

Throughout this dissertation, we study positional games played on various boards. In the next section we define a new move variation which we use to play a positional game on the variouse boards. In Chapter 3 we play on the finite planes $AG(2,q)$ and $PG(2,q)$ and in Chapter 5 on nested boards constructed from them. When playing on a finite plane, we define the board by letting E be the set of points of the plane and $\mathcal{W}$ the set of all its lines. That is, for each line in the plane we get a corresponding winning set $A \in \mathcal{W}$ containing all of the points on the line.

1.4 Game Variation: The Hop

The primary variation from traditional positional games that we explore derives from a new method of play. In a traditional game of Tic-Tac-Toe, played on any board, the move made by a player involves placing a piece in an open position of the board. We refer to this type of Tic-Tac-Toe game as a *traditional positional game*. In our variation, the two players alternately make a move called a 'hop' on the board. A *hop* consists of two parts: move and replace. These are most easily understood using an example. On Xavier's turn he will choose any of his pieces already on the board and **move** it to any open position on the board; he will also **replace** his piece with one of Olivia's in the position he vacated. Similarly, on her turn Olivia, will **move** one of her own pieces and **replace** it with one of Xavier's. We refer to a game played using the hop move as a *hop-positional game*.

Since a move by either player requires an existing piece on the board, we need a special starting rule for a hop-positional game. Since Xavier plays first, the game will always begin with one X on the board. One aspect of hop-positional games which is different from traditional positional games is that one needs to explore how the starting position affects the outcome of the game. Another is that in a hop-positional game, both players have the same number of pieces in play after any of Xavier's turns while Xavier has one more piece on the board after any turn by Olivia.

Hop-positional games also have slightly different game endings than traditional positional games. After a single hop, the positions occupied by both players change. This means that in one move both players could claim winning sets. Thus we need to clarify game endings and introduce a new outcome. A player is said to win a hop-positional game if after a hop has been completed that player has a winning set but the other player does not. Note that a player must complete both parts of the hop to claim a win. Also a player can hop and not claim a winning set, but give his or her opponent a winning set and thus a win. Just as with traditional positional games, a game ends in a draw if the board is full and neither player has a winning set. The new possible outcome is that both players may simultaneously claim a winning set. If a game progresses to a point where a single hop results in winning sets for both players, we say the game has ended in a *tie*. We stress a tie is not the same as a draw; in a tie both players win while in a draw neither player wins. A board is said to admit a tie if the points E of the board can be partitioned into $\mathcal{X}$ and $\mathcal{O}$ in such a way that both $\mathcal{X}$ and $\mathcal{O}$ contain a winning set from $\mathcal{W}$. Just as not all boards admit drawn positions so too not all boards admit ties, and a board admitting a tie does not mean that a game on that board ends in a tie.

Given the possibility of a tie, when a hop-positional game is called first player win, we show that the winning strategy results in an actual win for the first player and does not lead to a tie. We note that just as with traditional positional games, in a hop-positional game Olivia cannot have a winning strategy. If she did, then using a strategy stealing argument again Xavier would also have one, a contradiction. It follows that Olivia cannot have a winning strategy in a hop-positional game.

1.5 Statement of Results

We begin in Chapter 2 by exploring the game Hop-Tic-Tac-Toe, in which both players hop on the traditional 3×3 Tic-Tac-Toe grid. We note that under optimal play, Olivia has a drawing strategy in the traditional positional game of Tic-Tac-Toe [2]. We will see that the game outcome changes when we play Hop-Tic-Tac-Toe. In Theorem 2.1.1 we provide a winning strategy for the first player.

In Chapter 3 we explore the hop-positional game played on $PG(2,q)$ and $AG(2,q)$. The results of this chapter rely heavily on two weight functions to define the players' strategies. The results of this chapter can be summarized into the following theorems.

Theorem 3.3.4. Xavier has a winning strategy hopping first on $PG(2,2)$, while Olivia has a drawing strategy hopping second on $PG(2,q)$ with $q \geq 3$.

Theorem 3.4.4 Olivia can force a draw hopping second on $AG(2,q)$ for all $q \geq 5$.

In Chapter 4, we explore a new positional game, "Fire and Ice"™, played on a board constructed from copies of $PG(2,2)$. This game provided the inspiration for the hop move that we explore in the rest of this thesis. We prove in this chapter that the first player, FIRE, has a winning strategy and discuss the strategy.

We continue by generalizing the “Fire and Ice”™ board in Chapter 5, to create a class of nested boards and play both traditional and hop-positional games on them. We play in particular on four classes of nested boards whose components are affine or projective planes over finite fields. For each class of nested board we prove thresholds for the game to be a second player draw and provide strategies for those boards. The results for the hop-positional game played on such boards is summarized in the following theorem.

Theorem 5.3.5 Olivia’s strategy of hopping from α to the open position of highest α-weight is a drawing strategy for all nested boards $[A(2,q_1) : P(2,q_2)]$, $[P(2,q_1) : P(2,q_2)]$, $[A(2,q_1) : A(2,q_2)]$, and $[P(2,q_1) : A(2,q_2)]$ where $q_2 \geq 8$.

Finally, in Chapter 6 we include some open questions and possible future directions of study.

Chapter 2

Hopping on a traditional Tic-Tac-Toe board

In this chapter, we explore the game Hop-Tic-Tac-Toe in which Xavier and Olivia play on the traditional nine-square grid of a Tic-Tac-Toe board using the new hop move defined in Section 1.4 instead of the traditional method of placing pieces. The traditional board admits both drawn and tied states; however, we show Xavier can win Hop-Tic-Tac-Toe playing first from any of the three distinct starting positions. That is, we demonstrate that Xavier has a winning strategy in the hop-positional game played on the traditional Tic-Tac-Toe board, which we refer to as Hop-Tic-Tac-Toe.

Recall that the traditional Tic-Tac-Toe board has nine points and eight winning sets (straight lines) of three points each. Of the nine points, one is on four lines (the middle), four are on three lines (the corners), and four are on only two lines (the edges). Because of this difference in the points, we need to consider three different starting positions, namely the initial X placed in the middle, a corner, or an edge position. In each case we present a winning first move by Xavier and explore the corresponding game trees to see that Olivia can no longer force a draw in any case. We further demonstrate that Xavier can win in each case without creating a tie. Where necessary we refer to specific positions on the board as numbered in Figure 2.1.

1	**2**	**3**
4	**5**	**6**
7	**8**	**9**

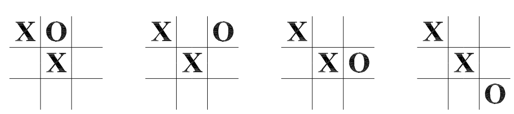
Figure 2.1: The board along with position numbers for the traditional Tic-Tac-Toe board.

Theorem 2.1.1. *Xavier has a winning strategy hopping first on a traditional Tic-Tac-Toe board from any starting position.*

Proof. Our proof is divided into three cases, corresponding to the three distinct starting positions of middle, position {5}, corner, positions {1,3,7,9}, and edge, positions {2,4,6,8}.

Starting in the Middle

Starting with the initial X in the middle position of the board, Xavier can always win by hopping to a corner on his first move. Given the symmetries of the board we assume he hops from position 5 to position 1. From this state Olivia has four distinct options, again with respect to board symmetries, as depicted in Figure (2.2). Notice that any move by Olivia results in Xavier having X's in positions 1 and 5.



A **B** **C** **D**

Figure 2.2: The four possible configurations, up to isomorphism, after Xavier and Olivia each make one hop on the board starting with an X in the middle position with Xavier hopping to any corner. We refer to these states as A, B, C, or D as indicated.

Now we address Xavier's second turn. Xavier needs to consider the following facts:

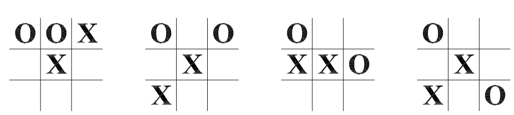
• Olivia's second turn will change one of her two O's into an X.

• After Olivia's second turn he will have three pieces to Olivia's two. That is he will have enough pieces on the board to win on his third turn.

• Parallel lines exist in the Tic-Tac-Toe board. Thus, he needs to be sure to avoid hopping to a tie on his third turn.

Xavier's winning move is to a position such that (1) his two pieces are on a line and (2) each of his two pieces is also on a line with each of Olivia's two pieces. Since the middle position is on a line with every other position on the board he should not hop away from position 5. By maintaining an X at position 5, he can be sure of satisfying the first requirement that his two $X's$ share a line. The X at position 5 will be on a line with each of the two O's after his turn. This means Xavier must move the piece from position 1. In order to determine the destination that satisfies condition (2), we note first that he should hop to a position that is on a line with position 1. By doing so he ensures that his second X is on a line with the new O after his turn. He should also hop to a position on a line with Olivia's existing O. That is, he should hop along one of the two lines containing position 1 until it intersects with a line containing Olivia's existing piece.

Suppose for example after Olivia's turn the game is in state A. Olivia's existing O is at position 2, which is on the two lines $\{1,2,3\}$ and $\{2,5,8\}$. Xavier must therefore hop to one of the open position $\{3,8\}$, while staying on a line with 1. The X at position 1 is on three lines, $\{1,2,3\}$, $\{1,4,7\}$ and $\{1,5,9\}$. According to his strategy, Xavier must hop from 1 along the line $\{1,2,3\}$ to the open position 3. An optimal move for Xavier for each of Olivia's options can be seen in Figure 2.3. Again we note that in each case, Xavier hops from position 1 so each of his pieces is on a line with each of Olivia's.



A' **B'** **C'** **D'**

Figure 2.3: An optimal second hop for Xavier from each of Olivia's options, where in each case Xavier's turn began with pieces in positions 1 and 5. He hops from 1 to a position such that each of his pieces is on a line with each of Olivia's.

By ensuring that each of his pieces is on a line with each of Olivia's he guarantees that after Olivia's turn each pair of his three pieces is on a line. After Olivia's turn either Xavier has three points on a line or his three pieces determine three different lines. As Olivia has only two pieces on the board Xavier can be sure at least one of the three lines does not contain an O. That is one of the three lines is *unblocked* by Olivia. Thus he can claim a winning line on his next turn. The question now is: can he claim a winning line without also giving Olivia a winning line? That is, can he avoid creating a tie? The board admits tied states, so we must verify that Xavier has not moved into one of them.

A tie on Xavier's turn can only result from a game position in which two parallel lines are occupied in a specific configuration: one line, call it l, contains two O's and one X, while the second line, call it m, contains two X's and one open position. In such a case Xavier could create a tie by hopping to complete line m. On his second turn, Xavier moves deliberately so that his three points prior to his third turn would determine three lines, $\{m, n, q\}$. It follows that both n and q intersect l at the position containing the X. Since both of Olivia's O's are on line l, neither n nor q is blocked (has an O), and Xavier can win by completing either line thus preventing a tie. Following this strategy, Xavier can always win Hop-Tic-Tac-Toe if the game begins with an X in the middle.

Starting in a Corner

If the game begins with an X in a corner, we may assume, given board symmetries, that it begins with an X in position 1. Xavier can win by hopping to the middle of the board on his first turn. As Olivia is forced to hop from position 1 (it is the only O), after her first turn the four possible configurations are the same as for the game beginning in the middle (Figure 2.2). Thus, following the same strategy as above, Xavier can always win Hop-Tic-Tac-Toe when the game begins in a corner.

Starting on an Edge

If the game begins at an edge, we may assume the initial X is in position 2. Xavier can win by hopping to position 4. From this game state Olivia has five distinct options for her turn as depicted in Figure 2.4. We divide the five options into two types based on the positions of the pieces relative to one another. We say a game state is of type I if each X is on a line with the O, as is the case in states EA and EB of Figure 2.4. A game state is said to be of type II if at least one X is not on a line with the O, as is the case in states EC, ED and EF of Figure 2.4.

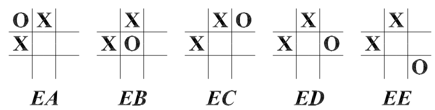


Figure 2.4: The five possible configurations after Xavier and Olivia each make one hop on the board starting with an X in the edge position 2 with Xavier playing to the adjacent edge position 4. We refer to these states as EA, EB, EC, ED, or EE as indicated to distinguish them from the states achieved in the middle and corner starting positions.

Type I:

For game states EA, EB, where both X's are on a line with the O after Olivia's first hop, Xavier should hop so that the board has two parallel horizontal lines, one with two X's (call it l) and one with two O's (call it m), leaving a vertical line completely open. In particular in EA he hops from 2 to 5 and in EB from 4 to 1. These resulting game states can be seen in Figure 2.5. By hopping in this way he has created one unblocked horizontal line with two X's. Also by leaving a vertical line open, Xavier ensures that both of Olivia's O's are on vertical lines with one of his X's. This means that after her turn Olivia needs to block not only the horizontal line l, but also a vertical line with two X's. As she cannot block both lines in one turn, Xavier can claim a line on his third turn.

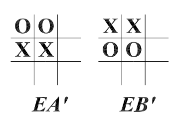
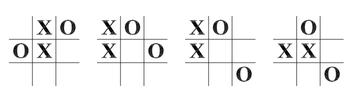


Figure 2.5: Optimal second hops for Xavier from each of Olivia's options of type I: EA and EB. He hops to create a pair of parallel horizontal lines, one with two X's and one with two O's. He also hops to leave a vertical line completely open.

Finally, we consider next the possibility of a tie in this case to ensure that Xavier's strategy results in a win. As we recall from our discussion of tying from the middle position, a tie on Xavier's turn has to come from two parallel lines: one with one X and two O's, the other with two X's and an open position. Such a game state cn be achieved in one of two ways, First, Olivia can hop on the line m which had two O's, filling it with two O's and one X. However, by doing so she has blocked neither the initial horizontal line l of two X's, nor the new vertical line with two X's which were guaranteed by Xavier's previous hop. It follows that Xavier can hop to complete the vertical line which intersects the horizontal line m, thus preventing a tie. The second way such a state can occur is if Olivia hops one of here O's to a line which previously had one O and one X, call it l, thus filling it with two O's and one X. By doing so she will have failed to block the horizontal line h with two X's. Further as this line h intersects l, Xavier can hop the piece not on h, thus completing the line without giving Olivia a win on l. It follows that Xavier's strategy prevents a tie, and therefore, Xavier has a winning strategy for game states EA and EB.

Type II:

For game states EC, ED, EE of type II as in Figure 2.4, at least one of the two X's in positions 2 and 4 is not on a line with the O. Let $\overline{X}$ be such an X. Xavier wins by moving $\overline{X}$ along one of its lines until he creates a line with two X's and an open position, call it l. For game state EC there is only one such move, hopping from position 4 to position 5 so that line $\{2,5,8\}$ has two X's. For ED only a hop from 2 to 1 creates such a line, in this case the line $\{1,4,7\}$. For game state EE there are two options up to board symmetry: to hop from 2 to 1, or from 2 to 5. The resulting four game states can be seen in Figure 2.6. After these choices for Xavier's hop, the two O's on the board do not share a line.



EC′ *ED′* *EE′* *EE″*

Figure 2.6: Optimal second hops for Xavier from each of Olivia's options EC, ED, EE. He hops along a line from a position not on a line with the O to create a line of with two X's and an open position.

In each case we see that both of Olivia's O's are on at least one line with one X, one O, and an open position; we denote this line type by $XO-$. Each of these lines necessarily intersects the line l which contains the two X's. It follows that any hop by Olivia leaves at least one unblocked line with two X's on the board, either the one Xavier created, l, or one created by the replace phase of her hop. It follows that Xavier can hop to win on his next turn. Furthermore, Xavier can do so without tying because in order for Olivia to create a potential tying position, she must hop to create a line m of type XOO (one X and two O's) as we have discussed before. As her two O's are not on a line, she can only create such a line by hopping onto one of the $XO-$ lines. However, as the line l with two X's intersects both of the $XO-$ lines, it still intersects the line m after Olivia hops, and thus, by completing the line l, Xavier can win without creating a tie. Therefore, Xavier can win if Olivia hops on her first turn to any of the game states EC, ED, or EE.

It follows that Xavier can win Hop-Tic-Tac-Toe if the game begins with an X on an edge.

We have seen that Xavier can win Hop-Tic-Tac-Toe from any starting position of the initial X. $\square$

Chapter 3

Hopping on the finite planes AG(2, q) and PG(2, q)

In this chapter we explore the hop-positional game where we use the hop move defined in Section 1.4 playing on finite projective and affine geometries, in particular on the boards $AG(2, q)$ and $PG(2, q)$. As with traditional Tic-Tac-Toe, the object of the game is for a player to gain control of a winning set on the board. When playing on a finite geometry, we define the board by letting E be the set of points of the geometry and $\mathcal{W}$ the sets corresponding to the lines of the geometry. That is, for each line in the geometry we get a corresponding winning set $A \in \mathcal{W}$ containing all of the points on the line. Given this choice, we refer to winning sets as lines. In the paper "Tic-Tac-Toe on a Finite Plane," Carroll and Dougherty [5] explore the traditional positional game in which a move by any player consists of placing a piece in an open position played on the boards $AG(2, q)$ and $PG(2, q)$. There are other boards one could define using finite geometries by making other choices for $\mathcal{W}$; however, in this dissertation we will use the lines as winning sets.

Before we begin we state some properties of the automorphism groups of our boards which can be found in any text of finite geometries or matroids (see , e.g., [13–15]). First, every finite plane $AG(2, q)$

or $PG(2,q)$ has at least a doubly transitive automorphism group. A doubly transitive automorphism group means that for any two pairs of points α, β and σ, ρ there exists an automorphism that maps $\alpha \to \sigma$ and $\beta \to \rho$. Having a transitive automorphism group means no point in any plane is distinguished. Thus all possible starting states, that is the placement of a single X on the board prior to the start, are equivalent for the purpose of the game. Therefore, unlike the situation of Chapter 2, we do not need to consider multiple starting states for any of the boards we now are concerned with. The automorphism group of these planes also behave predictable on lines. In any affine plane, $AG(2,q)$, any pair of parallel lines can be mapped via an automorphism to any other pair of parallel lines. Also in both $AG(2,q)$ and $PG(2,q)$, any pair of intersecting lines can be mapped via an automorphism to any other pair of intersecting lines. These properties of the automorphism groups of our boards will allow us to reduce the number of distinct cases to be considered for many of our boards.

3.1 Planes of small order

We begin by considering planes of small order, namely $AG(2,2), AG(2,3)$ and $PG(2,2)$. These boards are small enough to analyze by hand and are in fact too small to adhere to our more general solution of later sections.

Proposition 3.1.1. *Xavier has a winning strategy hopping on $AG(2,2)$.*

Proof. The affine plane $AG(2,2)$ consists of four points and six lines, in which any two points form a line. The points and lines of $AG(2,2)$ are depicted in Figure 3.1. Since the goal of the game is to claim an entire line on the board we see that in this case the first player to have two pieces on the board necessarily wins. The game begins with one X and no O's on the board. After Xavier's first turn there will be one X and one O on the board. Following Olivia's turn Xavier will have two pieces to her one, and thus, no matter what moves either player makes, Xavier always wins hopping on $AG(2,2)$. □

We can also see that Xavier has a winning strategy in another way. Since any two points form a line,

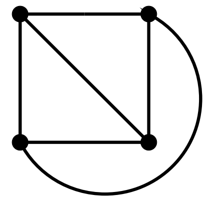


Figure 3.1: The four points of $AG(2,2)$ are arranged on six lines divided into three parallel classes of two lines each. We note that the diagonal and curved line are parallel in this affine plane.

the board $AG(2,2)$ does not admit a drawn state. To see this we note that in a partition of the four points of $AG(2,2)$ into two sets at least one player has at least two points. That player then necessarily has a winning line and no drawn state exists. Since the game cannot end in a draw we can see by a strategy stealing argument the first player must have a winning strategy. We should note here that as Xavier will have two points on the board when Olivia has only one, the game cannot end in a tie as it ends before Olivia has enough pieces on the board.

We next consider the smallest of our projective planes $PG(2,2)$. We should note that as all lines of projective planes are intersecting, we do not need to consider the possibility of ties for any of the boards $PG(2,q)$. This will not be the case for the affine planes $AG(2,q)$, where parallel classes of lines allow for the possibility of a tie.

Proposition 3.1.2. *Xavier has a winning strategy hopping on $PG(2,2)$.*

Proof. Let us consider $PG(2,2)$, the smallest projective plane. We begin by confirming that $PG(2,2)$ seen in Figure 3.2 does not admit a drawn state. That is, once the board is full and the points are partitioned into those belonging to Xavier ($\mathcal{X}$) and those belonging to Olivia ($\mathcal{O}$), some line is contained entirely in one of the two sets. Thinking of the labels X and O as colors, we can state this in another way: we prove that every 2-coloring of the points of $PG(2,2)$ contains a monochromatic line.

Consider a partition of the points of $PG(2,2)$ into two sets $\mathcal{X}$ and $\mathcal{O}$. We first note that as any two lines intersect, both $\mathcal{X}$ and $\mathcal{O}$ cannot contain a line. By the pigeonhole principle, and without loss of generality, $|\mathcal{X}| \geq 4$.

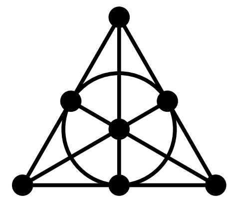


Figure 3.2: The seven points of $PG(2,2)$ form seven lines of three points each.

Suppose $|\mathcal{X}| = 4$ and $|O| = 3$. If O is a line, then we are done. We therefore assume that O is not a line. Using basic counting arguments, there are $3 \cdot \binom{3}{1} - 1 \cdot \binom{3}{2} = 6$ lines which intersect O. As there are 7 lines in $PG(2,2)$, it necessarily follows that the remaining line is contained in $\mathcal{X}$.

If $|\mathcal{X}| = 5$, the two points in O account for or block $3 \cdot \binom{2}{1} - 1 \cdot \binom{2}{2} = 5$ lines and $\mathcal{X}$ contains the two remaining lines. Since any five points necessarily contain a line so do any 6 or any 7 points.

As any full board $PG(2,2)$ contains a monochromatic line, the hop-positional game on $PG(2,2)$ cannot end in a draw. It follows by a strategy stealing argument that Xavier must have a winning strategy for hopping on this board. □

Lemma 3.1.3. *In every partition of the points of $PG(2,2)$ into two sets $\mathcal{X}$ and O, one of the two partitions contains a complete line.*

In fact, given that the board does not admit a draw we can define a winning strategy for Xavier. Xavier can play to any open position on his first two turns, the strategy will find a winning move on his third turn. Let us consider the game at Xavier's third turn. Prior to his turn, the board contains three X's and two O's. If Xavier's three X's form a line then we are done, therefore suppose this is not the case. After Xavier's turn, the board contains three X's, three $O's$ and has one open position. This is the first point at which Olivia has enough pieces on the board to claim a line and thus is the first time Xavier really needs to be concerned about which position he hops from. Let us consider the line containing Olivia's two O's prior to Xavier's turn, call it L. There are two possible cases: either L is full, also has an X, or it is not.

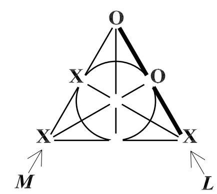


Figure 3.3: A representative game state after Olivia's second turn where the line L containing the two O's in play is full. All such configurations are isomorphic to the one pictured here. The line M determined by the two X's not on L must also be full.

Suppose L is full as in Figure 3.3. Xavier does not want to hop from his X on L, so let us consider the other two X's on the board. These two X's are on a unique line, call it M, which intersects L. Since we have assumed Xavier has not yet won, M and L intersect at one of the points occupied by Olivia. Xavier can safely hop from either position without creating a line of three O's. Furthermore each of the X's is on an unblocked line with the X on L. Thus Xavier can win on his third turn by hopping one of the X's on M to complete the line with the other two X's on the board.

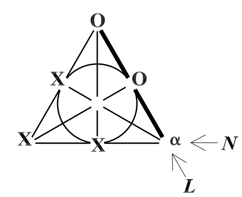


Figure 3.4: A representative game state after Olivia's second turn where the line L containing the two O's in play is not full. All such configurations are isomorphic to the one pictured here. There is a line N containing two X's and the open position α on L.

Now suppose L is not full as pictured in Figure 3.4. Let α be the open position on L. Since each point in $PG(2,2)$ is on exactly three lines, the other two lines through α are unblocked by Olivia. These two lines contain five points total including α, three of which must contain X's. That is one line through α (call it N) has two X's. If Xavier hops from the X not on N to α he claims all of the points on N and thus wins the game. We should note that by claiming a line Xavier is in effect blocking Olivia as all lines in $PG(2,2)$ are intersecting. Thus we have demonstrated a winning strategy for Xavier on $PG(2,2)$.

So far the boards we have considered $AG(2,2)$ and $PG(2,2)$, apart from being the smallest boards of each class, have the property of not admitting a drawn state. Next we consider the board $AG(2,3)$. We consider this board for several reasons. $AG(2,3)$ is the next smallest board in terms of the number of points it has (9). It is also a natural extension of the traditional Tic-Tac-Toe board. Finally it is the only remaining board from our two classes which does not admit a drawn state, and therefore is the last of the small "easy" cases.

Proposition 3.1.4. *Xavier has a winning strategy hopping first on $AG(2,3)$.*

Proof. Let's consider the game on $AG(2,3)$. The board is comprised of the same nine points as a traditional Tic-Tac-Toe board with the addition of the four new winning lines shown in Figure 3.5. The 9 points of $AG(2,3)$ are contained in 12 total lines. Each point is on exactly four lines, one in each

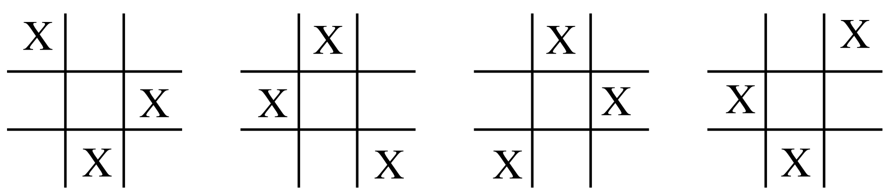


Figure 3.5: The four new lines added to a traditional Tic-Tac-Toe board to get the lines of $AG(2,3)$.

of four parallel classes of lines. The parallel classes are the three vertical lines, the three horizontal lines, and two new "diagonal" classes that can be seen in Figure 3.6. We should note that as with

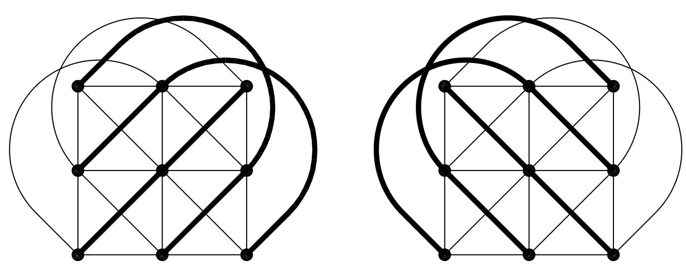
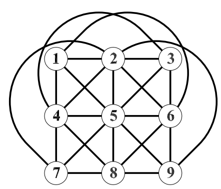


Figure 3.6: The addition of the four new lines creates two new diagonal parallel classes in $AG(2,3)$ of three lines each as seen in bold above.

$PG(2,2)$ the board $AG(2,3)$ does not admit a draw. To see this, suppose that we partition the points of $AG(2,3)$ into two parts $\mathcal{X}$ and $\mathcal{O}$. By the pigeonhole principle, we may assume $|\mathcal{X}| \geq 5$. We may further suppose that $\mathcal{O}$ does not contain any of the three point lines. Under this assumption, the points of $\mathcal{X}$ necessarily meet each of the three lines in each of the four parallel classes. Suppose $|\mathcal{X}| = 5$, and let us consider for example the class of "vertical" lines, call them l_1, l_2, and l_3. Assuming none of these lines is in $\mathcal{X}$ we may further assume that $\mathcal{X}$ contains two points each of l_1 and l_2 but only one point of l_3. Let $\mathcal{X} \cap l_1 = \{\alpha, \beta\}$ and $\mathcal{X} \cap l_3 = \gamma$. The line between α and γ intersects l_2 at a different point than the line between β and γ since any two points of $AG(2,3)$ determine a unique line. However as l_2 contains only three points and two of them are in $\mathcal{X}$ it is necessarily true that at least one of these two lines is contained in $\mathcal{X}$. Thus, we see that $AG(2,3)$ does not admit a draw.

It follows by a strategy stealing argument that playing first Xavier can win if he plays optimally. We demonstrate a winning strategy for Xavier by considering the game openings. We refer to the position numbers just as in the Tic-Tac-Toe board as before.



After Xavier's first turn there are one X and one O on the board. As noted above $AG(2,3)$ has a doubly transitive automorphism group. We may therefore assume that the X and O fall in positions 1 and 2 respectively on the board. We will use $[\mathcal{X}, \mathcal{O}]$ to denote the state where Xavier has pieces in the positions of $\mathcal{X}$ and Olivia in the positions of $\mathcal{O}$. Thus Xavier plays to the game state $[\{1\},\{2\}]$. Olivia now has two distinct options for her first turn. She can either play on the same line to the state $[\{1,2\},\{3\}]$ or she can play off of the line. If she chooses this second option then up to isomorphism she plays to the state $[\{1,2\},\{4\}]$. These two options are depicted in Figure 3.7.

We first consider the position $[\{1,2\},\{3\}]$. The two X's on the board are indistinguishable (that is an automorphism exists which switches points 1 and 2 while fixing point 3). Thus Xavier can hop from

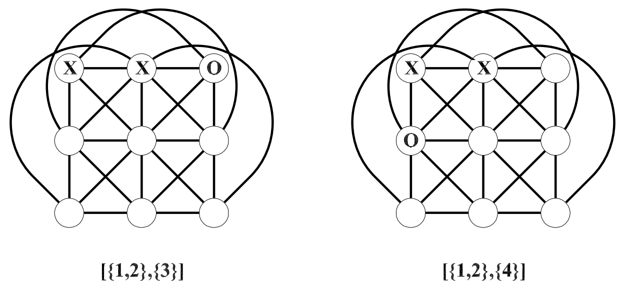


Figure 3.7: The game state on the left is $[\{1,2\},\{3\}]$. Both Xavier and Olivia have made one move and Olivia's move was on the line with the two previously occupied positions. The game state on the right is $[\{1,2\},\{4\}]$, which up to isomorphism is the other possible state after Xavier and Olivia each play once.

either of them. Furthermore allowing for the interchange of points 1 and 2 any of the remaining six points can be mapped to point 4 while still fixing the point 3. Thus, Xavier can be assumed to play to the position $[\{1,4\},\{2,3\}]$ as depicted in Figure 3.8. From this position we can see that Xavier will win on his next move by noting the following:

- There exists an automorphism of $AG(2,3)$ which fixes points 1 and 4 while interchanging points 2 and 3 and thus the two O's are interchangeable.
- Once Olivia has moved, the board will contain three X's and two O's.
- Any three points either (a) form a complete line in $AG(2,3)$ or (b) determine three intersecting lines in $AG(2,3)$, one for each pair of points.
- Three intersecting lines can only be blocked by two points if at least one of those points is at a point of intersection of two of the lines.

Thus after Olivia's second move, the game is in a state where there are three X's on three intersecting lines, each at a point of intersection of two of the lines. It follows that Olivia can have blocked at most two of the three lines with her two O's neither of which is at a point of intersection of two or more of the lines. On his next turn, Xavier can move his extra piece to complete the unblocked winning line.

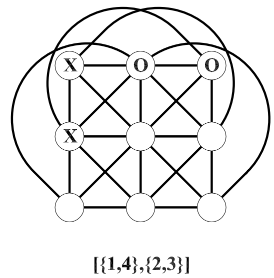


Figure 3.8: The state of the game on $AG(2,3)$ after Xavier's second turn.

We note that as both players have three pieces on the board at the completion of Xavier's turn we need to verify that both Xavier and Olivia do not have winning lines, that is that the game does not end with a tie. We notice that a tie could be possible in an affine plane due to the existence of parallel classes. Let us suppose that Xavier's move does in fact also give Olivia a win. Then both of the lines must exist in the same parallel class $\{l_1, l_2, l_3\}$. Assume Xavier has claimed line l_1 and Olivia line l_2. Prior to Xavier's move he had two points on l_1 and the third was open. Meanwhile Olivia had two points on l_2 and the third was an X. Since two points determine a unique line, the X on line l_2 is on a line with each X on l_1 which is unblocked by either of Olivia's O's. Thus Xavier can win by completing either of these two lines maintaining line l_2 as having two O's and one X, therefore preventing a tie. It follows that Xavier has a winning strategy if Olivia's first move is on the line with the previously occupied points to the state $[\{1,2\},\{3\}]$.

Now let us suppose that on her first turn Olivia played off the line to the position $[\{1,2\},\{4\}]$. Once again Xavier's pieces are indistinguishable and either can be moved. Bearing in mind that the piece he moves will become an O he should move to block the line of O's. That is he moves to a position wherein there is a line with two O's and one X, with the second X off of that line. This however is exactly the same situation as before. In fact an automorphism exists which can map any such configuration to the state $[\{1,4\},\{2,3\}]$. As we have seen Xavier has a winning move from this game state, and thus Xavier has a winning strategy if Olivia plays off of the line of previously occupied points to a state equivalent to $[\{1,2\},\{4\}]$.

As Xavier can win given either first move by Olivia, we see that Xavier has a winning strategy hopping first on $AG(2,3)$. □

3.2 Weight functions on finite planes

Our goal in the following section is to prove that Olivia has a drawing strategy on boards of sufficiently large order. We follow the example of the paper "Tic-Tac-Toe on a Finite Plane" [5] in the use of weight functions to define player strategies. Recall that when Xavier performs a hop move, he replaces one of Olivia's pieces on the board in the position he hopped from. We refer to this piece as the newest O on the board and its position as α. We shall prove that in most cases Olivia can force a draw by always hopping from α. This strategy is not necessarily optimal in terms of the number of moves required to gain a draw, but does provide a drawing strategy for many games. The existence of any drawing strategy for Olivia proves that the game is second player draw. For such boards we provide an algorithm for choosing the position to which she should hop from α. To that end we represent the current state of the game prior to Olivia's i^{th} turn by

$$\sigma_i = [\mathcal{X}_i, \{O_1, O_2, \ldots, O_{i-1}\}, \alpha]$$

where

- $\mathcal{X}_i$ is the set of i positions currently occupied by Xavier,
- $\mathcal{O}_i = \{O_1, O_2, \ldots, O_{i-1}\} \cup \{\alpha\}$ contains the i positions occupied by Olivia,
- α is the position of the newest O on the board.

After Olivia's i^{th} turn, the state of the game is represented by

$$\sigma_i' = [\mathcal{X}_i \cup \{\alpha\}, \{O_1, O_2, \ldots, O_{i-1}, O_i\}] = [\mathcal{X}_i', \mathcal{O}_i']$$

where

• Olivia hopped from α to the open position O_i on her turn,

• $\mathcal{X}'_i = \mathcal{X}_i \cup \{\alpha\}$ contains the $i + 1$ positions occupied by Xavier,

• $\mathcal{O}'_i = \{O_1, O_2, \ldots, O_{i-1}, O_i\}$ contains the i positions occupied by Olivia.

We let $\sigma'_0 = [\{\gamma\}, \emptyset]$ denote the state of the game at the start of the game, that is where the initial X is at position γ. Let σ_∞ denote the game state at the end of the game, that is a game state in which the board is full or in which one of the players has claimed a wining line. As an unbiased positional game, we know using a strategy stealing argument that Olivia cannot win. For Olivia the best outcome is to force a draw by having at least one point on each line of a full board, thus blocking Xavier from winning. We provide a drawing strategy for Olivia that involves always hopping from α, the position of the newest O on the board. One reason for this strategy is any lines Olivia has blocked at game state σ'_i remain blocked in any game state after it.

In order to define a strategy for Olivia, we define two weight functions on the winning sets of the board, recalling that for the boards in question a winning set is comprised of the points of a line of the plane. Thus we define the following weight functions for the lines L of the board. The first weight function, called the α-weight function is used by Olivia to determine which lines are close to being completed and therefore which point should be claimed to block those lines. The second weight function, called the β-weight function, will be used by Xavier to determine where to hop from. Our α-weight function gives higher weights to unblocked lines with more X positions. The α-weight function considers a line blocked if it contains an O not in the position α.

Prior to Olivia's i^{th} turn in game state $\sigma_i = [\mathcal{X}_i, \{O_1, O_2, \ldots, O_{i-1}\}, \alpha]$ and for all $l \in L$ we define the α-weight of the line l at state σ_i,

$$w_\alpha(l|\sigma_i) = \begin{cases} 0 & \text{if } l \text{ is blocked, i.e. } l \cap \{O_1, O_2, \ldots, O_{i-1}\} \neq \emptyset, \\ 2^{-u} & \text{otherwise, where } u \text{ is the number of open positions on } l, \text{ i.e. } u = \left| l \setminus \{\mathcal{X}_i \cup \{\alpha\}\} \right|. \end{cases}$$

We should note that for the purpose of this α-weight function Olivia treats the position α as already belonging to Xavier since her strategy is always to move this piece. Thus this α-weight function takes into account the number of X's on a line after the position α switches back to an X but before Olivia has chosen her next point O_i.

Let $P_i = P \setminus (\mathcal{X}_i \cup \{O_1, O_2, \ldots, O_{i-1}\} \cup \{\alpha\})$, the set of all open positions prior to Olivia's i^{th} turn where P is the set of all points on the board. With this notation the α-weight of the game at state σ_i is

$$w_\alpha(\sigma_i) = \sum_{l \in L} w_\alpha(l).$$

For any two points $p, q \in P_i$, the α-weight of an open point p and the α-weight of a pair of distinct open points p, q at state σ_i are

$$w_\alpha(q|\sigma_i) = \sum_{\substack{l \in L \\ q \in l}} w_\alpha(l) \qquad \text{and} \qquad w_\alpha(p, q|\sigma_i) = w_\alpha(l), \text{ where } \{p, q\} \subseteq l \in L.$$

That is the α-weight of the game at state σ_i is the sum of the weights of the unblocked lines. The α-weight of an open point is the sum of the weights of the lines through it, and the α-weight of a pair of open points is the α-weight of the unique line through them.

We note at this point that a winning line for Xavier has α-weight $2^0 = 1$, thus if Xavier has won the game then the α-weight of the game at state σ_∞ is at least 1, that is $w_\alpha(\sigma_\infty) \geq 1$. Also if Olivia has forced a draw by blocking every line of the board, then each line has α-weight 0 and thus $w_\alpha(\sigma_\infty) = 0$. Thus to show that Olivia has achieved a draw, we need examine the value of $w_\alpha(\sigma_\infty)$. To that end we consider how the game changes between states σ_i and σ_{i+1}. Representative pictures of the points effected during these game states can be seen in Figure 3.9.

Recall that Olivia's strategy is to always hop from α on any turn to an open position of highest α-weight. On her turn, Olivia hops from the game state $\sigma_i = [\mathcal{X}_i, \{O_1, O_2, \ldots, O_{i-1}\}, \alpha]$, to the game state $\sigma_i' = [\mathcal{X}_i \cup \{\alpha\}, \{O_1, O_2, \ldots, O_{i-1}, O_i\}]$. From this state there are two distinct types of states for σ_{i+1}. Either Xavier hops off α, or he does not.

(a) If Xavier hops off α to an open point $\gamma \in P_i \setminus O_i$, the game state is of the form

$\sigma_{i+1} = [\mathcal{X}_i \cup \{\gamma\}, \{O_1, O_2, \ldots, O_i\}, \alpha]$.

(b) If Xavier chooses to hop off some point $\beta \in \mathcal{X}_i$ to an open point $\gamma \in P_i \setminus O_i$, the game state is of the form $\sigma_{i+1} = [\mathcal{X}_i \setminus \{\beta\} \cup \{\alpha, \gamma\}, \{O_1, O_2, \ldots, O_i\}, \beta]$.

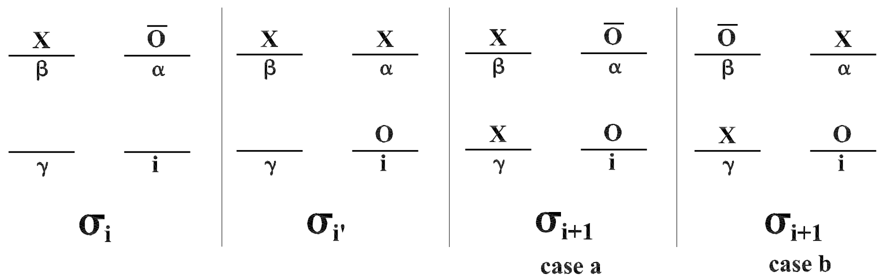


Figure 3.9: Beginning at state σ_i, Olivia hops from α to an open position i which has highest α-weight. On his $(i+1)^{st}$ turn Xavier has two options: hop from α as in case a of σ_{i+1}, or hop from some position β as in case b. In states σ_i and σ_{i+1}, the $\overline{O}$ represents the newest O on the board which is treated as an X by the α-weight function.

The question now is how does each of these moves by Xavier effect the α-weight of the game from state σ_i to state σ_{i+1}?

In case (a), the net effect is that of Xavier and Olivia each placing a piece on the board since each gained a new position and kept all old positions (including the newest O at α). In calculating $w_\alpha(\sigma_{i+1})$, the only lines whose weights were effected from $w_\alpha(\sigma_i)$ are those which contain either O_i or γ. Thus to calculate $w_\alpha(\sigma_{i+1})$ we consider the changes to $w_\alpha(\sigma_i)$. Those lines in L which where unblocked by $\{O_1, \ldots, O_{i-1}\}$, but which contain O_i now do not contribute to the α-weight of the game at state σ_{i+1}, so the α-weight of O_i must be subtracted from $w_\alpha(\sigma_i)$. The weights of the lines containing γ increase from 2^{-u} in $w_\alpha(\sigma_i)$ to $2^{-u+1} = 2 \cdot 2^{-u}$ in $w_\alpha(\sigma_{i+1})$. Thus $w_\alpha(\gamma|\sigma_i)$ is added to $w_\alpha(\sigma_i)$. Now consider the line which contains O_i and γ. We subtracted its α-weight when we subtracted $w_\alpha(O_i|\sigma_i)$ and then added it back when we added $w_\alpha(\gamma|\sigma_i)$. Since it is now blocked we do need to subtract its α-weight so we need to subtract $w_\alpha(O_i, \gamma|\sigma_i)$ from $w_\alpha(\sigma_i)$. That is in calculating $w_\alpha(\sigma_{i+1})$ we get

$$w_\alpha(\sigma_{i+1}) = w_\alpha(\sigma_i) - w_\alpha(O_i|\sigma_i) + w_\alpha(\gamma|\sigma_i) - w_\alpha(O_i, \gamma|\sigma_i).$$

Some basic algebra gives us the equivalent expression

$$w_\alpha(\sigma_i) - w_\alpha(\sigma_{i+1}) = w_\alpha(O_i|\sigma_i) - w_\alpha(\gamma|\sigma_i) + w_\alpha(O_i, \gamma|\sigma_i).$$

In case (b), the net effect is Xavier and Olivia switch ownership of α and β while each claim a new position. Let's first consider the switching of α and β in terms of our α-weight function. Any lines blocked by $\{O_1, O_2, \ldots, O_{i-1}\}$ are still blocked, thus their α-weights are still zero. For those lines which were previously unblocked, since the α-weight function treats the newest O as an X, switching the newest O at α with an existing X at β does not change their α-weights as the same number of positions are still open. Thus the switching of α and β does not change the α-weights of the unblocked lines. It follows that the change in the α-weight of the game in this case is the same as the change induced by each of Xavier and Olivia claiming a new position as in case (a). Thus again we have the change in the α-weight of the game between states σ_i and σ_{i+1} is

$$w_\alpha(\sigma_i) - w_\alpha(\sigma_{i+1}) = w_\alpha(O_i|\sigma_i) - w_\alpha(\gamma|\sigma_i) + w_\alpha(O_i, \gamma|\sigma_i). \tag{3.1}$$

Equation (3.1) gives us Olivia's strategy. Since Olivia is trying to get the α-weight of the game states below 1 she hops from α to maximize $w_\alpha(\sigma_i) - w_\alpha(\sigma_{i+1})$ which she can do by selecting O_i of maximum α-weight $w_\alpha(O_i|\sigma_i)$. To counter this strategy Xavier wants to minimize $w_\alpha(\sigma_i) - w_\alpha(\sigma_{i+1})$ and responds by hopping to the remaining open position γ which maximizes the quantity $w_\alpha(\gamma|\sigma_i) - w_\alpha(O_i, \gamma|\sigma_i)$. Using these strategies for Olivia and Xavier, along with equation (3.1) we get the following result about the sequence of α-weights prior to Olivia's turns. This result will be used repeatedly throughout this dissertation and provides the necessary test for many of the results which follow.

Proposition 3.2.1. *Following the strategy of always hopping from α to a position O_i of maximum α-weight, Olivia can guarantee that on any board $w_\alpha(\sigma_{i+1}) \leq w_\alpha(\sigma_i)$ for all $i \geq 1$.*

Proof. Suppose the current state of the game is $\sigma_i = [\mathcal{X}_i, \{O_1, O_2, \ldots, O_{i-1}\}, \alpha]$, and Olivia is about to make her i^{th} hop. Since the α-weight function assigns more weight to those lines on which Xavier is closest to winning, Olivia should play to an open point of maximal α-weight. That is Olivia plays to $O_i \in P_i$ such that $w_\alpha(O_i|\sigma_i) = max\{w_\alpha(p|\sigma_i) \mid p \in P_i\}$. By Equation (3.1), $w_\alpha(\sigma_{i+1}) \leq w_\alpha(\sigma_i)$ therefore holds for all i since γ and O_i are both open points. That is by Olivia's choice of O_i over γ it is necessarily the case that $w_\alpha(O_i|\sigma_i) \geq w_\alpha(\gamma|\sigma_i)$ and therefore $w_\alpha(\sigma_i) - w_\alpha(\sigma_{i+1}) = w_\alpha(O_i|\sigma_i) - w_\alpha(\gamma|\sigma_i) + w_\alpha(O_i, \gamma|\sigma_i) \geq 0$. □

We now consider the game played on specific classes of boards, first projective planes $PG(2, q)$, followed by affine planes $AG(2, q)$.

3.3 Draws on Projective Planes $PG(2, q)$

Proposition 3.3.1. *Olivia can force a draw hopping second on every projective plane $PG(2, q)$ of order q with $q \geq 5$.*

Proof. To prove that Olivia can force a draw we must provide an algorithm which determines Olivia's move at any point in the game and then prove that the strategy leads to a draw. That is in this case to prove that the sequence of prescribed moves leads to $w_\alpha(\sigma_\infty) < 1$. Given the result of Proposition 3.2.1 this is equivalent to showing there exists N, where $1 \leq N < \infty$ such that $w_\alpha(\sigma_N) < 1$.

In any projective plane, after Xavier's first move the state of the game is $\sigma_1 = [\{\gamma\}, \emptyset, \alpha]$. At this point all $q^2 + q + 1$ lines are considered unblocked by our α-weight function. There is one line containing both α and γ, which has α-weight $2^{-(q-1)}$. Since each point is on exactly $q + 1$ lines of $q + 1$ points each, there are $2(q + 1 - 1) - 1 = 2q$ lines with one or the other of α or γ, each of which has α-weight 2^{-q}. The remaining $q^2 + q + 1 - 2q - 1 = q^2 - q$ lines are completely open and have α-weight $2^{-(q+1)}$.

Thus we have

$$w_\alpha(\sigma_1) = \frac{1}{2^{q-1}} + \frac{2q}{2^q} + \frac{q^2 - q}{2^{q+1}} = \frac{4 + 4q + q^2 - q}{2^{q+1}} = \frac{q^2 + 3q + 4}{2^{q+1}}. \tag{3.2}$$

Here we see that $w_\alpha(\sigma_1) < 1$ when $q \geq 5$. Thus Olivia forces a draw on the projective planes of order $q \geq 5$ by hopping from α to an open point of maximum α-weight at each stage of the game. □

As we have already dealt with $PG(2, 2)$ in Section 3.1, this leaves only two projective planes for which we do not know the game outcome, $PG(2, 3)$ and $PG(2, 4)$. Before we analyze the game on these two boards we need a second weight function which calculates the weight of the lines of the board given the pieces currently in play without treating the piece at α differently. This particular weight function, which we call the β-weight, is used by Xavier and treats all O's as blocking the lines on which they are located. It is similar to the α-weight function in that lines with more X's have higher β-weight and therefore higher priority.

Let $\mathcal{X}$ and $\mathcal{O}$ denote the positions held by Xavier and Olivia respectively at any moment in the game. For example just prior to Olivia's i^{th} turn at game state $\sigma_i = [\mathcal{X}_i, \{O_1, O_2, \ldots, O_{i-1}\}, \alpha]$ we would have $\mathcal{X} = \mathcal{X}_i$ and $\mathcal{O} = \{O_1, O_2, \ldots, O_{i-1}\} \cup \{\alpha\}$ Then for all $l \in L$ we define the β-weight $w_\beta(l) =$

$$\begin{cases} 0 & \text{if } l \text{ is blocked, i.e. } l \cap \mathcal{O} \neq \emptyset, \\ 2^{-u} & \text{otherwise, where } u \text{ is the number of open positions on } l \text{, i.e. } u = \left| l \setminus \mathcal{X} \right|. \end{cases}$$

We note here that the main differences between the α-weight and the β-weight are (1) lines containing the position α are considered unblocked and having an extra "X" by the α-weight function but are considered blocked by the β-weight function and (2) the α-weight function can only be used just prior to Olivia's turn while the β-weight function can be used for any game state $\sigma = \sigma_i$ or σ_i'. Once again we define the β-weights of the game state σ, an open point q and an open pair of distinct points p, q as follows:

$$\begin{aligned} w_\beta(\sigma) &= \sum_{l \in L} w_\beta(l) \\ w_\beta(q|\sigma) &= \sum_{\substack{l \in L \\ q \in l}} w_\beta(l) \\ w_\beta(p, q|\sigma) &= w_\beta(l) \quad \text{where } \{p, q\} \subseteq l \in L \end{aligned}$$

We use the β-weight function to explore Xavier's choices in specific game examples. In particular it is used to demonstrate the differences in Xavier's choices of where to hop from. The β-weight function also helps to distinguish isomorphic hopping options. We should note however that Xavier also uses the α-weight function to determine where to hop to. Referring to Equation (3.1), just as Olivia is best served by maximizing $w_\alpha(\sigma_i) - w_\alpha(\sigma_{i+1})$ by hopping to O_i of maximum α-weight, Xavier's best strategy is to minimize $w_\alpha(\sigma_i) - w_\alpha(\sigma_{i+1})$ by hopping to an open position γ such that $w_\alpha(\gamma|\sigma_i) - w_\alpha(O_i, \gamma|\sigma_i)$ is maximal. Thus Xavier uses the α-weight function to determine where to hop to and the β-weight function to determine where to hop from.

We should note that the α- and β- weight functions are closely related to one another. In particular if one were to stop the game in the middle of Olivia's turn when she has replaced her piece at α with an X but has not yet placed her O in its new position on the board, then the β-weight of the board at that moment is the same as the α-weight she calculated prior to her turn. That is the α-weight function is the β-weight function evaluated at a very specific moment in time. The reason that we define the two weight functions separately is therefore a matter of convenience rather than one of necessity.

As we noted above, Xavier chooses where to hop to by finding the open position γ such that $w_\alpha(\gamma|\sigma_i) - w_\alpha(O_i, \gamma|\sigma_i)$ is maximal after Olivia hops from α to O_i on her i^{th} turn. Given the relationship between the α- and β-weight functions, it turns out that

$$w_\alpha(\gamma|\sigma_i) - w_\alpha(O_i, \gamma|\sigma_i) = w_\beta(\gamma|\sigma'_i). \tag{3.3}$$

Let us consider the positions effected by Olivia's turn and how those changes effect the weight functions. In particular, at state σ_i, the two weight functions differ only on unblocked (by $\{O_1, \ldots, O_{i-1}\}$) lines containing α. For the β-weight functions these lines have weight 0 while they have weight 2^{-u} for the α-weight function where u is the number of open positions. Now lets calculate $w_\beta(\gamma|\sigma_i')$ by considering the change from $w_\beta(\gamma|\sigma_i)$. Let L be the line shared by α and γ. If L is unblocked by $\{O_1, \ldots, O_{i-1}\}$ at state σ_i, then $w_\alpha(L|\sigma_i) = 2^{-U}$ where U is the number of open positions on L, otherwise $w_\alpha(L|\sigma_i) = 0$. In either case we see that when Olivia hops from α to O_i this weight must be added to the β-weight of γ as an X is places at position α. That is

$$\begin{aligned}
w_\beta(\gamma|\sigma_i') &= w_\beta(\gamma|\sigma_i) + w_\alpha(L|\sigma_i) - w_\beta(\gamma, O_i|\sigma_i) \\
&= [w_\beta(\gamma|\sigma_i) + w_\alpha(L|\sigma_i)] - w_\beta(\gamma, O_i|\sigma_i) \\
&= w_\alpha(\gamma|\sigma_i) - w_\alpha(\gamma, O_i|\sigma_i).
\end{aligned}$$

We see that for any open positions γ, O_i, $w_\alpha(\gamma|\sigma_i) - w_\alpha(O_i, \gamma|\sigma_i) = w_\beta(\gamma|\sigma_i')$. Given this relationship, Xavier's strategy of hopping to the remaining open point which maximizes $w_\alpha(\gamma|\sigma_i) - w_\alpha(O_i, \gamma|\sigma_i)$ can be restated as hopping to the open point of highest β-weight $w_\beta(\gamma|\sigma_i')$.

Proposition 3.3.2. *Olivia can force a draw playing second on $PG(2, 4)$ by always hopping from α.*

Proof. The projective plane of order 4 consists of 21 lines of 5 points each. For this plane $w_\alpha(\sigma_1) = \frac{q^2+3q+4}{2^{q+1}} = \frac{32}{32} = 1$ from Equation (3.2). In order to prove Olivia's strategy leads to a draw we show that $w_\alpha(\sigma_2) < 1$. At state $\sigma_1 = [\{\beta\}, \emptyset, \alpha]$ Olivia must choose between two types of points: points on the line with α and β call it L, or points off that line. Let p represent any of the three open points on L (all of which are isomorphic at this state). We calculate the α-weight of p: $w_\alpha(p|\sigma_1) = 4 \cdot 2^{-5} + 1 \cdot 2^{-3} = \frac{1}{4}$ since p is on the line L which has α-weight 2^{-3} and four other lines which are completely open and therefore have α-weight 2^{-5}. Next let p' represent any open point not on L (all of which are isomorphic at this state). We calculate the α-weight of p': $w_\alpha(p'|\sigma_1) = 3 \cdot 2^{-5} + 2 \cdot 2^{-4} = \frac{7}{32}$. We get this number by noting that p' is on one line with β and one line with α each of which has 4 open positions and thus

α-weight 2^{-4}, the other three lines containing p' are completely open with α-weight 2^{-5}. Thus we see that $w_\alpha(p|\sigma_1) > w_\alpha(p'|\sigma_1)$.

Thus following the strategy of hopping from α to the open point of highest α-weight and breaking ties arbitrarily, Olivia's opening move should be to an open point p on L, the line with α and β. That is Olivia hops to the state $\sigma_1' = [\{\beta, \alpha\}, \{p\}]$, blocking the line L as well as the other four lines through p.

Xavier now has two choices: he can hop on L or he can hop off L. We note that $PG(2, 4)$ has a doubly transitive automorphism group, so the choice of which position to hop from is arbitrary as the two X's on the board are interchangeable via some automorphism which fixes the position p. Thus, Xavier need only consider where to hop to.

There are two types of points for which we need to calculate the β-weight, points on L and points off L. Let δ be an open point on L (all of which are isomorphic at this state). The line L which contains δ is blocked and therefore has β-weight 0, the remaining four lines through δ are open with β-weight $2^{-5} = 1/32$. It follows that $w_\beta(\delta|\sigma_1') = 1/8$. Now let γ be an open point not on L. We note that all points off L are isomorphic at this game state. The line through γ containing the O at p is blocked with weight 0, each of the two lines through γ and containing one of the X's has weight 2^{-4}, and the remaining two lines through γ are open. Thus $w_\beta(\gamma|\sigma_1') = 2/16 + 2/32 = 3/16$. As the point γ has higher β-weight than the point δ, and Xavier should hop off of the line L.

We now calculate the α-weight of the resulting game state to determine if it has fallen below 1 thus proving Olivia has a drawing strategy. Using Equations (3.1) and (3.3), noting that Olivia hops to $O_1 = p$ followed by Xavier hopping to γ, we see that

$$\begin{aligned} w_\alpha(\sigma_2) &= w_\alpha(\sigma_1) - w_\alpha(O_1|\sigma_1) + w_\alpha(\gamma|\sigma_1) - w_\alpha(O_1,\gamma|\sigma_1) \\ &= = w_\alpha(\sigma_1) - w_\alpha(O_1|\sigma_1) + w_\beta(\gamma|\sigma_1') \\ &= 1 - \frac{1}{4} + \frac{3}{16} = \frac{15}{16} < 1. \end{aligned}$$

As the α-weight has dropped below 1, we can say by Proposition 3.2.1 that $w_\alpha(\sigma_\infty) < 1$, and therefore, Olivia's strategy of hopping from α to the open position of highest α-weight is a drawing strategy on $PG(2,4)$. □

We can also get this by considering the game state σ_2 after Xavier hops off line L. The game state has line L with one X, and two O's, one of which is at position α, and one X off of line L. The O not at α blocks 5 lines which have α-weight 0. There are two lines of α-weight 2^{-3}, one contains both X's the other the X off L and the new O at α. Eight lines contain exactly one X or just the new O at α and have α-weight 2^{-4}. The remaining $21 - 5 - 2 - 8 = 6$ lines are open with α-weight 2^{-5}. This again gives a total α-weight of $w_\alpha(\sigma_2) = 2 \cdot 2^{-3} + 8 \cdot 2^{-4} + 6 \cdot 2^{-5} = 15/16$. Next we consider the remaining projective plane $PG(2,3)$.

Proposition 3.3.3. *Olivia can force a draw playing second on $PG(2,3)$ by always hopping from α to an open point of maximum α-weight.*

Proof. The projective plane of order 3 consists of 13 lines of 4 points each, we refer to the points as labeled in Figure 3.10. After Xavier's first turn the state of the game up to isomorphism is $\sigma_1 = [\{2\}, \emptyset, 1]$. The α-weight of this state is $w_\alpha(\sigma_1) = \frac{q^2+3q+4}{2^{q+1}} = \frac{22}{16} > 1$ from Equation (3.2). Since this value is larger than 1, we must find an N such that $w_\alpha(\sigma_N) < 1$ to show Olivia has a drawing strategy.

On her first turn, Olivia must choose between two types of points: points on the line $\{1,2,3,12\}$, or points off that line. The points 3 and 12 are each on one line of α-weight 2^{-2} and three lines of α-

weight 2^{-4} and thus both have α-weights of $w_\alpha(3|\sigma_1) = w_\alpha(12|\sigma_1) = 3\cdot 2^{-4} + 2^{-2} = \frac{7}{16}$. Any point p off the line $\{1, 2, 3, 12\}$ is on two lines with α-weight 2^{-3} (one with point 1 and one with point 2) and two lines of α-weight 2^{-4}. Each point off the line therefore has α-weight $w_\alpha(p|\sigma_1) = 2 \cdot 2^{-4} + 2 \cdot 2^{-3} = \frac{6}{16}$. Following the strategy of hopping from α to an open point of maximum α-weight Olivia hops on the line. Given the points 3 and 12 are isomorphic at this game state we assume she hops to the state $\sigma_1' = [\{1, 2\}, \{3\}]$.

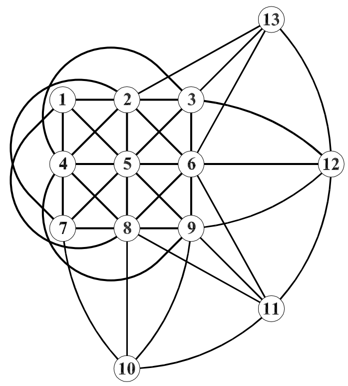


Figure 3.10: The points and lines of the projective plane $PG(2, 3)$.

From here Xavier now has the choice of hopping on the line $\{1, 2, 3, 12\}$ to point 12 or off of the line. We notice that the choice of where to hop from is arbitrary given the positions 1 and 2 are isomorphic at this state in the game. We assume he hops from position 2 and calculate the α-weights for the two different options to determine his better choice. Suppose first that Xavier hops to 12. The game position is then $\sigma_2 = [\{1, 12\}, \{3\}, 2]$. The α-weight of this game can be calculated by noting the following. Of the 13 lines, 4 are blocked by the O at position 3. The remaining 9 lines all intersect the line $\{1, 2, 3, 12\}$ at some other point and therefore have α-weight $1/8$. The total α-weight of this game state is $w_\alpha(\sigma_2) = \frac{9}{8}$.

Now assume he hops off the line, then up to isomorphism we are in the state $\sigma_2 = [\{1, 4\}, \{3\}, 2]$. In the case of the α-weight, we notice that 4 lines are blocked by point 3, one line contains $\{1, 4\}$, one contains $\{2, 4\}$, there are two open lines, and the remaining 5 lines contain exactly one of $1, 2, 4$. It follows that the α-weight of the game state is $w_\alpha(\sigma_2) = 2 \cdot 2^{-4} + 5 \cdot 2^{-3} + 2 \cdot 2^{-2} = \frac{10}{8}$.

We note that Xavier is making the choice of where to hop to by calculating the α-weight of the game state $w_\alpha(\sigma_2)$ after his turn for his two options. What we should recall is given Equation (3.1) by choosing to maximize $w_\alpha(\sigma_2)$ he is following his strategy of hopping to the open position γ which maximizes $w_\alpha(\gamma|\sigma_1) - w_\alpha(O_1, \gamma|\sigma_1)$. If he were to calculate this value for the two types of points 12 or any other open point p he would see that $w_\alpha(12|\sigma_1) - w_\alpha(3, 12|\sigma_1) = \frac{7}{16} - \frac{1}{4} = \frac{3}{16}$ while $w_\alpha(p|\sigma_1) - w_\alpha(3, p|\sigma_1) = \frac{6}{16} - \frac{1}{16} = \frac{5}{16}$. That is points off the line $\{1, 2, 3, 12\}$ maximize $w_\alpha(\gamma|\sigma_1) - w_\alpha(O_1, \gamma|\sigma_1)$ and he should move off the line. We also note that given Equation (3.3) there is a third method Xavier can use to calculate where to hop to on his turn. Xavier could choose by calculating the β-weights of the two types of points at game state σ'_1. First for the point 12: $w_\beta(12|\sigma'_1) = 1 \cdot 0 + 3 \cdot \frac{1}{16} = \frac{3}{16}$. Let p be any open point other than 12, then $w_\beta(p|\sigma'_1) = 1 \cdot 0 + 2 \cdot \frac{1}{8} + 1 \cdot \frac{1}{16} = \frac{5}{16}$. We note that these are the same values we got for the previous method. We will use this third method as we continue through this proof rather than calculating the α-weight of all of the game state options.

The reason for this choice is that the β-weights at state σ'_i are easy to calculate given a picture of the state, we simply add up the weights of the lines through each point. Using the first method we found that Xavier should hop off of the line to a state which has α-weight $5/4$. This method leads to a direct computation of $w_\alpha(\sigma_2)$. Using the other methods and Equations (3.1) and (3.3) we find $w_\alpha(\sigma_2) = \frac{22}{16} - \frac{7}{16} + \frac{5}{16} = \frac{20}{16} = \frac{5}{4}$, which is the same value we got using the first method.

Thus, assuming optimal play, Xavier hops off the line $\{1, 2, 3, 12\}$ to the game state $\sigma_2 = [\{1, 4\}, \{3\}, 2]$ as depicted in Figure 3.11. Note we can easily calculate the α-weight of this game state from the Figure by adding the α-weights of the remaining unblocked lines. There are 2 lines of α-weight $1/4$, 5 lines of α-weight $1/8$, and 2 lines of α-weight $1/16$ leading to the expected α-weight of $5/4$. Since the α-weight $w_\alpha(\sigma_2)$ is still greater than 1 we must consider the game further.

After Xavier plays to $\sigma_2 = [\{1, 4\}, \{3\}, 2]$ we have three types lines of nonzero α-weight, those with 2, 3, and 4 open positions. The types of lines can be seen in Figure 3.11. Using the figure we can calculate the α-weight of any open point by adding up the α-weights of the lines through the point.

p	$w_\alpha(p\|\sigma_2)$
5	$3/8 = 6/16$
6	$3/8 = 6/16$
7	$1/4 + 1/8 + 1/16 = 7/16$
8	$2/8 + 1/16 = 5/16$
9	$1/4 + 1/8 + 1/16 = 7/16$
10	$1/4 + 1/8 + 1/16 = 7/16$
11	$2/8 + 1/16 = 5/16$
12	$1/8 + 2/16 = 4/16$
13	$1/4 + 2/16 = 4/16$

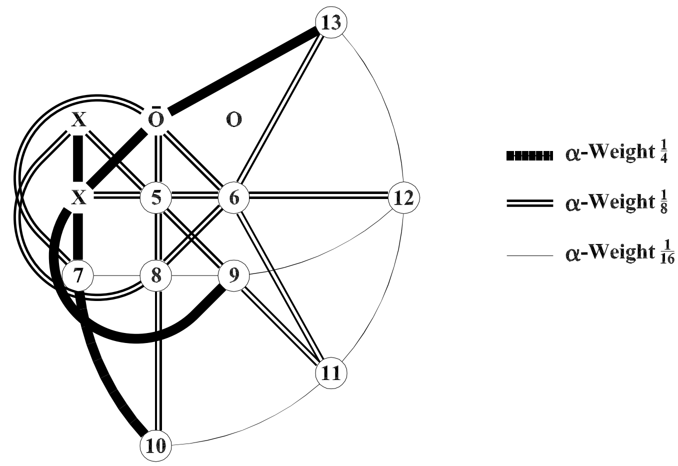


Figure 3.11: The game state depicted is $\sigma_2 = [\{1,4\},\{3\},2]$ which occurs after Xavier's second turn, Xavier having just hopped from 2 to 4. We indicate the newest O is in position 2 by $\overline{O}$ and omit any line blocked by the O at point 3. The α-weights of the open points p are calculated in the table on the left.

For example we see that point 7 is on one line with 2 open points, one with 3 open points, and one with 4 open points. It follows that $w_\alpha(7|\sigma_2) = \frac{1}{4} + \frac{1}{8} + \frac{1}{16} = \frac{7}{16}$. The α-weight calculations for the remaining open points can be seen in the table in Figure 3.11.

In order to play to a point of highest α-weight Olivia should hop from 2 to any of $\{7, 9, 10, 13\}$ which all have α-weight $\frac{7}{16}$. In any case she is playing to a board with two intersecting lines of the form $XXO-$ which intersect at a position occupied by Xavier. Since automorphisms exist which can map any two such lines to any other two these choices are isomorphic and we assume she plays to the state $\sigma_2' = [\{1,2,4\},\{3,7\}]$ as depicted in Figure 3.12.

Xavier's strategy is to find an open point $\gamma \in \{5, 6, 8, 9, 10, 11, 12, 13\}$ which maximizes $w_\alpha(\gamma|\sigma_2) - w_\alpha(7,\gamma|\sigma_2)$. We recall from Equation 3.3 that $w_\beta(\gamma|\sigma_2') = w_\alpha(\gamma|\sigma_2) - w_\alpha(7,\gamma|\sigma_2)$. Thus if we consider the game state σ_2' after Olivia's turn as depicted in Figure 3.12, he should move to the point γ of highest remaining β-weight, which can be calculated by adding the β-weights of the lines through each open point.

We see from the computations accompanying Figure 3.12, Xavier should play to position 13 as it has

γ	$w_\beta(\gamma \mid sigma'_2)$
5	$3/8 = 6/16$
6	$2/8 = 4/16$
8	$2/8 = 4/16$
9	$1/4 + 1/8 = 6/16$
10	$1/8 + 1/16 = 3/16$
11	$1/8 + 1/16 = 3/16$
12	$1/8 + 1/16 = 3/16$
13	$1/4 + 1/8 + 1/16 = 7/16$

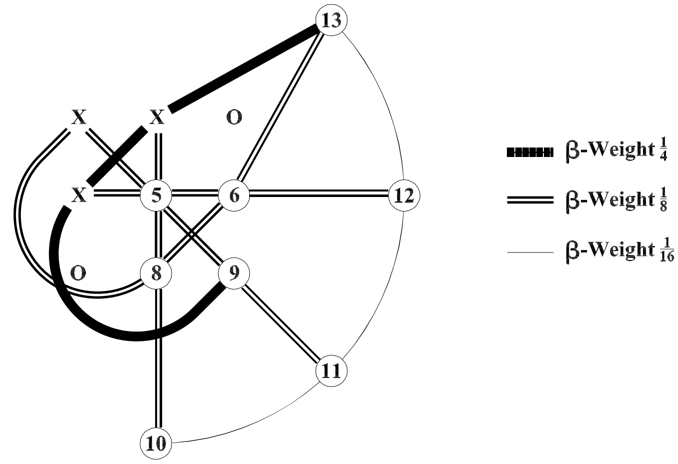


Figure 3.12: The game state depicted is $\sigma'_2 = [\{1, 2, 4\}, \{3, 7\}]$ which occurs after Olivia's second turn, she having just hopped from 2 to 7. We omit any lines blocked by the O's at 3 and 7. The β-weights of the open points are calculated in the table on the left.

the highest β-weight of $\frac{7}{16}$. The question now is which of his three positions $\{1, 2, 4\}$ as in Figure 3.12 should he hop from. Any of his three options result in the same α-weight, which is why he uses the β-weight function to make this decision. In order to determine which of his three pieces to hop, we calculate the β-weight of the state that results from each hopping option.

First hopping from position 1 to position 13 results in the game state $\sigma_3 = [\{2, 4, 13\}, \{3, 7\}, 1]$. Here there are O's on positions $1, 3, 7$ which block 9 of 13 lines. In This case however there is a line of β-weight 1/2 ($\{2, 4, 9, 13\}$) and three have β-weight 1/8 for a total $w_\beta(\sigma_3) = \frac{7}{8}$.

Next we consider the hop from position 2 to position 13 resulting in the game state $\sigma_3 = [\{1, 4, 13\}, \{3, 7\}, 2]$. Here there are O's on positions $2, 3, 7$, which again block 9 of the 13 lines. There is one line of β-weight 1/4 ($\{1, 6, 8, 13\}$) and three have β-weight 1/8 for a total $w_\beta(\sigma_3) = \frac{5}{8}$.

Finally we consider the hop from position 4 to position 13 which give the game state $\sigma_3 = [\{1, 2, 13\}, \{3, 7\}, 4]$. Here there are O's on positions $3, 4, 7$, which between them block 9 of the 13 lines. Of the remaining four lines, one has β-weight 1/4 ($\{1, 6, 8, 13\}$) and three have β-weight 1/8 for a total $w_\beta(\sigma_3) = \frac{5}{8}$.

Based on the information above Xavier's best option is to hop from position 1 to position 13 resulting in the game state $[\{2, 4, 13\}, \{3, 7\}, 1]$ depicted in Figure 3.13, which forces Olivia's next move. We should note that if Xavier hops from 2 or 4 then he is introducing an O on the line which currently has the highest weight. This is an intuitively bad decision which is confirmed by our β-weight function. Although all three game state options have the same α-weight, in practice there is no reason Olivia must hop from α. Thus maintaining the line $\{2, 4, 9, 13\}$ as unblocked is the best choice for Xavier to make.

We notice in this case the α-weight of the game does not change from state σ_2 to σ_3 since $w_\alpha(7|\sigma_2) = w_\alpha(13|\sigma_2) - w_\alpha(7, 13|\sigma_2) = w_\beta(13|\sigma_2')$. That is $w_\alpha(\sigma_3) = w_\alpha(\sigma_2) = \frac{5}{4} > 1$ and we therefore continue.

p	$w_\alpha(P\|\sigma_3)$
5	$3/8$
6	$1/8 + 1/4 = 3/8$
8	$1/8 + 1/4 = 3/8$
9	$1/2 + 1/8 = 5/8$
10	$2/8$
12	$2/8$
12	$2/8$

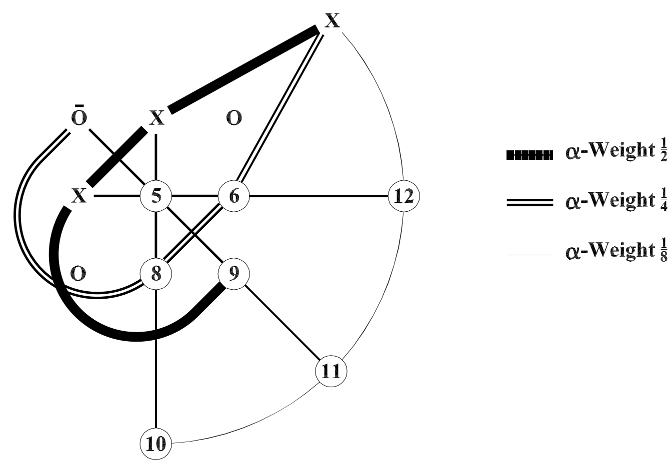


Figure 3.13: The game state depicted is $\sigma_3 = [\{2, 4, 13\}, \{3, 7\}, 1]$ which occurs after Xavier's third turn, Xavier having just hopped from 1 to 13. We indicate the newest O is in position 1 by $\overline{O}$ and omit any lines blocked by the O's at points 3 and 7. The α-weights of the open points p are calculated in the table on the left.

Using Figure 3.13, we see that in order to prevent a win on Xavier's next turn Olivia should move to 9 in order to prevent Xavier from winning on the line $\{2, 4, 9, 13\}$. This is confirmed by the α-weights of the points at this state of the game.

Maintaining her strategy of hopping from α or position 1 in this case, Olivia moves to the state $\sigma_3' = [\{1, 2, 4, 13\}, \{3, 7, 9\}]$ as depicted in Figure 3.14. From here Xavier needs to choose to maximize $w_\beta(\gamma|\sigma_3') = w_\alpha(\gamma|\sigma_3) - w_\alpha(9, \gamma|\sigma_3)$ over the remaining open points $\{5, 6, 8, 10, 11, 12\}$. We see these computations in the table accompanying Figure 3.14.

γ	$w_\beta(\gamma \mid \sigma'_3)$
5	$2/8$
6	$1/4 + 1/8 = 3/8$
8	$1/4 + 1/8 = 3/8$
10	$2/8$
11	$1/8$
12	$2/8$

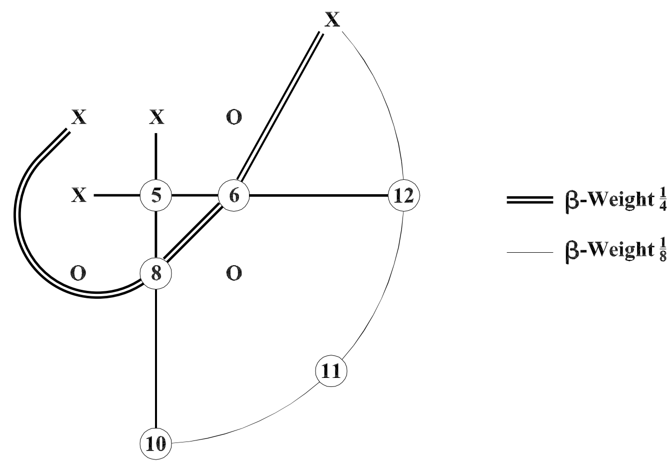


Figure 3.14: The game state depicted is $\sigma'_3 = [\{1, 2, 4, 13\}, \{3, 7, 9\}]$ which occurs after Olivia's third turn, she having just hopped from 1 to 9 to block the win on line $\{2, 4, 9, 13\}$. We omit any lines blocked by the O's at $3, 7$ and 9. The β-weights of the open points are calculated in the table on the left.

We can see from the computations in Figure 3.14, that after Olivia's third turn Xavier should choose to hop to one of 6 or 8. We should note that these two points are in fact isomorphic given the current state of the game. The sets $\mathcal{X}'_3 = \{1, 2, 4, 13\}$ and $\mathcal{O}'_3 = \{3, 7, 9\}$ are fixed by the automorphism $(2, 4)(3, 7)(6, 8)(10, 12)$, which interchanges the two positions in question while also maintaining the weights of the unblocked lines $\{1, 6, 8, 13\}, \{2, 5, 8, 10\}, \{4, 5, 6, 12\}, \{10, 11, 12, 13\}$. It follows that Xavier can arbitrarily choose which of the two to hop to, so we assume he hops to position 8. In order to decide which of the positions $\{1, 2, 4, 13\}$ to hop from we consider the weights of the game states resulting from each of his options.

Consider the hop from position 1 to position 8 which results in game state $\sigma_4 = [\{2, 4, 8, 13\}, \{3, 7, 9\}, 1]$. The O's at positions $1, 3, 7, 9$ block 10 of 13 lines. Of the three remaining lines, $\{2, 5, 8, 10\}$ has weight $1/4$ while $\{4, 5, 6, 12\}$ and $\{10, 11, 12, 13\}$ have weight $1/8$ so that $w_\beta(\sigma_4) = \frac{1}{2}$.

Consider the hop from position 2 to position 8 which results in game state $\sigma_4 = [\{1, 4, 8, 13\}, \{3, 7, 9\}, 2]$. The O's at positions $2, 3, 7, 9$ block 10 of 13 lines. Of the three remaining lines, $\{1, 6, 8, 13\}$ has weight

1/2 while $\{4, 5, 6, 12\}$ and $\{10, 11, 12, 13\}$ have weight 1/8 so that $w_\beta(\sigma_4) = \frac{3}{4}$.

Consider the hop from position 4 to position 8 which results in game state $\sigma_4 = [\{1, 2, 8, 13\}, \{3, 7, 9\}, 4]$. The O's at positions $3, 4, 7, 9$ block 10 of 13 lines. Of the three remaining lines, $\{2, 5, 8, 10\}$ has weight 1/4, $\{1, 6, 8, 13\}$ has weight 1/2, and $\{10, 11, 12, 13\}$ has weight 1/8 so that $w_\beta(\sigma_4) = \frac{7}{8}$.

Consider the hop from position 13 to position 8 which results in game state $\sigma_4 = [\{1, 2, 4, 8\}, \{3, 7, 9\}, 13]$. The O's at positions $3, 7, 9, 13$ block 11 of 13 lines. Of the two remaining lines $\{2, 5, 8, 10\}$ has weight 1/4 and $\{4, 5, 6, 12\}$ has weight 1/8 so that $w_\beta(\sigma_4) = \frac{3}{8}$.

Given these computations, Xavier is best served by hopping from 4 to 8 resulting in the game state $\sigma_4 = [\{1, 2, 8, 13\}, \{3, 7, 9\}, 4]$ as depicted in Figure 3.15. Prior to Olivia's turn we again calculate the α-weight of the game state to determine if it has fallen below 1. Using Figure 3.15 we see that $w_\alpha(\sigma_4) = \frac{1}{2} + \frac{1}{4} + \frac{2}{8} = 1$. Since this is not strictly less than 1 we need to continue.

p	$w_\alpha(p\|\sigma_4)$
5	$1/4 + 1/8 = 3/8$
6	$1/2 + 1/8 = 5/8$
10	$1/4 + 1/8 = 3/8$
11	$1/8$
12	$2/8$

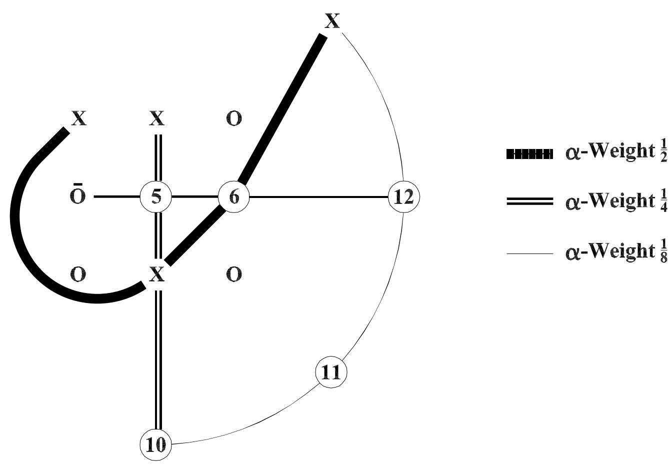


Figure 3.15: The game state depicted is $\sigma_4 = [\{1, 2, 8, 13\}, \{3, 7, 9\}, 4]$ which occurs after Xavier's fourth turn, Xavier having just hopped from 4 to 8. We indicate the newest O is in position 4 by $\overline{O}$ and omit any lines blocked by the O's at points $3, 7$ or 9. The α-weights of the open points are calculated in the table on the left.

We note that once again Xavier has forced Olivia's move by creating a line with only one open position remaining. It follows that Olivia should hop from α or position 4 to position 6 to block that line. Computing the α-weights of the remaining open points $\{5, 6, 10, 11, 12\}$, we see that position 6 does in fact have the highest α-weight and Olivia hops to the game state $\sigma'_4 = [\{1, 2, 4, 8, 13\}, \{3, 7, 9, 6\}]$ as depicted in Figure 3.16, the α-weight function having confirmed the intuitive decision to block the line $\{1, 6, 8, 13\}$.

γ	$w_\beta(\gamma\|\sigma'_4)$
5	$1/4 = 2/8$
10	$1/4 + 1/8 = 3/8$
11	$1/8$
11	$1/8$

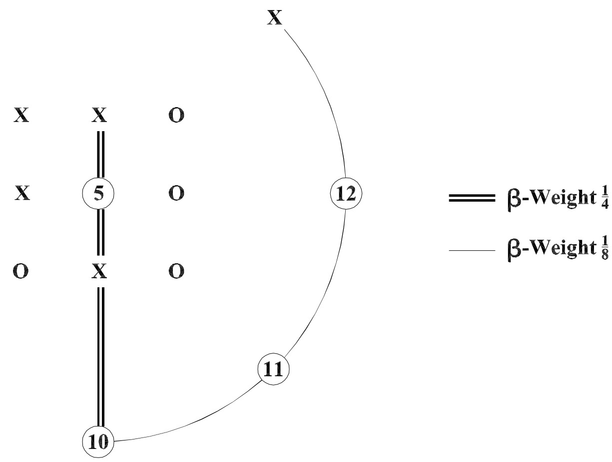


Figure 3.16: The game state depicted is $\sigma'_4 = [\{1, 2, 4, 8, 13\}, \{3, 7, 9, 6\}]$ which occurs after Olivia's third turn, she having just hopped from 4 to 6. We omit any lines blocked by the O's at $3, 6, 7$ and 9. The β-weights of the open points are calculated in the table on the left.

Now needing to maximize $w_\beta(\gamma|\sigma'_4) = w_\alpha(\gamma|\sigma_4) - w_\alpha(6, \gamma|\sigma_4)$, Xavier must choose between the points $\{5, 10, 11, 12\}$. Given the calculation in the table accompanying Figure 3.16, Xavier should move to 10 the point on the intersection of the two remaining unblocked lines. Before we consider where he should hop from, we note that Olivia decreased the α weigh by $\frac{5}{8}$ and Xavier was only able to recover $\frac{3}{8}$. It follows from Equation (3.1) that $w_\alpha(\sigma_5) = 1 - \frac{5}{8} + \frac{3}{8} = \frac{3}{4} < 1$. As the α-weight of the game has fallen below 1, Olivia has a drawing strategy on $PG(2, 3)$. Moreover hopping from α to the open point of highest α-weight is a drawing strategy for Olivia. $\square$

The following theorem summarizes the results of this section.

Theorem 3.3.4. *Xavier has a winning strategy hopping first on $PG(2, 2)$, while Olivia has a drawing strategy hopping second on $PG(2, q)$ with $q \geq 3$.*

3.4 Draws on Affine Planes AG(2, q)

We recall that the affine plane of order q, which we denote by $AG(2, q)$, contains $q^2 + q$ lines of q points each. Every point in the plane is on exactly $q + 1$ lines, one from each of the $q + 1$ parallel classes of lines. We can picture the plane as a $q \times q$ grid of points which we number from left to right, top to bottom. We further recall that $AG(2, q)$ has a doubly transitive automorphism group so up to isomorphism the game state after Xavier's first move is $\sigma_1 = [\{\beta\}, \emptyset, \alpha]$.

Proposition 3.4.1. *Olivia can force a draw hopping second on $AG(2, q)$ for $q \geq 7$ by always hopping from α to the open point of highest α-weight.*

Proof. Recall from Proposition 3.2.1 we know that $w_\alpha(\sigma_i) \geq w_\alpha(\sigma_{i+1})$ for all i, so it remains to show that for some N, $w_\alpha(\sigma_N) < 1$ in the affine plane.

Let us consider the game at state $\sigma_1 = [\{\beta\}, \emptyset, \alpha]$. Again recall that for the purpose of the α-weight function both α and β are considered to contain X's. Thus of the $q^2 + q$ lines in the board, 1 has α-weight $2^{-(q-2)}$, $2(q + 1 - 1) = 2q$ have α-weight $2^{-(q-1)}$ and the remaining $q^2 + q - (2q) - 1 = q^2 - q - 1$ are open with α-weight 2^{-q}. Thus we have the α-weight of the game after Xavier's first move is:

$$w_\alpha(\sigma_1) = \frac{1}{2^{q-2}} + \frac{2q}{2^{q-1}} + \frac{q^2 - q - 1}{2^q} = \frac{q^2 + 3q + 3}{2^q}.$$

We see that $w_\alpha(\sigma_1) < 1$ for $q \geq 6$, however as no affine plane of order 6 exists we have that Olivia can force a draw on all affine planes of order $q \geq 7$ by always hopping from α the position of the newest O. □

As we considered $AG(2, 2)$ and $AG(2, 3)$ in Section 3.1, we have only two boards remaining to consider, $AG(2, 4)$ and $AG(2, 5)$. The complexity of play on these boards prohibits analysis by hand. However, we can use our β-weight function to define a brute force strategy which is optimal for both players. The best strategy for each player is to, on each turn, calculate for every possible hop the

β-weight of the resulting game state. Xavier will choose the hop which results in the highest β-weight and Olivia the hop which results in the lowest β-weight. Given this strategy, the question remains of if it is a winning strategy for Xavier or a drawing strategy for both players. In order to answer this question we use an analysis by computer to determine the game outcomes for these two boards. The result of these computations are as follows.

Proposition 3.4.2. *Xavier has a winning strategy hopping first on $AG(2, 4)$ against any play by Olivia.*

In fact Xavier can win by always hopping his newest X. That is by borrowing Olivia's method of playing, he can overcome any strategy of Olivia's, in particular the optimal brute force strategy described above. The MATLab code for this result can be found in Appendix A. One should note that a drawn state does exist for this board, so the fact that Xavier still has a winning strategy is in some sense unexpected as all of our other first player win boards did not admit a drawn state.

Proposition 3.4.3. *Olivia can force a draw hopping second on $AG(2, 5)$.*

Furthermore Olivia's strategy of hopping from α to the open point of highest α-weight is a drawing strategy on this board although if Xavier plays with the brute force strategy described above the game must be played almost to completion before $w_\alpha(\sigma_i)$ drops below 1. The MATLab code for this result can be found in Appendix B. The following theorem summarizes the results of this section.

Theorem 3.4.4. *Olivia can force a draw hopping second on $AG(2, q)$ for all $q \geq 5$.*

Chapter 4

The game “Fire and Ice”™

“Fire and Ice”™ is a variation on traditional positional games like Tic-Tac-Toe. The game was designed by Jens-Peter Schliemann and is produced by Pin International. The game board exploits the symmetries of the finite projective plane $PG(2,2)$, also known as the Fano Plane. A depiction of the game board can be seen in Figure 4. Players attempt to control the board by possessing three islands in a line. Rather than placing a piece in an open position, a player moves a piece to an *allowed* position, and then replaces the piece with an opponent's. That is, the game is played using a move like the hop we explored in Chapters 2 and 3 where the new position must be chosen under some restrictions.

4.1 The game “Fire and Ice”™

The game board for “Fire and Ice”™ is divided into seven islands, each containing seven positions, arranged in the shape of $PG(2,2)$, as seen in Figure 4. That is, each island contains seven playing positions, arranged into seven lines of three positions each. The seven islands are themselves also arranged in the shape of $PG(2,2)$, with each island occupying the position of a single point.

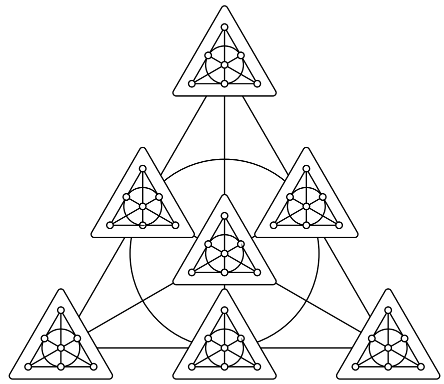


Figure 4.1: The seven islands of “Fire and Ice”™ are arranged in the shape of $PG(2,2)$. A player wins by claiming any three islands which form a line in the larger board.

The game is designed to be played by two players: FIRE who plays first and ICE who plays second. We use X’s to represent FIRE’s pieces and O’s to represent ICE’s pieces. The players FIRE and ICE each attempt to gain control of the board by possessing three islands in a line on the larger board. To gain control of an island a player must have his or her pieces in three positions which form a line on the island. That is, to win the larger game a player must win smaller island games in strategic locations. The board configuration for the game of “Fire and Ice” has several important properties which we explore, which among other things means that the game can end in neither a draw nor a tie.

The method of play for “Fire and Ice”™ differs from playing traditional Tic-Tac-Toe and also from playing our new hop-positional games. “Fire and Ice” begins with a single X in the central position of the central island. $PG(2,2)$ has a transitive automorphism group, thus any position on the board can be mapped to any other position and the choice of the central position provides a convenient place to begin play. The game proceeds with FIRE playing first, and continues with FIRE and ICE alternating play until one player has gained control of three islands in a line. A single turn consists of two parts: move and replace. This is very much like the hop move defined in Section 1.4 with one difference. We restrict the positions one is allowed to move to by defining two special types of hop: *slide* and *jump*.

A *slide* is a move of a player’s piece which affects only one island. The player moves an existing piece to *any* open position on the island. The player’s turn is completed by replacing the moved piece with one of opposing color. That is, a slide is a hop on a single fixed island. An example can be seen in Figure 4.2.

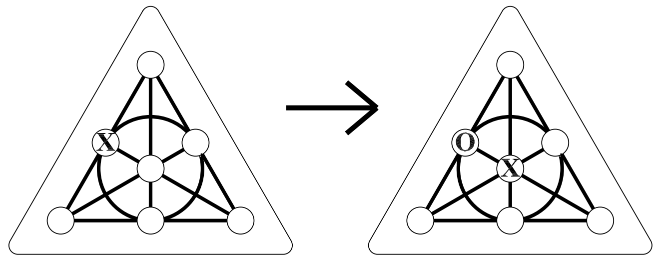


Figure 4.2: An example of a slide move on a single island. The *X* is moved from the left-side position to the center position on the same island and replaced with an *O*.

A *jump* is a move by one player which affects two islands. The player moves an existing piece off its initial island to an unoccupied position in the same relative location on any other island. For example a piece from the center of one island can jump to the center of another island. Once again, the piece is replaced by an opposing piece to complete the player’s turn. An example of a jump by FIRE can be seen in Figure 4.3. A jump is then in one sense a hop of the islands, where one chooses which island to hop from and which to hop to, with the condition that the position on the island is predetermined (and therefore must be open on the destination island).

4.2 Analyzing “Fire and Ice”™

“Fire and Ice”™ is a game of perfect information wherein each player’s options are known to both players. The game is finite as the board has only 49 positions and each turn results in one more position being occupied. To win the game a player needs to occupy at least 9 positions. Thus the game could end in as little as 8 turns by FIRE and 8 turns by ICE (since FIRE begins with one piece on the board). It could also be played until all positions are occupied which would take 24 turns by FIRE and 24 by ICE. The game therefore lasts between 16 and 48 total player turns.

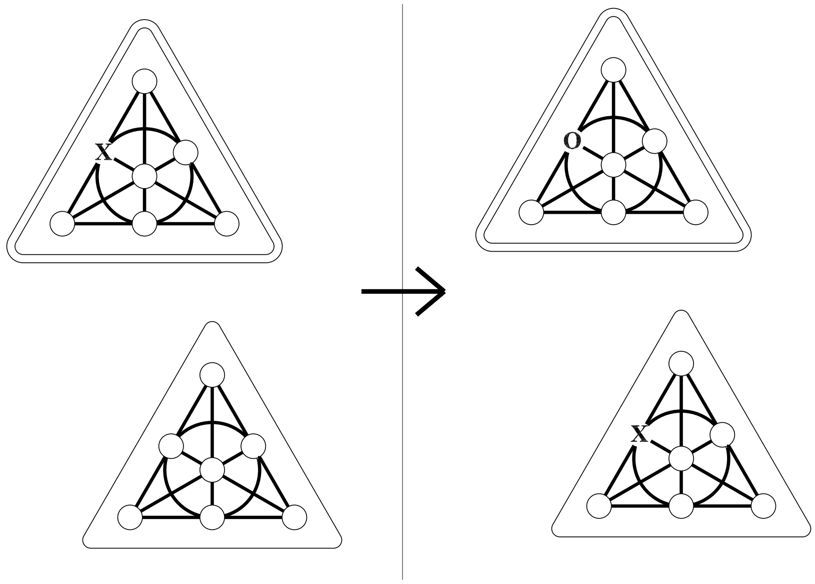


Figure 4.3: An example of a jump between two islands. FIRE jumps from the left-side position on one island to the left-side position on another. This can be performed between any two islands as long as the same relative position is open on the destination island.

Some questions we consider are: Can the game end in a tie? Can the game end in a draw? Which player (if any) has an advantage? To answer the first two questions we explore properties of the board to determine if such states can exist. The third question is a question about “Fire and Ice”™ as a positional game. We will also consider game strategies for “Fire and Ice”™ .

Proposition 4.2.1. *The game “Fire and Ice”™ cannot end in a tie.*

Proof. We first note that the projective plane $PG(2,2)$ has the property that every pair of lines is intersecting. This is guaranteed by the axioms of projective planes discussed in Chapter 1. Since any pair of lines in $PG(2,2)$ intersect, is not possible for both FIRE and ICE to claim lines on a single island. Thus an island can be claimed by at most one player. Similarly both players cannot gain control of winning lines of islands in the overall game. The game can therefore be won by at most one player or stated another way, the game cannot end in a tie. $\square$

Now to the question of drawing the game, that is, the game ending with neither player having claimed a winning line. We have the following proposition about "Fire and Ice"™.

Proposition 4.2.2. *The game "Fire and Ice"™ cannot end in a draw.*

Proof. Suppose the game ends in a draw, i.e. the board is full and neither player has claimed a winning set of positions. First, as the board is full every island is full and must therefore be claimed by either FIRE or ICE. We see this by recalling Lemma 3.1.3, which states that any partition of the points of $PG(2,2)$ necessarily has a line contained in one partition. Thus the islands of "Fire and Ice"™ must all be claimed in a full board. Put another way, the islands of the board are partitioned into those claimed by FIRE and those claimed by ICE. Applying Lemma 3.1.3 to the islands which are themselves arranged in the shape of $PG(2,2)$, it is necessarily the case that one of FIRE or ICE must have a line of islands. This contradicts our assumption that the game ended in a draw. Thus the game "Fire and Ice"™ cannot end in a draw. □

Since the game can end in a neither a tie nor a draw, either FIRE can force a win playing first, or ICE can force a win playing second. However, we will show that ICE cannot have a winning strategy and thus "Fire and Ice" ™ is a first player (FIRE) win game.

Proposition 4.2.3. *FIRE has a winning strategy playing first.*

Proof. To see this we use a standard strategy stealing argument and begin by supposing that ICE, playing second, has a winning strategy. Let FIRE make a random, and allowed, first move, i.e. either a slide on the center island or a jump to the center position of some other island. FIRE will then play the rest of the game pretending to be the second player and using ICE's winning strategy. Specifically FIRE plays as if that first move had not been made. If at any point FIRE's strategy would involve making that first move another random move should be made. Any necessary random moves, including the opening move, cannot hurt FIRE since FIRE is gaining a position on some island. Following ICE's winning strategy playing second, FIRE can win the game. This contradicts the assumption that ICE had a winning strategy. It follows that FIRE has a winning strategy playing first.

□

This proposition proves that "Fire and Ice"™ is a first player win game, however it does not provide an actual winning strategy.

Using the β-weight function of Chapter 3, FIRE can win the game by calculating on each turn the β-weight of the game state that results from each of his possible moves. Recall the β-weight function assigns weights to the winning sets of a board based on the number of open positions remaining on lines unblocked by Olivia. This computation is more complicated than others we have used as for each piece on the board FIRE must calculate the β-weight resulting from all of the possible slides and all of the possible jumps. Furthermore, because FIRE and ICE have different options (places they are allowed to move to by the slide and jump rules) at any game state, we cannot say that the weights of the game states are non-increasing as we did in Proposition 3.2.1. Instead we use a computer program to play the complete game using the strategy described. In doing so we allow ICE to also make the optimal move on each turn. That is, ICE will also consider all possible moves and make the move which reduces the β-weight of the game state the most. The MATLab code for the game can be seen in Appendix C.

The game "Fire and Ice"™ inspired a new type of board which we call nested boards which we explore in Chapter 5.

Chapter 5

Positional Games on Nested Boards of Finite Geometries

In this chapter we explore positional games on a new type of board inspired by the board of the game "Fire and Ice"™ which we explored in Chapter 4.

5.1 Nested Boards

Let E be a set of points or positions and $\mathcal{W} \subseteq 2^E$ be some collection of subsets of E which we call winning sets. We define the board $\mathcal{M} = (E, \mathcal{W})$ as the positions of E along with the winning sets $\mathcal{W}$. If $\mathcal{M}_1 = (E_1, \mathcal{W}_1)$ and $\mathcal{M}_2 = (E_2, \mathcal{W}_2)$ are two boards, we define the nested board $[\mathcal{M}_1 : \mathcal{M}_2]$. To create the nested board we replace each of the points of $\mathcal{M}_1$ with a copy of $\mathcal{M}_2$. We refer to $\mathcal{M}_1$ as the outer component and $\mathcal{M}_2$ as the inner component. The nested board $[\mathcal{M}_1 : \mathcal{M}_2]$ has point set $E_1 \times E_2$. We denote by $[\mathcal{W}_1 : \mathcal{W}_2]$ the collection of winning sets of the nested board $[\mathcal{M}_1 : \mathcal{M}_2]$, where

$$\begin{aligned}[\mathcal{W}_1 : \mathcal{W}_2] &= \{\cup_{i=1}^{|W|} \{p_i \times W_i\} \mid W = \{p_1, p_2, \ldots, p_{|W|}\} \in \mathcal{W}_1, W_i \in \mathcal{W}_2 \text{ for } i = 1, 2, \ldots, |W|\} \\ &= \{W \times (W_i)_{i=1}^{|W|} \mid W \in \mathcal{W}_1, W_i \in \mathcal{W}_2, i = 1, 2, \ldots, |W|\}.\end{aligned}$$

Thus, we obtain a winning set in the nested board by independently replacing each point p_i of some winning set $W \in \mathcal{W}_1$ with the set $p_i \times W_i$ where W_i is some winning set in $\mathcal{W}_2$. We denote such a winning set by $W \times (W_i)_{i=1}^{|W|}$.

For example, we denote by $[PG(2,2) : PG(2,2)]$ the board we used in "Fire and Ice"™. In "Fire and Ice"™ a player wins the game by claiming a winning set of islands or inner components. These islands replaced a set of points which form a winning set in the outer component. An island is claimed by collecting a wining set of positions on the island, that is a winning set of the inner component.

Before we study games on nested boards, we explore some of the properties of nested boards whose component boards satisfy certain desirable regularity conditions. In particular we restrict ourselves to component boards for which all of the winning sets have the same cardinality and for which each point is in exactly the same number of winning sets. Consider a board $\mathcal{M} = \mathcal{M}(E, \mathcal{W})$ satisfying these conditions. We let k be the number of points per winning set, that is $k = |W|$ for all $W \in \mathcal{W}$. Also let l be the number of winning sets containing each point, $n = |E|$ be the number of points, and $m = |\mathcal{W}|$ the number of winning sets. Where necessary we use subscripts to distinguish between these constants for the two component boards. For example k_2 represents the number of points per winning set in the component $\mathcal{M}_2$. We refer to a nested board whose components both satisfy these conditions as regular nested boards. For the remainder of this chapter we consider only regular nested boards. For clarity we refer to a winning set in a component as a line in the component to distinguish from the winning sets of the nested board.

We first investigate the properties of nested boards that satisfy our desired regularity conditions. That is boards for which both components have well define values for k and l. As we replace each point of

$\mathcal{M}_1$ with a copy of $\mathcal{M}_2$, the regular nested board $[\mathcal{M}_1 : \mathcal{M}_2]$ has $N := n_1 \times n_2 = |E_1 \times E_2|$ total points. We next calculate $\left|[\mathcal{W}_1 : \mathcal{W}_2]\right|$. To find a winning set we first select a line in $\mathcal{M}_1$, of which there are m_1. Next for each of the k_1 islands corresponding to the points of that line in $\mathcal{M}_1$, we independently select a line from $\mathcal{M}_2$, of which there are m_2. It follows that there are $M := m_1 \cdot (m_2)^{k_1}$ winning sets in the nested board $[\mathcal{M}_1 : \mathcal{M}_2]$. Each winning set $W \times (W_i)_{i=1}^{k_1}$ in the nested board contains $K := k_1 \cdot k_2$ points; this since each of the k_1 islands or inner components, forming a line in $\mathcal{M}_1$, has a corresponding line from $\mathcal{M}_2$ containing k_2 points. Further, every point is in $L := l_2 \cdot l_1 \cdot (m_2)^{k_1 - 1}$ winning sets on the nested board. We consider a fixed point p which is on some specific island I. The point p is in l_2 winning sets on the island I. In turn I, is in l_1 lines of islands in $\mathcal{M}_1$. For each of the remaining $k_1 - 1$ islands in the line there are m_2 lines to choose from in $\mathcal{W}_2$ on the island.

The nested board $[\mathcal{M}_1 : \mathcal{M}_2]$ exists for any two boards $\mathcal{M}_1$ and $\mathcal{M}_2$. If we restrict ourselves to boards with winning sets of fixed cardinality and points in a fixed number of winning sets, then the resulting regular nested board has the above properties. If we further restrict the component boards to those for whom a pair of points determines a unique winning set, then we gain even more predictable characteristics of the nested board, which are useful in determining game outcomes. For this reason we consider now nested boards whose components are finite geometries $AG(2, n)$ and/or $PG(2, n)$. That is we consider the board $[\mathcal{M}_1 : \mathcal{M}_2]$ where both the outer component $\mathcal{M}_1$ and the inner component $\mathcal{M}_2$ are finite planes. We refer to such a board as a nested board of planes.

Recall as in Chapter 3, to define the board corresponding to a finite geometry, for each line L of the geometry we define the corresponding winning set to be the set of all points on that line. For any such finite geometry the number of points per line, the number of lines through a point, and the number of lines through any two points are fixed. Nested boards of planes are therefore both regular and have the added characteristic that in either component two positions determine a unique line. By restricting our analysis to nested boards of finite geometries we can take advantage of these and other features of the two components when determining the structure of the larger nested board.

Having further restricted our interest to finite planes of type $AG(2, q)$ or $PG(2, q)$ we are able to determine the number of winning sets two points in the nested board of planes share. In all affine and projective geometries, two points determine a unique line. We consider first two points $\{p, q\}$ on the nested board of planes $[\mathcal{M}_1 : \mathcal{M}_2]$ which are on the same island I. As I is a finite plane $AG(2, q)$ or $PG(2, q)$, the points $\{p, q\}$ are on a unique line on I. The island I is on l_1 lines of islands in $\mathcal{M}_1$. To complete a winning set we must have a winning line on each of the other $k_1 - 1$ islands of that line, of which there are m_2 to choose from. Thus, we can say that in the nested board two points on the same island share $L_s := 1 \cdot l_1 \cdot (m_2)^{k_1-1}$ winning sets. Next suppose that p and q are on two different islands I and J respectively. The point p is on l_2 lines on I just as q is on l_2 lines on J. I and J determine a unique line of islands in $\mathcal{M}_1$ which has $k_1 - 2$ other islands. To complete a winning set we select any of the m_2 lines for each of these remaining islands. Therefore two points on two different islands share $L_d := (l_2)^2 \cdot 1 \cdot (m_2)^{k_1-2}$ winning sets on the nested board. We note that if our components have the property that every pair of points determine λ lines, then these computations can be performed in more generality, however as all of our component boards have $\lambda = 1$, this will be sufficient for our purposes.

We recall the values of n, m, k, l for the finite planes $AG(2, q)$ and $PG(2, q)$, where as before we define the winning sets to correspond to the lines of the plane:

	Points n	Lines m	Points/Line k	Lines/Point l
$AG(2, q)$	q^2	$q^2 + q$	q	$q + 1$
$PG(2, q)$	$q^2 + q + 1$	$q^2 + q + 1$	$q + 1$	$q + 1$

Consider for example the nested board $[PG(2, 2) : PG(2, 2)]$ which was used in the game "Fire and Ice"™. For $PG(2, 2)$ we have $m = n = 2^2 + 2 + 1 = 7$ and $k = l = 2 + 1 = 3$ which lead to the following quantities for the nested board:

$N = 7 \times 7 = 49$ points

$M = 7 \times 7^3 = 2401$ lines

$K = 3^3 = 9$ points per line

$L = 3 \times 3 \times 7^2 = 441$ lines through each point

$L_s = 3 \times 7^2 = 147$ lines shared by two points on the same island

$L_d = 3^2 \times 7^1 = 63$ lines shared by two points on different islands

We are now ready to define various games on nested boards of planes. In Section 5.2 we play a traditional positional game on these nested boards. In Section 5.3 we hop on nested boards of planes.

5.2 Tic-Tac-Toe on Nested Boards

We begin by considering a traditional positional game played on a nested board of affine and/or projective planes. Once again we recall that in a traditional positional game two players alternately place pieces in open positions on the board; once a piece is played it is never moved for the remainder of the game. We refer to this game as Tic-Tac-Toe on the nested board. The game ends when one of two conditions is met, either some player wins by accumulating the points of a winning set or the game ends in a draw when the board is full and neither player has a winning set.

We first consider which nested boards admit a draw, that is for which boards could the game end in a draw. A draw on a nested board can occur in one of two ways, either (1) the islands have been claimed in such a way that they form a drawn state in the outer component $\mathcal{M}_1$ or (2) enough of the islands are unclaimed (full with no line claimed) that neither player has a line of islands. That is for a draw to occur on the nested board $[\mathcal{M}_1 : \mathcal{M}_2]$ at least one of the component boards $\mathcal{M}_1, \mathcal{M}_2$ must admit a drawn state. As we saw in Chapter 3, $AG(2,2)$, $AG(2,3)$ and $PG(2,2)$ do not admit drawn states and thus we get the following result by a strategy stealing argument.

Proposition 5.2.1. *The nested board $[\mathcal{M}_1 : \mathcal{M}_2]$ does not admit a drawn state if $\mathcal{M}_1$ and $\mathcal{M}_2$ are chosen from $AG(2,2), AG(2,3)$, and $PG(2,2)$. That is the game Tic-Tac-Toe on the nested board $[\mathcal{M}_1 : \mathcal{M}_2]$ is a first player win game if $\mathcal{M}_1$ and $\mathcal{M}_2$ are chosen from $AG(2,2), AG(2,3)$, and $PG(2,2)$.*

Any nested board whose components are both affine or projective planes, for which at least one component is not chosen from among $AG(2,2), AG(2,3)$, and $PG(2,2)$, admits a drawn state. We begin to consider such nested boards by considering the more general class of nested boards where Olivia has a drawing strategy playing the traditional positional game on the board $\mathcal{M}_2$. We claim that she has a drawing strategy playing the traditional positional game on the nested board $[\mathcal{M}_1 : \mathcal{M}_2]$. For this result no restrictions are placed on the component boards $\mathcal{M}_1$ and $\mathcal{M}_2$.

Proposition 5.2.2. *Suppose that Olivia has a drawing strategy S playing the traditional positional game on the board $\mathcal{M}_2$, then Olivia has a drawing strategy using S playing on the board $[\mathcal{M}_1 : \mathcal{M}_2]$ for any board $\mathcal{M}_1 = (E_1, \mathcal{W}_1)$ where $|E_1| \geq 1$.*

Proof. Let $I_1, I_2, \ldots, I_n$ be the islands of the nested board $[\mathcal{M}_1 : \mathcal{M}_2]$ where $n = |E_1|$. We demonstrate a drawing strategy for Olivia on $[\mathcal{M}_1 : \mathcal{M}_2]$ using the strategy S. Suppose on his i^{th} turn Xavier places a piece on island I_j, Olivia should counter by applying her strategy S to island I_j which is a copy of $\mathcal{M}_2$. By always playing on the same island as Xavier and always following her drawing strategy on that island, Olivia can guarantee that Xavier cannot claim any island of the board. The game therefore ends with all islands drawn and therefore no winning set of $[\mathcal{M}_1 : \mathcal{M}_2]$ claimed by either player. □

We apply this result to our boards of interest by using a result from the paper "Tic-Tac-Toe on a Finite Plane" [5] where Carroll and Dougherty provide strategies for Xavier and Olivia playing the traditional positional game on finite planes using weight functions. The authors prove the following theorems:

Theorem 5.2.3 (Draw Theorem for $PG(2,q)$.)**.** *Olivia can force a draw on every projective plane of order q with $q \geq 3$.*

Theorem 5.2.4 (Draw Theorem for $AG(2,q)$.)**.** *Olivia can force a draw on every affine plane of order q with $q \geq 5$.*

Using these two results we get the following corollaries to Proposition 5.2.2 regarding our boards of interest.

Corollary 5.2.5. *Olivia has a drawing strategy playing the traditional positional game on* $[\mathcal{M} : PG(2, q)]$ *when* $q \geq 3$*, where* $\mathcal{M}$ *is any finite affine or projective plane.*

Corollary 5.2.6. *Olivia has a drawing strategy playing the traditional positional game on* $[\mathcal{M} : AG(2, q)]$ *when* $q \geq 5$ *where* $\mathcal{M}$ *is any finite affine or projective plane.*

Given these two corollaries along with Proposition 5.2.1 we have reduced the number of boards that need further study significantly. We consider four specific classes of nested boards of finite affine and projective planes.

1. $[AG(2, q_1) : PG(2, q_2)]$
2. $[PG(2, q_1) : PG(2, q_2)]$
3. $[AG(2, q_1) : AG(2, q_2)]$
4. $[PG(2, q_1) : AG(2, q_2)]$.

Given the result of Corollary 5.2.5 we know that Tic-Tac-Toe played on $[A(2, q_1) : P(2, q_2)]$ and $[P(2, q_1) : P(2, q_2)]$ are second player draw games if $q_2 \geq 3$. It follows that for these two classes of boards, we only need to consider $[A(2, q) : P(2, 2)]$ and $[P(2, q) : P(2, 2)]$. Further, from Proposition 5.2.1 we know that Tic-Tac-Toe on $[A(2, 2) : P(2, 2)]$, $[A(2, 3) : P(2, 2)]$ and $[P(2, 2) : P(2, 2)]$ is first player win. We are left with the boards $[A(2, q) : P(2, 2)]$ for $q \geq 4$ and $[P(2, q) : P(2, 2)]$ for $q \geq 3$ all of which admit drawn states. Also, given the result of Corollary 5.2.6, we know that Tic-Tac-Toe on $[A(2, q_1) : A(2, q_2)]$ and $[P(2, q_1) : A(2, q_2)]$ are second player draw games for $q_2 \geq 5$. This leaves the boards $[A(2, q_1) : A(2, q_2)]$ and $[P(2, q_1) : A(2, q_2)]$ where $q_2 \in \{2, 3, 4\}$.

For the remaining boards the inner component does not admit a draw, this means that if the game ends with all positions full, then all of the islands are claimed by either Xavier or Olivia. To find a strategy for play we consider one of the weight functions we used in Chapter 3. Since we are only placing pieces, not hopping, we use the β-weight function which assigns weights to the winning sets of a board based on the pieces currently in play. Winning sets with an O are considered blocked and assigned weight 0 while all others are assigned a β-weight based on the number of open positions remaining.

Let $\mathcal{X}$ and $\mathcal{O}$ denote the positions held by Xavier and Olivia respectively at game state σ. For all winning sets $A \in [\mathcal{W}_1 : \mathcal{W}_2]$ we define the β-weight:

$$w_\beta(A|\sigma) = \begin{cases} 0 & \text{if } A \text{ is blocked, i.e. } A \cap \mathcal{O} \neq \emptyset, \\ 2^{-u} & \text{otherwise, where } u \text{ is the number of open positions in } A, \text{ i.e. } u = \left|A \setminus \mathcal{X}\right|. \end{cases}$$

Once again we define the weights of the game state $\sigma = [\mathcal{X}, \mathcal{O}]$, an open point q and an open pair p, q as follows:

$$\begin{aligned} w_\beta(\sigma) &= \sum_{A \in [\mathcal{W}_1 : \mathcal{W}_2]} w_\beta(A|\sigma) \\ w_\beta(q|\sigma) &= \sum_{\substack{A \in [\mathcal{W}_1 : \mathcal{W}_2] \\ q \in A}} w_\beta(A|\sigma) \\ w_\beta(p, q|\sigma) &= \sum_{\substack{A \in [\mathcal{W}_1 : \mathcal{W}_2] \\ \{p,q\} \subseteq A}} w_\beta(A|\sigma) \end{aligned}$$

We use this weight function to provide strategies for Xavier and Olivia on nested boards. We consider the change in the β-weight function from game state $\sigma_i = [\{X_1, X_2, \ldots, X_i\}, \{O_1, O_2, \ldots, O_{i-1}\}]$ just prior to Olivia's i^{th} turn to the game state $\sigma_{i+1} = [\{X_1, X_2, \ldots, X_i, X_{i+1}\}, \{O_1, O_2, \ldots, O_{i-1}, O_i\}]$ just prior to her $(i+1)^{st}$ turn. The difference between states σ_i and σ_{i+1} is that two pieces are added, one X at position X_{i+1} and one O at position O_i. Thus, the β-weight of any winning set containing neither position is unchanged. The unblocked winning sets containing O_i at state σ_i are now blocked and the unblocked winning sets containing X_{i+1} have doubled in β-weight. That is,

$$w_\beta(\sigma_{i+1}) = w_\beta(\sigma_i) - w_\beta(O_i|\sigma_i) + w_\beta(X_{i+1}|\sigma_i) - w_\beta(O_i, X_{i+1}|\sigma_i).$$

Rearranging terms we get the alternate form

$$w_\beta(\sigma_i) - w_\beta(\sigma_{i+1}) = w_\beta(O_i|\sigma_i) - w_\beta(X_{i+1}|\sigma_i) + w_\beta(O_i, X_{i+1}|\sigma_i). \tag{5.1}$$

The relationship described by Equation (5.1) is similar to the relationship described by Equation (3.1), where the later applies only in the hop-positional game when Olivia uses the strategy of hopping from

α. We use Equation (5.1) to describe the strategies for Xavier and Olivia playing Tic-Tac-Toe on nested boards, recalling that both players are placing pieces on open positions, not hopping. Notice that if Xavier claims a winning set then that set has β-weight $2^{-0} = 1$. Thus, if the game ends at state σ_∞ in which the board is full and $w(\sigma_\infty) < 1$, then Xavier cannot have claimed a winning set and Olivia has forced a draw. It follows that Olivia's goal should be to reduce $w_\beta(\sigma)$ as much as possible on each turn while Xavier should try to prevent this decline. That is, Olivia should play to the open position O_i of maximum β-weight. Xavier should respond by playing to the remaining open position X_{i+1} which maximizes the quantity $w_\beta(X_{i+1}|\sigma_i) - w_\beta(O_i, X_{i+1}|\sigma_i)$. That is he should play to the open point γ of maximum β-weight at state $\sigma_i' = [\{X_1, X_2, \ldots, X_i\}, \{O_1, \ldots, O_{i-1}, O_i\}]$.

Proposition 5.2.7. *Olivia can guarantee $w_\beta(\sigma_i) \geq w_\beta(\sigma_{i+1})$, by playing to the open position of highest β-weight. That is, she can guarantee the β-weight of the game states prior to her turn is non-increasing over time.*

Proof. On her turn Olivia has a choice between the position O_i and the position X_{i+1} both of which are open. As Olivia plays to the open point of highest β weight, it is necessarily the case that $w_\beta(O_i|\sigma_i) \geq w_\beta(X_{i+1}|\sigma_i)$. It follows from equation 5.1 that

$$w_\beta(\sigma_i) - w_\beta(\sigma_{i+1}) = w_\beta(O_i|\sigma_i) - w_\beta(X_{i+1}|\sigma_i) + w_\beta(O_i, X_{i+1}|\sigma_i) \geq 0,$$

and therefore $w_\beta(\sigma_i) \geq w_\beta(\sigma_{i+1})$ as desired. □

To determine if Olivia has a drawing strategy on a nested board is a matter of finding a state σ_N such that $w_\beta(\sigma_N) < 1$ and therefore showing that $w_\beta(\sigma_\infty) < 1$ by the proposition. For all of the remaining boards of planes not covered by the results of Corollaries 5.2.5 and 5.2.6, the optimal strategy for both Xavier and Olivia is to play to the open position of highest β-weight on each turn. Determining for the remaining nested boards of planes (those whose inner component does not admit a draw) if the game outcome is a win for Xavier or a draw requires analysis specific to each board, and is currently an open question. To date it appears that such analysis will require the use of a computer as well knowledge of the actual structure of the component boards.

5.3 Hopping on Nested Boards

We now consider the hop-positional games played on nested boards. That is, we consider the positional game where both players make a hop move on each turn rather than placing their own piece in an open position. Recall that a hop consists on moving one's own piece to *any* open position and replacing it with an opponent's piece. We again play on nested boards of planes $[\mathcal{M}_1 : \mathcal{M}_2]$ where both $\mathcal{M}_1$ and $\mathcal{M}_2$ are finite geometries $AG(2, q)$ or $PG(2, q)$. As all of these boards have doubly transitive automorphism group and recalling that the points of $[\mathcal{M}_1 : \mathcal{M}_2]$ are elements of the Cartesian product $E_1 \times E_2$, we may assume the game begins with an X in position $(1, 1)$, that is the game begins at state $\sigma_0' = [\{(1, 1)\}, \emptyset]$. We should note that unlike Tic-Tac-Toe on nested boards, we cannot use a drawing strategy on the inner component to create a drawing strategy for the nested board. This same idea will not work as pieces are allowed to move between islands. Instead we turn to the α-weight function of Chapter 3 and explore for which nested boards Olivia can force a draw by always hopping from α, the position of her newest O. Once again we represent the current state of the game prior to Olivia's i^{th} turn by $\sigma_i = [\mathcal{X}_i, \{O_1, O_2, \ldots, O_{i-1}\}, \alpha]$ where

- $\mathcal{X}_i$ is the set of i positions currently occupied by Xavier,
- $\mathcal{O}_i = \{O_1, O_2, \ldots, O_{i-1}\} \cup \{\alpha\}$ contains the i positions occupied by Olivia,
- α is the position of the newest O on the board.

After Olivia's i^{th} turn, the state of the game is represented by $\sigma_i' = [\mathcal{X}_i \cup \{\alpha\}, \{O_1, O_2, \ldots, O_{i-1}, O_i\}] = [\mathcal{X}_i', \mathcal{O}_i']$ where

- Olivia hopped from α to the open position O_i on her turn,
- $\mathcal{X}_i' = \mathcal{X}_i \cup \{\alpha\}$ contains the $i + 1$ positions occupied by Xavier,
- $\mathcal{O}_i' = \{O_1, O_2, \ldots, O_{i-1}, O_i\}$ contains the i positions occupied by Olivia.

Let σ_∞ again denote the game state at the end of the game.

Prior to Olivia's i^{th} turn in game state $\sigma_i = [\mathcal{X}_i, \{O_1, O_2, \ldots, O_{i-1}\}, \alpha]$ and for all $A \in [\mathcal{W}_1 : \mathcal{W}_2]$ we define the α-weight at state σ_i,

$$w_\alpha(A|\sigma_i) = \begin{cases} 0 & \text{if } A \text{ is blocked, i.e. } A \cap \{O_1, O_2, \ldots, O_{i-1}\} \neq \emptyset, \\ 2^{-u} & \text{otherwise, where } u \text{ is the number of open positions on } A\text{, i.e. } u = \left|A \setminus \{\mathcal{X}_i \cup \{\alpha\}\}\right|. \end{cases}$$

Recall the α-weight function treats the position α as already belonging to Xavier since Olivia's strategy is always to move this piece. Thus, this α-weight function takes into account the weight of a line after the position α switches back to an X but before Olivia has chosen her next point O_i.

Let $P_i = P \setminus (\mathcal{X}_i \cup \{O_1, O_2, \ldots, O_{i-1}\} \cup \{\alpha\})$, the set of all open positions prior to Olivia's i^{th} turn where P is the set of all points on the board. With this notation the α-weight of the game at state σ_i, of an open point q and an open pair p, q are:

$$w_\alpha(\sigma_i) = \sum_{A \in [\mathcal{W}_1 : \mathcal{W}_2]} w_\alpha(A) w_\alpha(q|\sigma_i) = \sum_{\substack{A \in [\mathcal{W}_1 : \mathcal{W}_2] \\ q \in A}} w_\alpha(A) w_\alpha(p, q|\sigma_i) = \sum_{\substack{A \in [\mathcal{W}_1 : \mathcal{W}_2] \\ \{p,q\} \subseteq A}} w_\alpha(A).$$

That is the α-weight of the game at state σ_i is the sum of the weights of the unblocked lines. The α-weight of an open point is the sum of the weights of the winning sets containing it, and the α-weight of a pair of open points is the sum of the α-weights of the winning sets containing both of them. Once again if Olivia always hops from α, then the change in the α-weights of the game states between just prior to her i^{th} and $(i+1)^{st}$ turns satisfies

$$w_\alpha(\sigma_i) - w_\alpha(\sigma_{i+1}) = w_\alpha(O_i|\sigma_i) - w_\alpha(\gamma|\sigma_i) + w_\alpha(O_i, \gamma|\sigma_i). \tag{5.2}$$

Just as in Proposition 3.2.1, if Olivia always hops from α to the open position of highest α-weight, she can guarantee the α-weight of the game states just prior to her turn is non-increasing over time. That is $w_\alpha(\sigma_i) \geq w_\alpha(\sigma_{i+1})$. To prove that the strategy of hopping from α to the open point of highest α-weight is a drawing strategy for Olivia on a given nested board we need to find a value N for which

$w_\alpha(\sigma_N) < 1$ on that nested board. We begin by calculating $w_\alpha(\sigma_1)$ for nested boards, again restricting our interest to nested boards of planes whose components are finite planes $AG(2, q)$ and/or $PG(2, q)$.

On his first turn Xavier has two distinct options on any nested board. He may hop on the starting island, or he may hop off of the starting island. Once again due to the automorphism groups of the component boards, we may assume he hops either on island 1 to the game state $\sigma_1 = [\{(1, 2)\}, \{(1, 1)\}]$ or from island 1 to island 2 to the game state $\sigma_1 = [\{(2, 1)\}, \{(1, 1)\}]$. Given these two options, when calculating $w_\alpha(\sigma_1)$ we need to compute two values, of which Xavier's strategy chooses the higher in order to maximize the α-weight of the game state and therefore the amount by which Olivia must decrease its value to ensure a draw.

If Xavier hops on island 1 there are L_s lines with weight 2^{K-2} which contain both occupied positions, $2(L - L_s)$ lines of weight 2^{K-1} which contain exactly one occupied position, and $M - L_s - 2(L - L_s)$ open lines of weight 2^K. That is if Xavier hops on the island he hops to a game state of α-weight

$$w_\alpha(\sigma_1) = \frac{L_s}{2^{K-2}} + \frac{2(L - L_s)}{2^{K-1}} + \frac{M - L_s - 2(L - L_s)}{2^K} = \frac{M + 2L + L_s}{2^K}.$$

If on the other hand Xavier hops off of the starting island, the only difference is that the two occupied positions share L_d lines instead of L_s, thus he hops to a game state of α-weight $\frac{M+2L+L_d}{2^K}$.

Recall for regular nested boards $[\mathcal{M}_1 : \mathcal{M}_2]$, we have $M := m_1 \cdot (m_2)^{k_1}$ winning sets each with $K := k_1 \times k_2$ points. Every point is in $L := l_2 \cdot l_1 \cdot (m_2)^{k_1-1}$ winning sets on the nested board. Also, for nested boards of planes, two points on the same island share $L_s := l_1 \cdot (m_2)^{k_1-1}$ winning sets while two points on two different islands share $L_d := (l_2)^2 \cdot (m_2)^{k_1-2}$ winning sets. Note $L_d = L_s \cdot \frac{l_2^2}{l_1 m_2}$.

For finite geometries $AG(2, q)$ and $PG(2, q)$, $l = q + 1$ and $m = q^2 + q$ or $q^2 + q + 1$. That is $L_d = L_s \cdot \frac{(q_2+1)^2}{(q_1+1)(q_2^2+q_2+\phi(2))}$ where $\phi(2) = 0$ if $\mathcal{M}_2$ is and affine geometry and $\phi(2) = 1$ if it is projective. Given this we note that $L_d \le L_s$ for any nested board $[\mathcal{M}_1 : \mathcal{M}_2]$ whose components are finite geometries, and thus Xavier on his first turn will hop on the starting island to game state $\sigma_1 = [\{(1, 2)\}, \{(1, 1)\}]$

which has α-weight

$$w_\alpha(\sigma_1) = \frac{M + 2L + L_s}{2^K} = \frac{m_2^{k_1-1}[m_1 m_2 + 2l_1 l_2 + l_1]}{2^{k_1 k_2}}. \tag{5.3}$$

Given this initial α-weight, we find threshold values for q_1 and q_2 for which $w_\alpha(\sigma_1) < 1$ when playing on our nested boards of finite affine and projective planes. We will see that starting this early in the game does provide interesting results about game outcomes. In order to consider the game at states σ_2 or σ_n one would need to consider many cases for the configurations of points on the board in question. By beginning with the calculation of threshold values for $w_\alpha(\sigma_1) < 1$ we reduce the number of boards to be considered further.

Recall that just as in Proposition 5.2.1, as the boards $[\mathcal{M}_1 : \mathcal{M}_2]$ with $\mathcal{M}_1$ and $\mathcal{M}_2$ chosen from $AG(2,2), AG(2,3)$, and $PG(2,2)\}$ do not admit drawn states, these boards are first player win when playing the hop-positional game. Now for each of the four board classes we determine for which values q_1, q_2 the α-weight $w_\alpha(\sigma_1)$ is below 1 and thus for which values we can be sure Olivia's strategy guarantees a draw.

Theorem 5.3.1. *The hop-positional game played on* $[AG(2, q_1) : PG(2, q_2)]$ *is a second player draw game for any board such that* $q_2 \geq 7$.

Proof. For the class of boards $[AG(2, q_1) : PG(2, q_2)]$, we have $m_1 = q_1^2 + q_1$, $k_1 = q_1$, $l_1 = q_1 + 1$ and $m_2 = q_2^2 + q_2 + 1$, $k_2 = q_2 + 1$, $l_2 = q_2 + 1$. In evaluating the α-weight $w_\alpha(\sigma_1)$ from equation 5.3, we find that $w_\alpha(\sigma_1) < 1$ for any nested board of this class for which q_2 is at least 7. That is as long as the inner component board is of at least order 7, Olivia can force a draw by always hopping from α to the open point of highest α-weight.

If we consider boards for which q_2 is less than 7 we find that there is a threshold value $\tau(q_2)$ for which the game is also a draw. This information can be found in the following table.

q_2	2	3	4	5
$\tau(q_2)$	63	35	13	5

That is for example the game on $[AG(2, q_1) : PG(2, 4)]$ is a second player draw for any $q_1 \geq 13$ where Olivia's strategy is to hop from α, the position of her newest O, to the open position of highest α-weight. Note this does not mean that Xavier has a winning strategy for all boards $[A(2, q_1) : P(2, 4)]$ where $2 \leq q_1 \leq 12$. □

Theorem 5.3.2. *The hop-positional game played on the nested board* $[PG(2, q_1) : PG(2, q_2)]$ *is a second player draw game for any board such that* $q_2 \geq 5$.

Proof. For the class of boards $[PG(2, q_1) : PG(2, q_2)]$,, we have the values $m_1 = q_1^2 + q_1 + 1$, $k_1 = q_1 + 1$, $l_1 = q_1 + 1$ and $m_2 = q_2^2 + q_2 + 1$, $k_2 = q_2 + 1$, $l_2 = q_2 + 1$. Using these values in equation 5.3, we find that $w_\alpha(\sigma_1) < 1$ for any nested board of this class for which q_2 is at least 5. That is if the inner component is a projective plane of order at least 5, then Olivia's strategy of always hopping from α to the open point of highest α-weight is a drawing strategy.

Once again we can also find threshold values for boards whose inner component has order less than 5:

q_2	2	3	4
$\tau(q_2)$	61	33	11

□

Theorem 5.3.3. *The hop-positional game played on the nested board* $[AG(2, q_1) : AG(2, q_2)]$ *is a second player draw game if* $q_2 \geq 8$.

Proof. For nested boards of the class $[AG(2, q_1) : AG(2, q_2)]$, we know $m_1 = q_1^2 + q_1$, $k_1 = q_1$, $l_1 = q_1 + 1$ and $m_2 = q_2^2 + q_2$, $k_2 = q_2$, $l_2 = q_2 + 1$. In calculating $w_\alpha(\sigma_1)$ using equation 5.3 we find that $w_\alpha(\sigma_1) < 1$ for $q_2 \geq 8$. That is, Olivia can force a draw by always hopping from α to the open position of highest α-weight if the inner components is an affine plane of order at least 8.

We also find that $[A(2, q_1) : A(2, 5)]$ is a draw for $q_1 \geq 157$ and $[A(2, q_1) : A(2, 7)]$ for $q_1 \geq 4$. The threshold values for $q_2 = 2, 3, 4$ are unknown, but larger than 10^8.

□

Theorem 5.3.4. *The hop-positional game played on the nested board* $[PG(2, q_1) : AG(2, q_2)]$ *is a second player draw game if* $q_2 \geq 7$.

Proof. For nested boards of the class $[PG(2, q_1) : AG(2, q_2)]$, we know $m_1 = q_1^2 + q_1 + 1$, $k_1 = q_1 + 1$, $l_1 = q_1 + 1$ and $m_2 = q_2^2 + q_2$, $k_2 = q_2$, $l_2 = q_2 + 1$. In calculating $w_\alpha(\sigma_1)$ using equation 5.3 we find that $w_\alpha(\sigma_1) < 1$ for $q_2 \geq 7$. For nested boards of this class whose inner components is an affine plane of order at least 7, Olivia can force a draw by always hopping from α to the open position of highest α-weight.

We find for this class that $[P(2, q_1) : A(2, 5)]$ is a draw for $q_1 \geq 157$. Again, the threshold values for $q_2 = 2, 3, 4$ are unknown, but larger than 10^8.

□

We notice that the inner component seems to have more effect on the outcome of the game played on a nested board than does the outer component. In particular summarizing the previous four theorems we get the following general result about nested boards whose components are both finite planes.

Theorem 5.3.5. *Olivia's strategy of hopping from α to the open position of highest α-weight is a drawing strategy for all nested boards* $[A(2, q_1) : P(2, q_2)]$, $[P(2, q_1) : P(2, q_2)]$, $[A(2, q_1) : A(2, q_2)]$, *and* $[P(2, q_1) : A(2, q_2)]$ *where* $q_2 \geq 8$.

Chapter 6

Future Directions

6.1 Positional Games on matroids

A *matroid* is a mathematical structure which generalizes the concept of linear independence in vector spaces. There are numerous equivalent definitions for a matroid, and ways of moving between the definitions called *cryptomorphisms*. In any definition of a matroid we begin with a finite set E of elements called the *ground set* of the matroid. We then consider either a collection of subsets of E or a function on all subsets of E and a list of axioms that must be satisfied. For example if we consider the definition of a matroid in terms of independent sets, we have the following definition from *Matroid Theory* by Oxley [13].

"A *matroid* M is an ordered pair $(E, \mathcal{I})$ consisting of a finite set E and a collection $\mathcal{I}$ of subsets of E satisfying the following three conditions:

(I1) $\emptyset \in \mathcal{I}$.

(I2) If $I \in \mathcal{I}$ and $I' \subseteq I$, then $I' \in \mathcal{I}$.

(I3) If I_1 and I_2 are in $\mathcal{I}$ and $|I_1| < |I_2|$, then there is an element e of $I_2 - I_1$ such that $I_1 \cup e \in \mathcal{I}$. ”

Given a matroid $M = (E, \mathcal{I})$ we call the members of $\mathcal{I} = \mathcal{I}(M)$ the independent sets of the matroid. Any subset of E that is not a member of $\mathcal{I}$ is called a dependent set. Given a matroid $M = (E, \mathcal{I})$, we can identify other important structures, which in turn can be used to define the matroid. For example if we let C be the collection of all minimal dependent sets of the matroid M, we get the set of *circuits* of the matroid. By minimal dependent set we mean a set all of whose proper subsets are independent sets in $\mathcal{I}$. Just as with independent sets we get a set of conditions for a pair (E, C) to define a matroid M, namely:

(C1) $\emptyset \notin C$.

(C2) If C_1 and C_2 are members of C and $C_1 \subseteq C_2$, then $C_1 = C_2$.

(C3) If C_1 and C_2 are distinct members of C, and $e \in C_1 \cap C_2$, then there is a member C_3 of C such that $C_3 \subseteq (C_1 \cup c_2) - e$.

To show that these two definitions in fact give the same matroid structure, we define maps between the collection of independent sets $\mathcal{I}$ and the collection of circuits C and show that the resulting sets satisfy the appropriate axioms and that composition in both orders results in the identity map. This mapping between $\mathcal{I}$ and C forms a cryptomorphism between the two definitions.

We see here two examples of definitions of matroids which lead naturally to a board for positional games. That is for example given a matroid $(E(M), \mathcal{I}(M))$ we can define the board $(E, \mathcal{W})$ by letting $E = E(M)$ and $\mathcal{W} = \mathcal{I}(M)$. That is the goal of the positional game would be to claim an independent set. Similarly we could play to gain a circuit, a basis, a flat, or any of the many other structures inherent in a matroid. Which structure we choose for the winning sets has a marked impact on the outcome of the game. Let us consider for example the rank three whirl $\mathcal{W}^3$ as pictured in Figure 6.1.

The matroid $\mathcal{W}^3$ pictured in Figure 6.1 has ground set $E = \{1, 2, 3, 4, 5, 6\}$. To find the independent

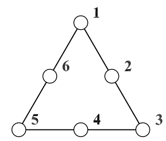


Figure 6.1: The matroid $\mathcal{W}^3$ with ground set $E = \{1, 2, 3, 4, 5, 6\}$. The circuits of this matroid are the three 3-point lines indicated along with any 4-set not containing one of those lines.

sets we consider the geometry of the plane, recalling that the empty set is independent. Let us first imagine that the matroid is sitting in the Cartesian plane, and therefore can be coordinatized in two dimensions, further supposing that no point is at the origin. First note that every point is independent. When thinking about these points using language from vector spaces we would say that each point spans a flat of rank 1. If we had placed a point on the origin its coordinates would be the all zero vector and it would itself be dependent. We call such an element a loop. In playing positional games we will generally play on loop-less matroids to avoid potential issues which arise from the presence of loops. Similarly every pair of points in the matroid is independent as each point has distinct coordinates, that is no point is a scalar multiple of another. A pair of dependent points is pictured as touching and is called a multiple-point. As with loops, multiple points can cause confusion in playing games on matroids and we limit ourselves to simple, loop-less with no multiple points, matroids. A pair of points in this matroid span a line, a rank 2 flat. When we begin to consider collections of three elements of the ground set we see that there are two cases. the set $\{1, 2, 3\}$ forms a line in the matroid, that is the three points together still only span the line. It follows that the third point is a linear combination of the first two and the set is therefore dependent. Similarly the sets $\{1, 5, 6\}$ and $\{3, 4, 5\}$ are dependent. All other three element sets are independent and span the entire rank 3 matroid.

Thus in a positional game played on $\mathcal{W}^3$ where the winning sets are chosen to be independent sets of the matroid, one needs to claim no points, a single point, a pair of points, or any triple other than $\{1, 2, 3\}$, $\{1, 5, 6\}$ or $\{3, 4, 5\}$ to win. Given the first condition for independent sets (I1), which says the empty set is always independent, we can see that choosing independent sets as winning sets is probably not the best idea. Let us consider instead some of the other structure of a matroid an discuss there possible merits in future research into positional games on matroids. When considering boards to play on in Chapter 3 we chose boards with two regularity conditions: first all winning sets have the same cardinality, second every position is in the same number of winning sets. When considering matroids, the collection of bases satisfy the first regularity condition. A basis is a maximal independent set, that is a set all of whose proper super-sets are dependent. One property of basis is that they all have the same cardinality called the rank of the matroid. Thus if we were to define a board using the collection of bases as our winning sets we would have satisfied the condition that all winning sets have the same cardinality. This does not necessarily guarantee the second regularity condition is met, but does give a viable place to begin playing games on matroids. Also if we restrict ourself to loopless matroids we can be sure every element of E is in at least one basis , which is a feature that might also be desirable in a board.

An interesting class of matroids is the class of paving matroids. A *paving matroid* is a matroid all of whose circuits are size r or $r + 1$ where r is the rank of the matroid. If we consider play on such matroids where we let the circuits be the winning sets then we have not quite achieved our first regularity condition, but we are very close. It would be interesting to discover what allowing this slight variation in the cardinality of winning sets does to game play. Other possible structures to consider include lines of rank-2 flats, hyperplanes, co-circuits, and many others. This wealth of available structure makes matroids and interesting place to consider playing positional games.

6.2 Nested Boards

We left many boards unexplored in Chapter 5 and I am interested in trying to improve on the results we obtained. The first thing I would like to attempt is to extend a generic game further by one turn for each player in order to determine if playing until prior to Olivia's second turn will improve the bounds we found. This will require analysis of several cases, and may not lead to better bounds, but it might help for small outside component boards to improve the know threshold values. I would also like to consider some of the small threshold boards to determine if they are actually the dividing board or not.

We should note that this can only be done for those boards whose components have know structure, and would most likely require the use of a computer, but it is possible we can solve the game in specific instances.

I am also interested in the fact that the α-weight of the game state σ_1 has a quadratic factor in q_1 for fixed values of q_2. That is the sequence is not monotonic for some values of q_2. This leads tot he question of whether the smallest boards might in fact be second player draw boards, followed by some middle boards which are first player win before reaching the previously found threshold for all large boards being second player draw. The actual α-weights are large enough that I doubt this will be the case, but it is something that I would like to explore.

I define nested boards using two components, replacing the points of the outer component with copies of the inner component. I am interested in studying mixed nested boards where the positions of the outer component can be replaced by different inner components. The study of such boards would need to be attacked in a case-by-case way, but there might be some interesting patterns which emerge based on the types of boards used. For example if the outer component along with all of the inner components are chosen from among $AG(2,2), AG(2,3), PG(2,2)$ we still know that any positional game played on such a board must be first player win using the fact that none of the boards admit a draw. However, whit if we allow one inner components to be larger? Two? Etc. At what point does the balance tip between first player win and second player draw. Does the answer depend on which larger board is chosen? How does the placement of the inner boards effect the game outcome? These and other questions about mixed nested boards are questions I would like to explore in the future.

6.3 More Variations of Tic-Tac-Toe

We have been playing new positional games by changing the boards and the method of play, but what other variations might one consider? What happens if a game is played by three players instead of two? This variation is clear when placing pieces in open positions in the traditional way, but what variation should we employ if we want three players to hop? Who has the advantage in such a game? If players 2 and 3 play together against player one does this change the outcome of games on boards which were previously first player win? Is this different in any way from letting the second player play more than once on each turn? What happens if we do allow the second player to play more than once? twice? At what point does the second player actually gain a winning advantage?

What happens if we allow the two players to have different goals? That is what happens if we define two collections $\mathcal{W}_1$ and $\mathcal{W}_2$, and play a game on the board $(E, \mathcal{W}_1, \mathcal{W}_2)$ where player i wins by claiming an entire set from $\mathcal{W}_i$? This is reminiscent of the Shannon Switching Game in which one player attempts to create a path between two distinguished vertices of a graph and one player attempts to complete a cut disconnecting the two distinguished vertices.

So far we have considered planar boards, but we could certainly play in higher dimensions. For example what if we play the game on $AG(3,q)$ or $PG(3,q)$? Should the winning sets still be points on a line, or should they be the points in a plane? Can we determine the game outcome for each of these choices? One can find the 3-dimensional Tic-Tac-Toe game, however it is designed as nine pegs in a traditional 3×3 grid and you slide balls onto the pegs. In this way you are forced to play to the lowest position on a peg. What happens if you are allowed to play to any position of the board? What about hopping on this board? What happens if we play either game on larger $n \times n \times n$ boards where winning sets are n-point lines? Can we determine a strategy for these games?

I would like to explore these and other variations and extensions to my research in the future.

Guide to the Appendices

There are four appendices attached to this dissertation. Each contains one or more MATLab codes used throughout the thesis.

Appendix A contains codes used in proving that the hop-positional game on $AG(2,4)$ is a first player win. The first code contains the structure of $AG(2,4)$ and generates the vectors XAVIER and OLIVIA which store the positions occupied by both players. The second code directs game play, by determining how each player will make decisions. This code calls on two functions (one for Xavier and one for Olivia) which calculate the game weight following allowed hops for each player.

Appendix B contains four MATLab codes used in showing that Olivia can force a draw using the strategy of hopping from the position α of the newest O on the board to the open point of highest α-weight. The first code contains the structure of the board, the second the game, and the last two calculate the game weights for each player's options.

Appendix C contains the MATLab codes used in playing the game "Fire and Ice"™ using weight functions to make decisions. As before the first code contains the board structure, the second directs the game, and the last two are used to calculate game values for each player. These last two functions only calculate the game values for slides and jumps which can legally be made at that point in the game.

Appendix D contains a function called LineWeights which is called by every other code contained in these appendices. The purpose of LineWeights is to calculate the weight of each line based on the requirements of the specific player and the game. It can be used generically by all of the other codes as they contain the necessary information to customize its use.

The function LineWeigths is required to run the codes in any of the other appendices. However, using

LineWeights each other appendix could be run independent of the others.

Appendix A

AG(2, 4) is first player win

This MATLab code contains the structure of $AG(2,4)$ needed to play a positional game on the board.

```
% Variable input for AG(2,4)


%The following are the lines of AG(2,4) in their parallel classes. They are
%derived from the horizontal and vertical lines in a 4 by 4 grid and a
%set of three mutually orthogonal 4 by 4 latin squares.

%  [1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6]
H1=[1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0]';
H2=[0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0]';
H3=[0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0]';
H4=[0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1]';
%  [1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6]
V1=[1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0]';
```

```
V2=[0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0]';
V3=[0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0]';
V4=[0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1]';
%  [1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6]
R1=[1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1]';
R2=[0 1 0 0 1 0 0 0 0 0 0 1 0 0 1 0]';
R3=[0 0 1 0 0 0 0 1 1 0 0 0 0 1 0 0]';
R4=[0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0]';
%  [1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6]
S1=[1 0 0 0 0 0 1 0 0 0 0 1 0 1 0 0]';
S2=[0 1 0 0 0 0 0 1 0 0 1 0 1 0 0 0]';
S3=[0 0 1 0 1 0 0 0 0 1 0 0 0 0 0 1]';
S4=[0 0 0 1 0 1 0 0 1 0 0 0 0 0 1 0]';
%  [1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6]
T1=[1 0 0 0 0 0 0 1 0 1 0 0 0 0 1 0]';
T2=[0 1 0 0 0 0 1 0 1 0 0 0 0 0 0 1]';
T3=[0 0 1 0 0 1 0 0 0 0 0 1 1 0 0 0]';
T4=[0 0 0 1 1 0 0 0 0 0 1 0 0 1 0 0]';


%The matrix Lines stores all of the lines created above in a single matrix
%which we can use to check for wins by either Xavier or Olivia.


Lines=[H1 H2 H3 H4 V1 V2 V3 V4 R1 R2 R3 R4 S1 S2 S3 S4 T1 T2 T3 T4];
[m,n]=size(Lines);
lambda=sum(Lines(:,1));
```

```
%m=number of points, n=number of lines, lambda=number of points/line


%The following creates a vector for Xavier and one for Olivia corresponding
%to the points of the arrangement. A 1 indicates that player has a piece on
%the indicated point. We should note that Xavier and Olivia can't both
%occupy a point, so the two vectors will always be orthogonal for
%legitimate game positions.

OLIVIA=zeros(1,m);
XAVIER=zeros(1,m);

%The game begins with Xavier having one piece on the board. Since the board
%has doubly transitive automorphism group, we may assume the game begins
%with Xavier's piece at point 1. Thus the first coordinate of the vector
%XAVIER is set to 1.

XAVIER(1)=1;
```

The following MATLab code contains the hop positional game played on $AG(2,4)$. Xavier's strategy is to first block Olivia by finding the hop which minimizes the sum of the weights of the lines with respect to Olivia's points. Then among the hops which result in the same low game weight he will maximize the sum of the line weights with respect to his own points. Olivia's strategy is to by a brute force search find the hop which results in the lowest possible game weight with respect to Xavier's points. There are two functions called out by the program which are included after it. It also uses the function LineWeights which can be found in Appendix D.

```
%We play the game on the points of $AG(2,4)$ and show that Xavier can win
```

```
%by always hopping to first block Olivia, and then to move to the open point
%of highest weight with respect to his points and unblocked lines.
%Olivia plays by always moving to a game state of minimum weight which
%contains lines of smallest possible weight. That is she will block high
%weight lines thus reducing the overall weight of the game where possible.
%After that all ties will be broken arbitrarily.
%The matrices XAVIERGame and OLIVIAGame store the positions occupied by
%the two players throughout the game.

XAVIERGame=zeros(m,m);
OLIVIAGame=zeros(m,m);
XAVIERGame(1,:)=XAVIER;
OLIVIAGame(1,:)=OLIVIA;
TieBreakO=0;

for i=1:(m-1)
    %Xavier's turn on odd $i$. He will make the hop to block Olivia, moving
    %to the point of highest $O$-weight. In case of a tie he will consider
    %the point of highest $X$-weight
    if floor(i/2)~=i/2
        XAVIERTEMP=XAVIER;
        OLIVIATEMP=OLIVIA;
        XGW=XavierGVMOX(OLIVIATEMP,XAVIERTEMP,Lines);
        Best=find(XGW==min(min(XGW)));
        if length(Best)==1
            [q,p]=ind2sub(size(XGW),Best(1));
            XAVIER(p)=0;
            XAVIER(q)=1;
```

```
    OLIVIA(p)=1;
    XAVIERGame(i+1,:)=XAVIER;
    OLIVIAGame(i+1,:)=OLIVIA;
else
    t=length(Best);
    Choices=zeros(1,t);
    for k=1:t
        XAVIERTEMP=XAVIER;
        OLIVIATEMP=OLIVIA;
        [q,p]=ind2sub(size(XGW),Best(k));
        XAVIERTEMP(p)=0;
        XAVIERTEMP(q)=1;
        OLIVIATEMP(p)=1;
        LW=LineWeights(XAVIERTEMP,OLIVIATEMP,Lines);
        Choices(1,k)=sum(LW);
    end
    BestChoice=find(Choices==max(Choices));
    Pick=randsample(length(BestChoice),1);
    [q,p]=ind2sub(size(XGW),Best(BestChoice(Pick)));
    XAVIER(p)=0;
    XAVIER(q)=1;
    OLIVIA(p)=1;
    XAVIERGame(i+1,:)=XAVIER;
    OLIVIAGame(i+1,:)=OLIVIA;
end
if max(XAVIER*Lines)==lambda
    disp('Xavier has won on')
    turn=i
```

```
            disp('by completing')
            line=find(XAVIER*Lines==lambda)
            disp('which contains')
            for t=1:length(line)
             points=find(Lines(:,line(t))==1)
            end
            return
        end

        %Olivia plays on even $i$
    else
%        If Xavier has not won, then Olivia should make the hop which
%        results in the lowest possible game weight. She will break ties by
%        blocking strongest lines first. That is if two hops result in the
%        same game weight but have different line weight sequences, she
%        will move to the state with the smaller line weights as this gives
%        her more opportunities to block those lines. That is if she can
%        choose between a state with a line of weight 1/2 versus one with
%        two lines of weight 1/4 she will choose the latter.
        OGW=OliviaGVM(XAVIER,OLIVIA,Lines);
        Best=find(OGW==min(min(OGW)));
        %If there is one best move Olivia should make it
        if size(Best)==size([1])
            [q,p]=ind2sub(size(OGW),Best(1));
            OLIVIA(p)=0;
            OLIVIA(q)=1;
            XAVIER(p)=1;
            XAVIERGame(i+1,:)=XAVIER;
```

```
    OLIVIAGame(i+1,:)=OLIVIA;
    %If more than one hop results in the same low game value,
    %Olivia will consider the weights of the lines that result,
    %choosing a state with small line weights.
else
    Options=zeros(n,length(Best));
    Index=1:length(Best);
    Open=ones(size(OLIVIA))-XAVIER-OLIVIA;
    for j=1:length(Best)
        XAVIERTEMP=XAVIER;
        OLIVIATEMP=OLIVIA;
        [q,p]=ind2sub(size(OGW),Best(j));
        OLIVIATEMP(p)=0;
        OLIVIATEMP(q)=1;
        XAVIERTEMP(p)=1;
        LW=LineWeights(XAVIERTEMP,OLIVIATEMP,Lines);
        Options(:,j)=LW;
    end
    %Sort line weighs highest to lowest in each column (for each
    %option)
    Options=sort(Options,'descend');
    %If all options have the same line weight sequence, then choose
    %a random hop from among the options
    if rank(Options)<=1
        [r,c]=size(Options);
        a=randsample(c,1);
        J=Index(a);
        [q,p]=ind2sub(size(OGW),Best(J));
```

```
        OLIVIA(p)=0;
        OLIVIA(q)=1;
        XAVIER(p)=1;
        XAVIERGame(i+1,:)=XAVIER;
        OLIVIAGame(i+1,:)=OLIVIA;
        %If the options have different line weights, we choose one with
        %smallest weight lines.
    else
        TieBreakO=TieBreakO+1;
        stp=0;
        while stp==0
            T=find(Options(1,:)==min(Options(1,:)));
            %If one option has the smallest max line weight we
            %choose that option
            if size(T)==size([1])
                J=Index(T(1));
                stp=1;
                %If more that one option has the smallest max line
                %weight we consider only those options, removing the
                %other options from consideration
            else
                OTemp=[];
                ITemp=[];
                for k=1:length(T)
                    OTemp=[OTemp Options(:,T(k))];
                    ITemp=[ITemp Index(T(k))];
                end
                %If these options now all have the same line weight
```

```
            %sequence, we choose randomly
            if rank(OTemp)<=1
                [r,c]=size(OTemp);
                a=randsample(c,1);
                J=ITemp(a);
                stp=2;
                %If they are different we remove the first row adn
                %repeat the process of comparing the sequence of
                %line weights.
            else
                [r,c]=size(OTemp);
                Options=OTemp(2:r,:);
                Index=ITemp;
            end
        end
    end
end
[q,p]=ind2sub(size(OGW),Best(J));
OLIVIA(p)=0;
OLIVIA(q)=1;
XAVIER(p)=1;
XAVIERGame(i+1,:)=XAVIER;
OLIVIAGame(i+1,:)=OLIVIA;
if min(OLIVIA*Lines)==1
    disp('Olivia has forced a draw by blocking all lines on')
    turn=i
    return
end
```

```
        end
    end
end

function [XGW]=XavierGVMOX(O,X,L)

%This function calculates the weight of the game state that results
%from each possible hop by Xavier. The weight is stored in a matrix
%which can be searched later to find the best hop option. The Line Weights
%calculated are with respect to $X$ as a blocking set and $O$ as the set
%whose points add weight.
[m,n]=size(L);

%As Xavier wants to minimize the game weight with respect to Olivia's
%points, we will set the initial values high.
XGW=100*ones(m);

Occupied=X+O;
One=ones(size(O));
Open=One-Occupied;



%We want Xavier to consider all possible hops from points he
%occupies to open points. He makes the hop which will result in the lowest
%game value.
for i=1:m
    if X(i)==1
        for j=1:m
```

```
            if Open(j)==1
                XTemp=X;
                XTemp(i)=0;
                XTemp(j)=1;
                OTemp=O;
                OTemp(i)=1;
                [LW]=LineWeights(OTemp,XTemp,L);
                XGW(j,i)=sum(LW);
            end
        end
    end
end
XGW;
```

The following MATLab code calculates the games weights of possible hops for Olivia on her turn.

```
function [OGW]=OliviaGVM(X,O,L)



%Goal here is to calculate the weight of the game states that result from
%each possible hop by Olivia. It considers all of her choices of where to
%hop from and to and stores the weight of the resulting game state in a
%matrix which can be searched for the best choice.
[m,n]=size(L);

%Since Olivia wants to move to a point of minimum weight, we set initial
%values high rather than at zero.
OGW=100*ones(m);
```

```
Occupied=X+O;
One=ones(size(O));
Open=One-Occupied;

LW=LineWeights(X,O,L);



%We want Olivia to consider all possible hops from a point $i$
%which she owns to an open position $j$. We record the sum of the line
%weights if she hops from $i$ to $j$ in the matrix $OGW$. Olivia will then
%choose to hop by examining $OGW$ for the smallest game value.
for i=1:m
    if O(i)==1
        for j=1:m
           if Open(j)==1
               XTemp=X;
               XTemp(i)=1;
               OTemp=O;
               OTemp(j)=1;
               OTemp(i)=0;
               [LW]=LineWeights(XTemp,OTemp,L);
               OGW(j,i)=sum(LW);
           end
        end
    end
end
OGW;
```

Appendix B

AG(2, 5) is second player draw

This MATLab code contains the structure of $AG(2,5)$ needed to play the game.

```
% Variable input for AG(2,5)
%
%The following are the lines of AG(2,5) in their parallel classes. They are
%derived from the horizontal and vertical lines in a 5 by 5 grid and a
%set of four mutually orthogonal 5 by 5 Latin squares.

%  [1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5]
H1=[1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]';
H2=[0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]';
H3=[0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0]';
H4=[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0]';
H5=[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1]';
%  [1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5]
V1=[1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0]';
```

```
V2=[0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0]';
V3=[0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0]';
V4=[0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0]';
V5=[0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1]';
%  [1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5]
Q1=[1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0]';
Q2=[0 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0]';
Q3=[0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0]';
Q4=[0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 1]';
Q5=[0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0]';
%  [1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5]
R1=[1 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 0 1 0 0]';
R2=[0 1 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0]';
R3=[0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 1]';
R4=[0 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0]';
R5=[0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 0 1 0 0 0]';
%  [1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5]
S1=[1 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 1 0]';
S2=[0 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 1]';
S3=[0 0 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 1 0 1 0 0 0 0]';
S4=[0 0 0 1 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 1 0 0 0]';
S5=[0 0 0 0 1 0 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 1 0 0]';
%  [1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5]
T1=[1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1]';
T2=[0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 1 0 0 0 0]';
T3=[0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 1 0 0 0]';
T4=[0 0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0]';
T5=[0 0 0 0 1 1 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 1 0]';
```

```
%The matrix Lines stores all of the lines created above in a single matrix
%which we can use to check for wins by either Xavier or Olivia.

Lines=[H1 H2 H3 H4 H5 V1 V2 V3 V4 V5 Q1 Q2 Q3 Q4 Q5 R1 R2 R3 R4 R5 S1 S2 S3 S4 S5 T1 T2
[m,n]=size(Lines);
lambda=sum(Lines(:,1));
%m=number of points, n=number of lines, lambda=number of points/line
%
%
%The following creates a vector for Xavier and one for Olivia corresponding
%to the points of the arrangement. A 1 indicates that player has a piece on
%the indicated point. We should note that Xavier and Olivia can't both
%occupy a point, so the two vectors will always be orthogonal for
%legitimate game positions.

OLIVIA=zeros(1,m);
XAVIER=zeros(1,m);

%The game begins with Xavier having one piece on the board. Since the board
%has doubly transitive automorphism group, we may assume the game begins
%with Xavier's piece at point 1. Thus the first coordinate of the vector
%XAVIER is set to 1.
XAVIER(1)=1;
```

The following MATLab code plays the hop-positional game on $AG(2,5)$ where Olivia will hop from α to the position of highest α-weight and Xavier will counter by moving to the point of highest β-weight choosing where to hop from by considering all options and moving to the game state of highest β-weight. The two functions called out follow this code and the LineWeights function can be found in Appendix D.

```
% We play the game on the points of $AG(2,5)$ where Olivia will hop from the
% position alpha of the newest $O$ to the open position O_i of highest
% remaining alpha-weight. Xavier will counter by finding the hop combination
% $\beta->\gamma$ which results in the highest alpha-weight following his
% turn. As in Proposition 3.1  the alpha-weight function
% defines a non-increasing function over time and if it falls below $1$
% then Olivia has forced a draw. Olivia will break ties in deciding O_i
% arbitrarily using a random selection process. Xavier will break any ties
% by first moving to the point which creates a line (or lines) of highest
% weight and then arbitrarily if the collection of lines weights is the
% same.

XAVIERGame=zeros(m,m);
OLIVIAGame=zeros(m,m);
XAVIERGame(1,:)=XAVIER;
OLIVIAGame(1,:)=OLIVIA;
GWVector=-ones(1,m);


for i=1:(m-1)
```

```
%Xavier plays on odd $i$ and Olivia on even $i$
if floor(i/2)~=i/2
    XAVIERTEMP=XAVIER;
    OLIVIATEMP=OLIVIA;
    Alpha=zeros(size(OLIVIA));
    PW=PointWeights(XAVIERTEMP,OLIVIATEMP,Lines);
    Best=find(PW==max(PW));
    if max(PW)==0
        disp('Olivia has forced a draw by blocking all lines')
        i-1
        return
    else
        XAVIERTEMP=XAVIER;
        OLIVIATEMP=OLIVIA;
        q=Best(randsample(length(Best),1));
        Options=zeros(size(XAVIER));
        for j=1:m
            if XAVIER(j)==1
                XAVIERTEMP(q)=1;
                XAVIERTEMP(j)=0;
                OLIVIATEMP(j)=1;
                LW=LineWeights(XAVIERTEMP,OLIVIATEMP,Lines);
                Options(j)=sum(LW);
            end
        end
        Best=find(Options==max(Options));
        p=Best(randsample(length(Best),1));
        XAVIER(p)=0;
```

```
        XAVIER(q)=1;
        Alpha(p)=1;
        XAVIERGame(i+1,:)=XAVIER;
        OLIVIAGame(i+1,:)=OLIVIA+Alpha;
    end
    if max(XAVIER*Lines)==lambda
        disp('Xavier has won on')
        turn=i
        disp('by completing')
        line=find(XAVIER*Lines==lambda)
        disp('which contains')
        for t=1:length(line)
            points=find(Lines(:,line(t))==1)
        end
        return
    end
    %On Olivia's turn she will hop from Alpha to the point of highest
    %alpha-weight.
else
    PW=PointWeights(XAVIER+Alpha,OLIVIA,Lines);
    Best=find(PW==max(PW));
    p=find(Alpha==1);
    q=Best(randsample(length(Best),1));
    OLIVIA(q)=1;
    XAVIER(p)=1;
    Alpha=zeros(size(OLIVIA));
    XAVIERGame(i+1,:)=XAVIER;
    OLIVIAGame(i+1,:)=OLIVIA;
```

```
        LW=LineWeights(XAVIER,OLIVIA,Lines);
        GWVector(i+1)=sum(LW);
        if min(OLIVIA*Lines)==1
            disp('Olivia has forced a draw hopping from alpha on turn')
            i
            return
        end
    end
end


function [XGW]=XavierGVM(X,O,L)


%This function calculates the weight of the game state that results
%from each possible hop by Xavier. The weight is stored in a matrix
%which can be searched later to find the best hop option.
[m,n]=size(L);


%As Xavier wants to maximize the game weight, we set the initial
%values at zero.
XGW=zeros(m);


Occupied=X+O;
One=ones(size(O));
Open=One-Occupied;


LW=LineWeights(X,O,L);
```

```
%We want Xavier to consider all possible hops from points he
%occupies to open points. He make the hop which will result in the highest
%game value.
for i=1:m
    if X(i)==1
        for j=1:m
           if Open(j)==1
               XTemp=X;
               XTemp(i)=0;
               XTemp(j)=1;
               OTemp=O;
               OTemp(i)=1;
               [LW]=LineWeights(XTemp,OTemp,L);
               XGW(j,i)=sum(LW);
           end
        end
    end
end
XGW;

function [OGWA]=OliviaGVMalpha(X,O,A,L)

%Goal here is to calculate the resultant game value after Olivia hops from
%alpha to each of the open points on the board to determine which results
%in the alpha-weight.
[m,n]=size(L);

%Since Olivia wants to move to a point of minimum weight, we set initial
```

```
%values high rather than at zero.
OGWA=100*ones(m,1);

Occupied=X+O+A;
One=ones(size(O));
Open=One-Occupied;

%We want Olivia to consider all possible hops from a point alpha
%to an open position $j$. We record the sum of the line
%weights if she hops from $i$ to $j$ in the vector $OGW$. Olivia will then
%choose to hop by examining $OGW$ for the smallest game value.



XTemp=X+A;
for j=1:m
    if Open(j)==1
        OTemp=O;
        OTemp(j)=1;
        [LW]=LineWeights(XTemp,OTemp,L);
        OGWA(j)=sum(LW);
    end
end
OGWA;
```

Appendix C

“Fire and Ice” first player win strategy

The following MATLab code generates the Lines matrix for “Fire and Ice”™ using the lines of $PG(2,2)$. There are 49 points and 2401 lines in the board for “Fire and Ice”™ .

```
a=[1 0 0 0 1 1 0 ];
b=[1 0 0 1 0 0 1 ];
c=[1 1 1 0 0 0 0 ];
d=[0 0 1 0 0 1 1 ];
e=[0 0 1 1 1 0 0 ];
f=[0 1 0 0 1 0 1 ];
g=[0 1 0 1 0 1 0 ];
L=[a' b' c' d' e' f' g'];

Lines=zeros(49,2401);
Temp=zeros(49,1);
p=0;
for i=1:7
```

```
    v=L(i,:);
for k=1:7
for j=1:7
for l=1:7
t=0;
Temp=zeros(49,1);
for m=1:7
    if v(m)==1
        t=t+1;
        if t==1
            Temp((7*m-6):7*m)=L(j,:);
        elseif t==2
            Temp((7*m-6):7*m)=L(k,:);
        elseif t==3
            Temp((7*m-6):7*m)=L(l,:);
        end
    else
        Temp((7*m-6):7*m)=zeros(7,1);
    end
end
p=p+1;
Lines(:,p)=Temp;
end
end
end
end
```

```
[m,n]=size(Lines);
lambda=sum(Lines(:,1));
%m=number of points, n=number of lines, lambda=number of points/line
%
%
%The following creates a vector for Xavier and one for Olivia corresponding
%to the points of the arrangement. A 1 indicates that player has a piece on
%the indicated point. We should note that Xavier and Olivia can't both
%occupy a point, so the two vectors will always be orthogonal for
%legitimate game positions.

OLIVIA=zeros(1,m);
XAVIER=zeros(1,m);

%The game begins with Xavier having one piece on the board. Since the board
%has doubly transitive automorphism group, we may assume the game begins
%with Xavier's piece at point 1. Thus the first coordinate of the vector
%XAVIER is set to 1.

XAVIER(1)=1;
```

The following code plays the game “Fire and Ice”™ where Xavier moves to the game state of maximum weight and Olivia to the game state of minimum weight. The result of the game is a win for Xavier. The two functions that are called out follow.

```
%We play the game ''Fire and Ice'' and show that Xavier can win
%by always moving to the available game state of highest game weight.
```

```
XAVIERGame=zeros(m,m);
OLIVIAGame=zeros(m,m);
XAVIERGame(1,:)=XAVIER;
OLIVIAGame(1,:)=OLIVIA;
XavierChoiceMatrix=zeros(m,m);

for i=1:48
    %Xavier will make the hop which maximizes the weight of the game state
    %on odd numbered turns
    if floor(i/2)~=i/2
        for k=1:m;
            if XAVIER(k)==1
                Beta=zeros(size(XAVIER));
                Beta(k)=1;
                XGWB=XavierGVMBetaFandI(XAVIER-Beta,OLIVIA,Beta,Lines);
                XavierChoiceMatrix(:,k)=XGWB;
            end
        end
        Best=find(XavierChoiceMatrix==max(max((XavierChoiceMatrix))));
        choice=randsample(length(Best),1);
        [q,p]=ind2sub(size(XavierChoiceMatrix),Best(choice));
        %indicates a hop from p to q will maximize the weight of the game
        %state
        OLIVIA(p)=1;
        XAVIER(p)=0;
        XAVIER(q)=1;
```

```
    XAVIERGame(i+1,:)=XAVIER;
    OLIVIAGame(i+1,:)=OLIVIA;
    %If Xavier has colloected a winning line, then the game is over.
    if max(XAVIER*Lines)==lambda
        disp('Xavier has won on')
        turn=i
        disp('by completing')
        line=find(XAVIER*Lines==lambda)
        disp('which contains')
        for t=1:length(line)
            points=find(Lines(:,line(t))==1)
        end
        return
    end
    %It is Olivia's turn on even numbered turns
else
    XAVIERTEMP=XAVIER;
    OLIVIATEMP=OLIVIA;
    %If Xavier has not won, then Olivia should make the hop which
    %results in the lowest possible game weight. She will break ties by
    %moving to states with weak lines and then arbitrarily.
    OGW=OliviaGVMFandI(XAVIERTEMP,OLIVIATEMP,Lines);
    Best=find(OGW==min(min(OGW)));
    %If there is one best move Olivia should make it
    if size(Best)==size([1])
        [q,p]=ind2sub(size(OGW),Best(1));
        OLIVIA(p)=0;
        OLIVIA(q)=1;
```

```
        XAVIER(p)=1;
        XAVIERGame(i+1,:)=XAVIER;
        OLIVIAGame(i+1,:)=OLIVIA;
        %If more than one hop results in the same lowest game value,
        %Olivia will consider the weights of the lines that result,
        %choosing a state with small line weights, then breaking
        %ties arbitrarily
    else
        Options=zeros(n,length(Best));
        Index=1:length(Best);
        Open=ones(size(OLIVIA))-XAVIER-OLIVIA;
        for j=1:length(Best)
            XAVIERTEMP=XAVIER;
            OLIVIATEMP=OLIVIA;
            [q,p]=ind2sub(size(OGW),Best(j));
            OLIVIATEMP(p)=0;
            OLIVIATEMP(q)=1;
            XAVIERTEMP(p)=1;
            LW=LineWeights(XAVIERTEMP,OLIVIATEMP,Lines);
            Options(:,j)=LW;
        end
        %Sort line weighs highest to lowest in each column (for each
        %option)
        Options=sort(Options,'descend');
        %If all options have the same line weight sequence, then choose
        %a random hop from among the options
        if rank(Options)<=1
            [r,c]=size(Options);
```

```
        a=randsample(c,1);
        J=Index(a);
        [q,p]=ind2sub(size(OGW),Best(J));
        OLIVIA(p)=0;
        OLIVIA(q)=1;
        XAVIER(p)=1;
        XAVIERGame(i+1,:)=XAVIER;
        OLIVIAGame(i+1,:)=OLIVIA;
        %If the options have different line weights, we choose one with
        %smallest weight lines.
    else
        stp=0;
        while stp==0
            T=find(Options(1,:)==min(Options(1,:)));
            if size(T)==size([1])
                J=Index(T(1));
                stp=1;
            else
                OTemp=[];
                ITemp=[];
                for k=1:length(T)
                    OTemp=[OTemp Options(:,T(k))];
                    ITemp=[ITemp Index(T(k))];
                end
                if rank(OTemp)<=1
                    [r,c]=size(OTemp);
                    a=randsample(c,1);
                    J=ITemp(a);
```

```
                    stp=2;
                else
                    [r,c]=size(OTemp);
                    Options=OTemp(2:r,:);
                    Index=ITemp;
                end
            end
        end
    end
    [q,p]=ind2sub(size(OGW),Best(J));
    OLIVIA(p)=0;
    OLIVIA(q)=1;
    XAVIER(p)=1;
    XAVIERGame(i+1,:)=XAVIER;
    OLIVIAGame(i+1,:)=OLIVIA;
  end
 end
end

function [XGWB]=XavierGVMBetaFandI(X,O,B,L)

%The goal of this function is to calculate the weight of the game state if
%Xavier moves from the position indicated in the vector B to any of the
%allowed open positions. That is on the same island or in the same relative
%position.
[m,n]=size(L);

%As Xavier wants to maximize the game weight, we set the initial
```

```
%values at zero.
XGWB=zeros(m,1);

Occupied=X+O+B;
One=ones(size(O));
Open=One-Occupied;

%We want Xavier to consider all possible slides and jumps from beta to open
% points. He makes the move which will result in the highest game value.

beta=find(B==1);
OTemp=O+B;

%let I be the island number and consider the slides from position beta on I

I=floor((beta-1)/7);
for i=1:7
    s=7*I+i;
    if Open(s)==1
    XTemp=X;
    XTemp(s)=1;
    [LW]=LineWeights(XTemp,OTemp,L);
    XGWB(s)=sum(LW);
    end
end
%let p be the position number of beta and consider the jumps from
%position beta
```

```
p=beta-I*7;
for i=1:7
    j=7*(i-1)+p;
    if Open(j)==1
    XTemp=X;
    XTemp(j)=1;
    [LW]=LineWeights(XTemp,OTemp,L);
    XGWB(j)=sum(LW);
    end
end
XGWB;


function [OGW]=OliviaGVMFandI(X,O,L)


%This function calculate the weight of the game state that results from
%all of the possible moves Olivia can make on her turn, either on the same
%island or in the same relative position. The weights are stored in a
%matrix which can be search through to find the best option.
[m,n]=size(L);

%Since Olivia wants to move to a point of minimum weight, we set initial
%values high rather than at zero.
OGW=100*ones(m);

Occupied=X+O;
One=ones(size(O));
```

```
Open=One-Occupied;

LW=LineWeights(X,O,L);



%We want Olivia to consider the slides on the same island $I$ and all of
%the jumps in the same position $p$ for each of the points $i$ she currently
%occupies.



for i=1:m
    if O(i)==1
        I=floor((i-1)/7);
        p=i-I*7;
        for k=1:7
            %slides on island I
            s=7*I+k;
            if Open(s)==1
                XTemp=X;
                XTemp(i)=1;
                OTemp=O;
                OTemp(s)=1;
                OTemp(i)=0;
                [LW]=LineWeights(XTemp,OTemp,L);
                OGW(s,i)=sum(LW);
            end
            %jumps on position p
            j=7*(k-1)+p;
```

```
            if Open(j)==1
                XTemp=X;
                XTemp(i)=1;
                OTemp=O;
                OTemp(j)=1;
                OTemp(i)=0;
                [LW]=LineWeights(XTemp,OTemp,L);
                OGW(j,i)=sum(LW);
            end
        end
    end
end
OGW;
```

Appendix D

The function LineWeights

The function LineWeights calculates the weights of the lines in a board whose lines are stored as the columns of *L* at the game state [X,O].

```
function [LW]=LineWeights(X,O,L)

% This is an M file which determines the weights of the lines of a positional
% game played on a board with lines L with Xavier currently
% occupying X and Olivia O. The weight of a line l is zero if O\cap l\neq
% \emptyset and is 2^(-u) otherwise where u is the number of open
% positions. The output is a vector whose entries correspond to weights of
% the lines of the board.

%
%
[m,n]=size(L);
%m=number of points, n=number of lines
```

```
Occupied=X+O;
One=ones(size(O));
Open=One-Occupied;
LineWeight=zeros(n,1);

%To determine if a line contains a point occupied by Olivia we multiply
%Olivia's vector by the lines matrix. If Olivia owns a point on a line, the
%corresponding entry of the vector will be non-zero. If the line is
%unblocked then the entry will be zero.
Blocked=O*L;
for j=1:n
    if Blocked(j)==0
        u=sum(dot((L(:,j)),Open));
        LineWeight(j)=2^(-u);
    else
        LineWeight(j)=0;
    end
end
LW=LineWeight;
```

Bibliography

[1] C.I. Mándity A. Csernenszky and A. Pluhár, *On chooser picker positional games*, Discrete Math **309** (2009), 5141–5146.

[2] József Beck, *Combinatorial games: Tic-tac-toe theory (encyclopedia of mathematics and its applications)*, Cambridge University Press, New York, 2008.

[3] M. Bednarska and O. Pikhurko, *Biased positional games on matroids*, Eur. J. Combin. **26** (2005), 271–285.

[4] R.H. Bruck and H.J. Ryser, *The nonexistence of certain finite projective planes.* Canadian Journal of Mathematics **1** (1949), 88–93.

[5] Maureen T. Carroll and Steven T. Dougherty, *Tic-tac-toe on a finite plane*, Math Mag **77** (2004), no. 4.

[6] L.H. Theil C.W.H. Lam and S. Swiercz, *The non-existence of finite projective planes of order* 10, Canad. J. Math. **41** (1989), no. 6.

[7] M. Stojakovic D. Hefetz M. Krivelevich and T. Szabó, *A sharp threshold for the hamilton cycle makerbreaker game*, Random Structure Algorithms **34** (2009), 112–122.

[8] John H. Conway Elwyn R. Berlekamp and Richard K. Guy, *Winning ways for your mathematical plays*, Academic Press Inc. London-New York.

[9] P. Erdös and J.L. Selfridge, *On a combinatorial game*, J. Combinatorial Theory Ser. A **14** (1973), 298–301.

[10] K. Kruczek and E. Sundberg, *Potential-besed strategies for tic-tac-toe on the integer lattice with numerous directions*, Electron. J. Combin. **17** (2012), no. 1.

[11] A. Lehman, *A solution to the shannon switching game*, J. Soc. Indust. Appl. Math. **12** (1964), 687–725.

[12] Richard J. Nowakowski Michael H. Albert and David Wolfe, *Lessons in play; an introduction to mathematical game theory*, A K Peters, Ltd, New York, NY, 2007.

[13] J. Oxley, *Matroid theory*, Oxford Press, New York, NY, 1992.

[14] D.J.A Welsh, *Matroid theory*, Academic Press, Inc. New York, 1976.

[15] N. White, *Theory of matroids*, Vol. 26, Cambidge University Press, New York, 1986.