Hang Lung Mathematics Awards
Vol. 6 (2014)

Hang Lung Mathematics Awards 2014

Honorable Mention

Two Methods for Investigating the Generalized Tic-Tac-Toe

Team Members:	Kam Chuen Tung, Luke Lut Yin Lau
Teacher:	Ms. Mee Lin Luk
School:	La Salle College

Hang Lung Mathematics Awards
Vol. 6 (2014), pp. 153–182

TWO METHODS FOR INVESTIGATING THE GENERALIZED TIC-TAC-TOE

TEAM MEMBERS
Kam Chuen Tung, Luke Lut Yin Lau

TEACHER
Ms. Mee Lin Luk

SCHOOL
La Salle College

Abstract. In this paper, we look into the (m, n, k, p, q) game, one of the generalizations of the well-known Tic-Tac-Toe game. The objective of the game is to achieve 'k-in-a-row' with one's pieces before one's opponents does. We use two methods — exhaustion and pairing strategy — to investigate the results of the (m, n, k, p, q) game for several different values of the five parameters.

1. Introduction

1.1. Background

Tic-tac-toe is a simple but well-known game which can be traced back to the first century BC. [1] Two players participate in this game and each player uses a type of pieces (usually crosses or noughts) which are placed on a 3×3 board with 9 equally large squares. The players place one stone on an unoccupied square in turn to achieve a game state in which there are three of their own pieces connected horizontally, vertically, or diagonally. Though a lot of people know how to draw rather easily playing as the second player, some very interesting and much more difficult and complicated games have evolved from this ancient game. Among the games similar to tic-tac-toe, Gomoku and Renju are the most popular ones. They are played on a much larger board, and a player has to 'connect five' to win the game. In these games, various extensive researches have been done and results under perfect play can be determined as early as the second move. 'Perfect play' means that every move made by the players does not change the evaluation (win for B, win for W, or draw) of the position. [See reviewer's comment (2)] As a result, more complicated rules are added to a normal 'connect five' game to create different variants, in order to avoid the thorough analysis performed by previous researches.

Gomoku, Renju, and other games arising from tic-tac-toe are members of generalized (m, n, k) games [2] — connect-k games playing on a $m \times n$ board — which can be further developed into the (m, n, k, p, q) game, which is first introduced by I-Chen Wu and Dei-Yen Huang [3]. In the (m, n, k, p, q) game, each player places p stones on the board each turn, except for the first turn of the first player, in which q stones are placed. The generalisation of the (m, n, k, p, q) game has led to a massive number of possibilities of game states.

In this project, we will explore different specific variations with $k \geq 3$, as variations with $k < 3$ are trivial. Some general results will be achieved as well, which can help to solve certain variations of the game with similar characteristics.

1.2. Game rules

The (m, n, k, p, q) game is played between two players on a $m \times n$ board, where m is the number of columns and n is the number of rows. The first player (B) uses black stones, and the second player (W) uses white stones. B first places q stones, each of them on a formerly unoccupied square on the board. Then, the players alternately place p stones every turn, in the same manner.

The game terminates if one of the following two conditions is fulfilled:

1. If there are k stones of the same colour being placed on k consecutive squares horizontally, vertically or diagonally, then a *k-in-a-row* is said to have been achieved, and the game ends. The player who achieves a *k-in-a-row* wins the game.
2. If the whole board is occupied and neither player is able to achieve *k-in-a-row*, then the game ends and is declared as a draw.

1.3. Project Outline

This paper consists of six chapters.

Chapter 1 (the current chapter) is a brief introduction of our project.

In Chapter 2, we state and prove a useful theorem, to eliminate the possibility that the second player wins when $p \leq q$. [See reviewer's comment (3)]

In Chapter 3, we use exhaustion to predict the outcome of the game for specific sets of values of (m, n, k, p, q), given that both players play 'perfectly'.

In Chapter 4, we introduce the 'pairing strategy' and apply it in the case $p = q = 1$, i.e., each player only places one stone each turn. The pairing strategy is employed by the second player to secure a draw, for he/she will never win when $p \leq q$. We investigate the necessary condition of applying the pairing strategy and obtain some interesting results.

In Chapter 5, we evaluate the two methods (exhaustion and pairing) and summarize the results we have obtained.

Finally, in Chapter 6, we suggest some areas of the (m, n, k, p, q) game for further research.

1.4. Definition & Notations

In this table, we define several terms and introduce several notations that will appear in the paper.

Term/Notation	Definition
B	B stands for the first player.
W	W stands for the second player. [See reviewer's comment (4)]
Forced win for B	It refers to a game state in which B has a strategy ensuring a win, regardless of how W plays. The definition of 'forced win for W' is similar.
Forced draw for W	It refers to a game state in which W has a strategy which can prevent B from winning, regardless of how B plays. The definition of 'forced draw for B' is similar.
k-in-a-row	k stones of the same colour are being placed on consecutive squares horizontally, vertically or diagonally.
Irrelevant move	A move which does not change the nature of a certain game state, i.e., unable to parry opponent's threat and unable to create a threat.
Perfect play	Every move made by the players does not change the evaluation (win for B, win for W, or draw) of the position.
Horizontal/vertical/ diagonal pairing	Horizontal pairing is the pairing of two squares on the same row. Vertical and diagonal pairings have similar meanings.
a7, etc.	The letter represents the column number of a square, starting from column a. The integer represents the row number of a square, starting from row 1. For example, a7 means the seventh square of the first column (refer to Figure 1).
1. a7	In B's first turn, B places his/her stone on the square a7.

4. e5	In W's fourth turn, W places his/her stone on the square e5.
$1.a7\dots c3$	In the first round, B places his/her stone on the square a7, which W responses by playing his/her stone on square c3.
$1.a6, a7$	In B's first turn, B places his/her two available stones on squares a6 and a7.
$1.a6/a7$	In B's first turn, B places his/her stone on either square a6 or square a7.
(F)	It refers to a forced move for a player, a move which he/she is compelled to play, or the player will lose the game immediately.
#	It refers to a game state trivially known to be ended in victory for the player who just makes the move.

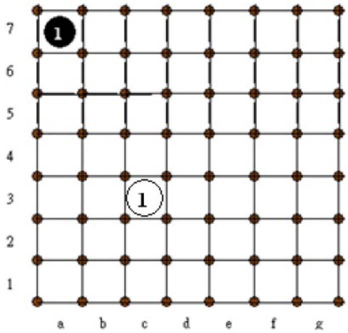

FIGURE 1. B places his/her first stone on the square a7, and W places his/her first stone on the square c3. We represent the sequence of moves as *1. a7... c3*.

In this paper, we assume, without loss of generality, that $m \geq n$.

We also assume that $k \geq 3$ for a nontrivial investigation.

In the main body of the paper, we use *italics* for sequences of moves, but not for square positions, to differentiate the two. Thus, if the description of Figure 1 appears in the main body, it will be formatted in this way: "B places his/her first stone on the square a7, and W places his/her first stone on the square c3. We represent the sequence of moves as *1. a7 ... c3*".

2. A Useful Theorem

Before we present our findings, we introduce a useful theorem which eliminate the possibility that the second player wins when $p \leq q$.

Theorem 1. *If $p \leq q$, then W does not have a forced win.*

Proof. In certain games, the player who makes the first move will not lose. Assume that the second player has a winning strategy. Then the first player can make an irrelevant move. As a result, the first player will 'become' the second player and can copy the winning strategy of the second player.

In the (m, n, k, p, q) game, where $p \leq q$, B can place q stones on the first turn, which can be considered as 'more than or equal to a move', as a player can normally place only p stones on a turn. Since the first move cannot be harmful to B, he/she can choose to make an irrelevant move if W has a winning strategy. Then, B will have a win in hand as he/she is now using W's winning strategy. Therefore, it can be proved by contradiction that if $p \leq q$, B can always at least draw the game with perfect play on both sides, i.e., W does not have a forced win. □

3. Method 1: Exhaustion

The first method we used to analyse the (m, n, k, p, q) game is exhaustion. Exhaustion is used in finding winning/drawing strategies of a certain player. If a player has a forced win (a forced draw), he/she can win (draw) the game regardless of his/her opponent's stone placement. By exhausting all possible responses of a player's opponent in a certain game state, we can help the player find different 'lines', or sequences of moves, to ensure a win or a draw.

We understand that the overflow of symbols may obstruct the flow of the paper and affect the reader's interest in reading it. Therefore, we include details of our analysis in the appendix, but not in the main body, of the paper.

We first explore the extremely simple $(3, 3, 3, 1, 1)$ game, known as the famous game tic-tac-toe. By Theorem 1, W cannot win the game. Therefore, W should be content with a draw, which is indeed the well-known outcome of the game.

To prove that $(3, 3, 3, 1, 1)$ ends in a draw under perfect play, we have to find appropriate responses for W against all possible moves by B.

Theorem 2. $(3, 3, 3, 1, 1)$ *is a forced draw for W.*

Proof. By symmetry, B has three distinct opening moves: *1. b2, 1. a2* and *1. a3.* The appropriate response by W for each of the three opening moves is included in Appendix A.

For example, *W* should play *1. ...a1* against *1. b2* and *1. ...b2* against the remaining two opening moves. See Figure 2 for the visualisation of the three W responses.

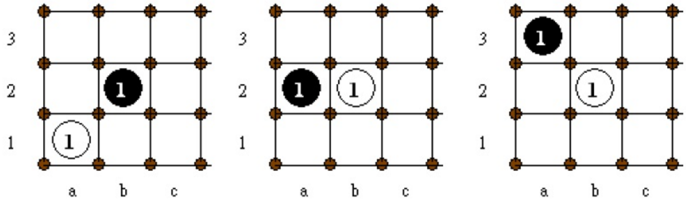

FIGURE 2. W's responses against B's three distinct opening moves: *1....a1* against *1.b2*, *1. ...b2* against *1. a2*, and *1. ...b2* against *1. a3.*

By finding drawing lines for W against all possible moves of B, we see that W has a drawing strategy against B in the $(3, 3, 3, 1, 1)$ game. □

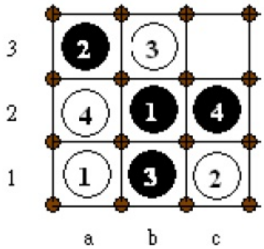

FIGURE 3. The line *1. b2...a1 2. a3...c1(F) 3. b1(F)...b3(F) 4. c2...a2(F).*

Figure 3 demonstrates one of the drawing lines of W in the $(3, 3, 3, 1, 1)$ game. We hope that the figure can help the reader to better understand the move symbols and the diagrams that appear in the paper.

Using exhaustion, we are able to prove several theorems, which are useful in predicting the outcomes of different (m, n, k, p, q) games under perfect play and finding winning/drawing strategies. We mainly worked on $k = 3$ and $k = 4$ because it is too difficult to perform exhaustion on cases with bigger values of k.

Theorem 3. $(4, 4, 4, 1, 1)$ *is a forced draw for W.*

Proof. After exploring a few cases by hand, we are convinced that $(4, 4, 4, 1, 1)$ is a forced draw for W. Figure 4 shows some of the game states we found, that will end in a draw.

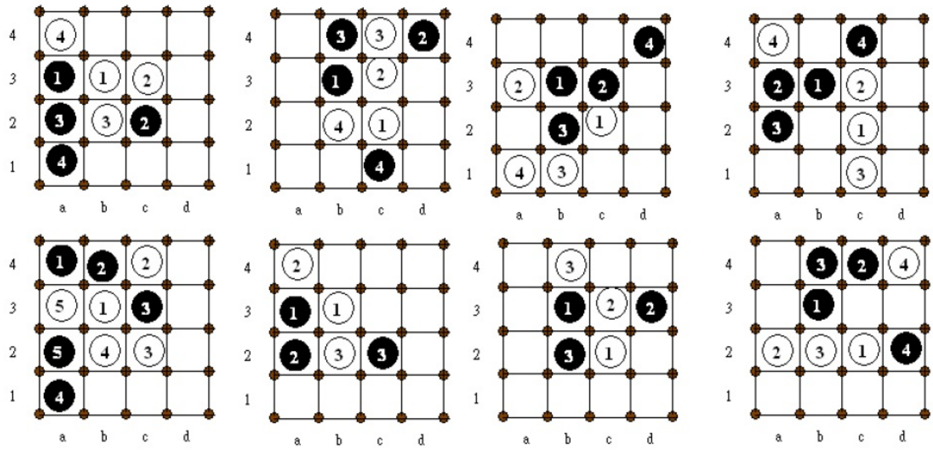

FIGURE 4. Several drawn game states of the $(4, 4, 4, 1, 1)$ game

However, it is very difficult to list all possibilities by hand. Therefore, an auxiliary C++ computer program is created for verification of the theorem. By exploring the whole game tree of the $(4, 4, 4, 1, 1)$ game, the program helps both players find the best move in a certain game state in his/her turn (winning better than drawing; drawing better than losing). The source code of the program can be found in **Appendix B**.

Here are some details of the program. The game states are represented by base-3 numbers: each digit of a base-3 number represents the three states of a certain square: 0 for 'empty', 1 for 'occupied by B', 2 for 'occupied by W'. Each game state has a value: 0 for a forced draw, 1 for a forced win by B, -1 for a forced win by W. In fact, the program conveys that the empty board has a value of 0, i.e., the $(4, 4, 4, 1, 1)$ game is a forced draw.

With the help of the program, we are able to exhaust all possible moves by B and find a forced draw for W in the $(4, 4, 4, 1, 1)$ game. □

In fact, proving that certain (m, n, k, p, q) games are forced draws are much harder than proving that certain (m, n, k, p, q) games are forced wins. We even need a program to verify, rigorously, that $(4, 4, 4, 1, 1)$ is a forced draw for W, because the workload is simply too large.

The remaining results are all proofs that certain (m,n,k,p,q) games are forced wins, though not necessarily for B.

Theorem 4. $(m,n,3,1,1)$ *is a forced win for B, when* $m>3$ *and* $n\geq 3$.

Proof. We would like to use a 4×3 board for demonstration of the winning strategy for B. This strategy, however, can be applied to other boards meeting the above condition as well.

B can play *1. b2*, a move that threatens *2. c2*, which results in a three-in-a-row on his/her third move. Therefore, W can only choose from *1. ... a2, 1. c2*, and *1. ... d2*. Each of W's possible responses can be answered with the strategy shown in Figure 5.

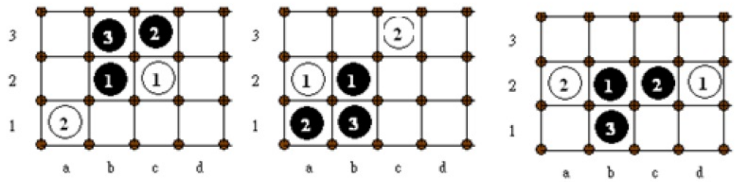

FIGURE 5. Three winning lines of B in the $(m,n,3,1,1)$ game ($m>3$ and $n\geq 3$).

Since W's moves are forced and none of his/her pieces are inter-connected, the size of the board does not affect the feasibility of the above strategy. Therefore, $(m,n,3,1,1)$ is a forced win for B, when $m>3$ and $n\geq 3$. [See reviewer's comment (5)] ☐

Theorem 5. $(5,5,4,2,2)$ *is a forced win for B.*

Proof. In this game, each player places two stones on two previously unoccupied squares on his/her turn. A winning strategy for B is to play *1. b3,c3*. Since B now threatens a four-in-arow along the third row, which can be a3-b3-c3-d3 or b3-c3-d3-e3. If W plays *1. ... a3,e3*, then *2. b4, d2#*, and W wins. Therefore, as the only available option, W must place one of his/her stones on d3 on his/her first turn. [See reviewer's comment (6)]

If W plays an irrelevant move, B can play *2. c2, c4#*, threatening *3. a2, d5* and *3. c1/c5*. With only two stones available, W cannot parry all three threats: c1, c5 and a2/d5. Therefore, *W* cannot play an irrelevant move.

Figure 6 shows the possibilities of W's placement of his/her second stone and the corresponding winning strategies of B.

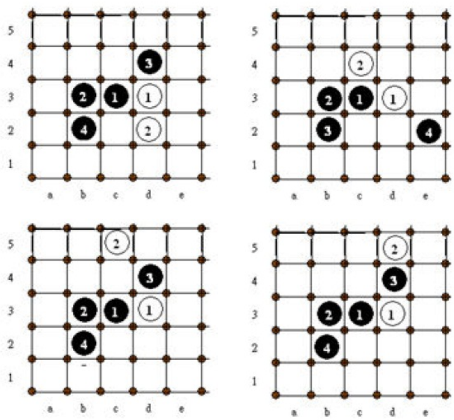

FIGURE 6. W's possible second stone placements and the corresponding winning lines for B in the $(5, 5, 4, 2, 2)$ game.

As B has a winning strategy against all of W's possible responses, the $(5, 5, 4, 2, 2)$ game is a forced win for B. □

Theorem 6. $(6, 6, 4, 1, 1)$ *is a forced win for B.*

Proof. B can win by placing his/her first stone on d4 and then following winning lines for different responses by W. B's winning lines for the responses

1.. . . c3, 1. . . . d3, 1. . . . e3, 1. . . . e4 and *1. . . . e5*

are shown in Figure 7. All moves by W are the best moves.

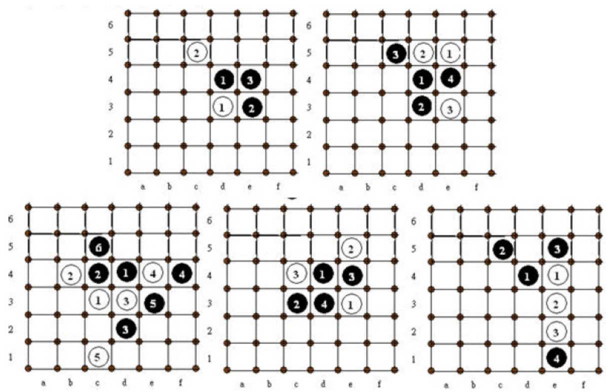

FIGURE 7. Some of B's winning lines in the $(6, 6, 4, 1, 1)$ game.

The full analysis file is included in Appendix C.

The winning lines of B, as shown in the full analysis file, indicate that $(6,6,4,1,1)$ is a forced win for B. ☐

Theorem 7. $(6,6,4,2,1)$ *is a forced win for W.*

Proof. In the $(6,6,4,2,1)$ game, B places one stone on a square on his/her first turn, and each player subsequently places two stones on two previously unoccupied squares in his/her turn.

If B plays *1. c4*, W should play *1. … d3,d4*, threatening to achieve a four-in-a-row along the d-column. Therefore, B has to place one of his/her stones on d2 or d5, and the other one on another square on the other side of the d-column.

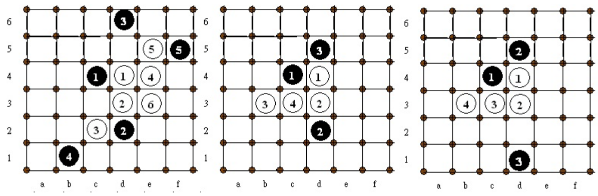

FIGURE 8. Winning lines of W after *1. c4* in the $(6,6,4,2,1)$ game

It can be seen, from Figure 8, that W can win if B places the first stone on one of the four central squares. Combined with W's winning lines in Figure 9 and Figure 10 (which is shown on the next page) for other possible first moves of B, it can be concluded that the $(6,6,4,2,1)$ game is a forced win for W. ☐

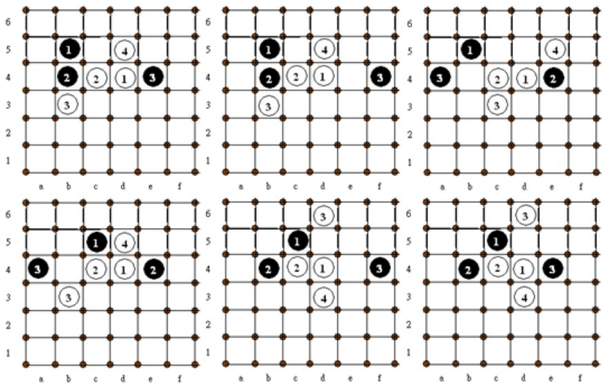

FIGURE 9. Winning lines of W after *1. b5/ c5* in the $(6,6,4,2,1)$ game

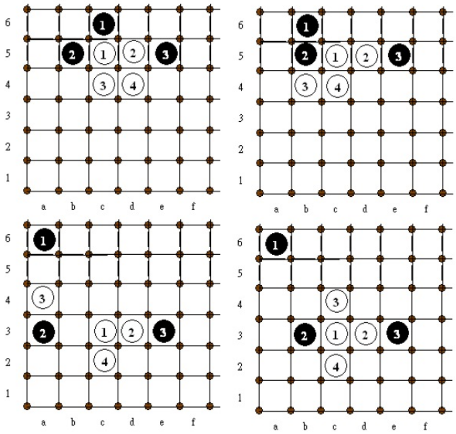

FIGURE 10. Winning lines of W after *1. a6/b6/c6* in the $(6,6,4,2,1)$ game. B's second and third stones can be placed in other squares on the same row. However, the variations do not affect W's winning lines.

Exhaustion may seem to be easy and straight-forward. Yet, there are many games solvable by exhaustion, but are unsolved due to the enormous number of game states. One examples of such games is Chess.

Volodymyr Sobotovych claimed to have proved in 2003 that $(9,4,4,1,1)$ is a forced win [4], although the work is not peer-reviewed. [See reviewer's comment (7)] We originally wished to prove, in this paper, that $(9,4,4,1,1)$ is a forced win. We used the opening line

1. e2. . . f3 2. c2 . . . d2 3. d3

as proposed by Sobotovych, but it seems that W can respond with *3. . . . f1* to ensure a draw, after which B cannot make an irrelevant move (like *4. b3* as we have tried), for W has a brilliant winning line *4. . . . f4 5. f2 (F) . . . e3 6. c1 (F) . . . h3 7. g3 (F) . . . g2 8. i4 (F) . . . e4 9. h1 (F) . . . g4#*. The line is shown in Figure 11.

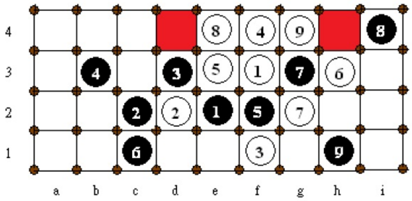

FIGURE 11. The winning line of W after *4. b3.*

Moreover, we have tried other responses by B, and it seems that none of them can yield a forced win after *3. ...f1.* The analysis is included in Appendix D. Therefore, we conjecture that the $(9, 4, 4, 1, 1)$ game may be a forced draw.

4. Method 2: Pairing Strategy

Although exhaustion can be used to determine the result of a game state under perfect play, a more elegant and pragmatic approach is the use of the 'pairing strategy'. The strategy is used by W to ensure a forced draw and works only when $p = q = 1$. The algorithm of the pairing strategy is described below.

Algorithm 4.1

The pairing strategy

Step 1 Before the start of the game, W pairs some of the squares on the board with each other, such that every k-in-a-row contains a set of the paired squares. The specific pairing pattern depends on the board configuration.

Step 2 In each of B's turn, B places a stone on a square, say, S_1. If S_1 is paired with another square (S_2) in Step 1, and S_2 is empty, then W places his/her stone on S_2. Otherwise, W can place it on any square.

Algorithm 4.1 ensures that every k-in-a-row contains a set of paired squares, and every set of paired squares contains at least one white stone. Therefore, W can apply Algorithm 4.1 to force a draw.

One of the most famous results on the $(m, n, k, 1, 1)$ game is the application of the pairing strategy to prove that $(\infty, \infty, 9, 1, 1)$ is a forced draw.

Theorem 8. $(\infty, \infty, 9, 1, 1)$ *is a forced draw.*

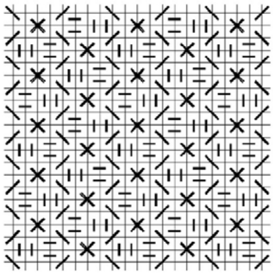

FIGURE 12. The pairing pattern for $(\infty, \infty, 9, 1, 1)$.

Proof. Figure 12 illustrates how the squares can be paired [4]. Note that the pairing pattern is periodic, so it can be extended to an infinite board. It can also be applied to all board sizes.

Then, by applying Algorithm 4.1, W can force a draw. □

We would like to use a similar approach to solve the $(m, n, k, 1, 1)$ game for a few other values of m, n and k.

Before that, we would like to state a necessary condition of the successful application of the pairing strategy

Lemma 9. *For a row, column or diagonal containing l squares, at least* $2\left\lfloor\frac{l-1}{k-1}\right\rfloor$ *squares have to be paired to cover all k-in-a-rows.*

Proof. We use greedy algorithm to pair up the squares. We consider all the k-in-a-rows from left to right (rotate by 90 degrees for columns). If the current k-in-a-row does not contain a set of paired squares, we greedily pair up its two rightmost squares. As a result, new pairs are formed when we consider the $1^{st}, k^{th}, (2k-1)^{th}, \ldots, [p(k-1)+1]^{th}$ k-in-a-rows. As there are $(l-k+1)$ k-in-a-rows, solving the simultaneous equation

$$\begin{cases} p(k-1)+1 \leq (l-k+1) \\ (p+1)(k-1)+1 > (l-k+1) \end{cases}$$

We have the number of *pairs* of squares
$= p+1 = \left\lfloor\frac{l-1}{k-1}\right\rfloor$

Therefore, the least number of paired squares
$= 2\left\lfloor\frac{l-1}{k-1}\right\rfloor$ □

Using Lemma 9, we now state and prove a necessary condition of the successful application of the pairing strategy.

Theorem 10. *For a certain set of values of (m, n, k), Algorithm 4.1 can be applied to the $(m, n, k, 1, 1)$ game to find a forced draw only if*

$$mn \geq 2\left(n\left\lfloor\frac{m-1}{k-1}\right\rfloor + m\left\lfloor\frac{n-1}{k-1}\right\rfloor + 4\sum_{i=1}^{n-1}\left\lfloor\frac{i-1}{k-1}\right\rfloor + 2(m-n+1)\left\lfloor\frac{n-1}{k-1}\right\rfloor\right).$$

Proof. LHS = number of squares on the board.

There are n horizontal lines of with m squares, so at least $2n\left\lfloor\frac{m-1}{k-1}\right\rfloor$ squares should be paired to cover all the horizontal k-in-a-rows.

There are m vertical lines of with n squares, so at least $2m\left\lfloor\frac{n-1}{k-1}\right\rfloor$ squares should be paired to cover all the vertical k-in-a-rows.

There are four diagonals of with i squares for all $1 \leq i < n$, so at least $8\sum_{i=1}^{n-1}\left\lfloor\frac{i-1}{k-1}\right\rfloor$ squares should be paired to cover all the k-in-a-rows in these short diagonals.

There are $2(m-n+1)$ diagonals with n squares, so at least $4(m-n+1)\left\lfloor\frac{n-1}{k-1}\right\rfloor$ squares should be paired to cover all the k-in-a-rows in these long diagonals.

Adding these terms, we get RHS = the least number of squares that need to be paired.

Therefore, it is necessary that LHS $\geq$ RHS holds, for otherwise the pairing cannot be completed. □

With Theorem 10 in hand, we try to explore some general cases and prove that they are forced draws for some values of k.

Theorem 11. *$(k, k, k, 1, 1)$ is a forced draw for $k \geq 3$.*

Proof. First of all, by Theorem 10, we *may* be able to apply Algorithm 4.1 if $k^2 \geq 2(k+k+0+2)$, or, after simplification, $(k-2)^2 \geq 8$, which gives $k \geq 5$ after solving for k, as $k \in \mathbb{N}$.

$(3, 3, 3, 1, 1)$ is proved a forced draw in Theorem 2.

$(4, 4, 4, 1, 1)$ is proved a forced draw in Theorem 3.

The respective pairing for $(5, 5, 5, 1, 1)$ is shown in Figure 13.

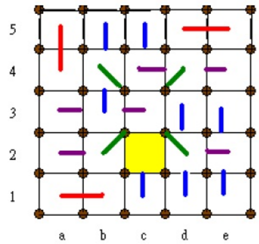

FIGURE 13. The pairing pattern for $(5, 5, 5, 1, 1)$.

Before we proceed, we use Figure 13 to illustrate how the pairing lines are drawn. Red lines refer to pairings of adjacent squares, while other colours refer to pairings of non-adjacent squares. Squares painted yellow are unpaired. The pointing direction of a pairing line indicates whether it is a horizontal, vertical, or diagonal pairing. For example, b4 is paired with d2, c1 is paired with c5, and a3 is paired with c3.

Finally, we describe a pairing pattern for $k \geq 6$. The four 2×2 corners of the board are alternately paired horizontally or vertically. Then, the remaining squares on the first two rows on the top are paired vertically, and those on the first two columns on the left are paired horizontally. Finally, consider the now-empty $(k-4) \times (k-4)$ square board in the centre of the board. Two pairs of squares on opposite corners are paired with each other. See Figure 14 and Figure 15 for the examples $(6, 6, 6, 1, 1)$ and $(7, 7, 7, 1, 1)$.

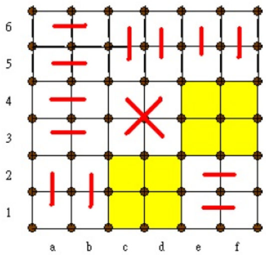

FIGURE 14. The pairing pattern for $(6, 6, 6, 1, 1)$.

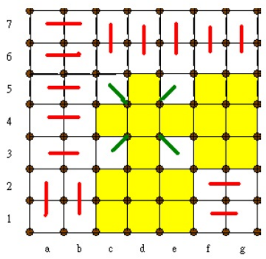

FIGURE 15. The pairing pattern for $(7, 7, 7, 1, 1)$.

As a result, we can apply Algorithm 4.1 to obtain a forced draw for the $(k, k, k, 1, 1)$ game, for $k \geq 5$.

Combining the results, $(k, k, k, 1, 1)$ is a forced draw for $k \geq 3$. □

Theorem 12. $(k+1, k+1, k, 1, 1)$ *is a forced draw for* $k \geq 6$.

Proof. First of all, by Theorem 10, we may be able to apply Algorithm 4.1 if $(k+1)(k+1) \geq 2((k+1)+(k+1)+4+2)$, or $(k-1)^2 \geq 16$. So, $k \geq 5$ is necessary.

However, if $k = 5$, using the corner squares will lead to extra pairing, but the board has no more unpaired squares. So, Algorithm 4.1 can only be used if $k \geq 6$. [See reviewer's comment (8)]

The pairing $(7, 7, 6, 1, 1)$ for is shown in Figure 16.

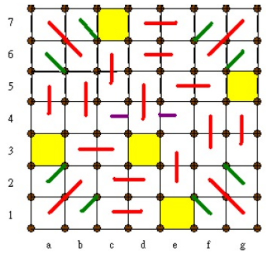

FIGURE 16. The pairing pattern for $(7, 7, 6, 1, 1)$.

For $k \geq 7$, we describe a symmetrical pairing pattern.

For each of the four diagonals with k squares, the two squares on either end of it are paired with each other. For each of the two diagonals with $k+1$ squares, counting from left to right, the first and second squares are paired with each other, so are the last and the second last squares paired with each other.

Then, for the first and second rows, squares are paired vertically except for the four squares on the third last and fourth last columns. The same applies for the last and second last rows. The pattern is similar for other squares on the first and second columns and the last and second last columns. Finally, consider the now-empty $(k-4) \times (k-4)$ square board in the centre of the board. For the 2×2 area on the top-left corner, the squares are paired horizontally, while those on the bottom-right corner are paired vertically. [See reviewer's comment (9)] Refer to Figure 17 and Figure 18 on the next page for demonstrations of the pairing strategy for $(8, 8, 7, 1, 1)$ and $(9, 9, 8, 1, 1)$.

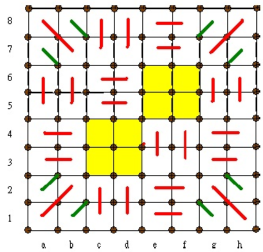

FIGURE 17. The pairing pattern for $(8, 8, 7, 1, 1)$.

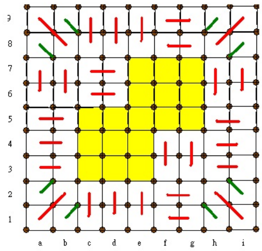

FIGURE 18. The pairing pattern for $(9, 9, 8, 1, 1)$.

In conclusion, $(k+1, k+1, k, 1, 1)$ is a forced draw for $k \geq 6$. □

Theorem 13. $(2k-1, k, k, 1, 1)$ *is a forced draw for* $k \geq 7$.

Proof. First of all, by Theorem 10, we may be able to apply Algorithm 4.1 if $2k^2 - 13k + 2 \geq 2$, which yields $k \geq 7$. The case $k = 7$ is verified, as shown in Figure 19.

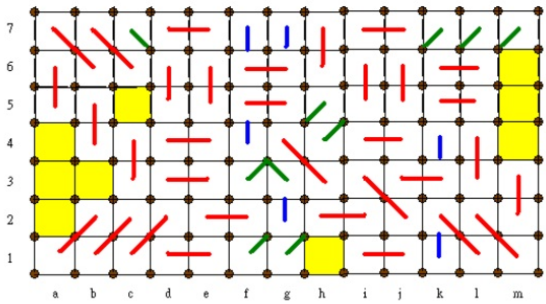

FIGURE 19. The pairing pattern for $(13, 7, 7, 1, 1)$.

Finally, we describe a pairing pattern for $k \geq 8$. The horizontal pairing lines are characterized by its wave-like horizontal pairing lines, one group bounded between column $(k-4)$ and column $(k-1)$, the other bounded between column $(k+1)$ and column $(k+4)$. The diagonal pairing lines are drawn as close to the periphery of the board as possible. However, the unoccupied squares on columns $k-3$, $k-2$, $k+2$, and $k+3$ are to be avoided, for they are preserved for vertical pairing lines. Finally, draw the vertical pairing lines arbitrarily. The 'staircase' pattern is one of the possible ways of drawing them. Figure 20, Figure 21, and Figure 22 are examples of the implementation of the pairing pattern.

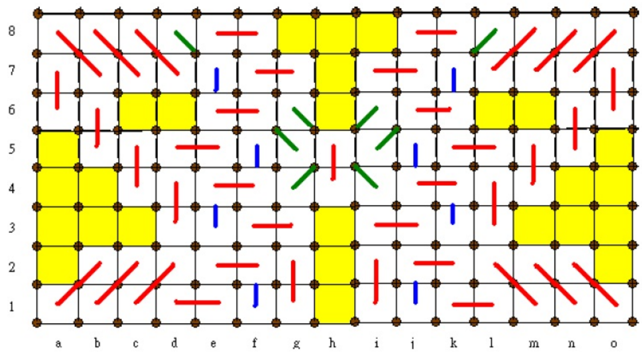

FIGURE 20. The pairing pattern for $(15, 8, 8, 1, 1)$.

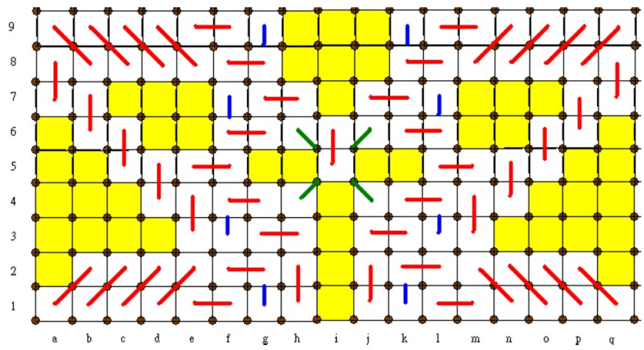

FIGURE 21. The pairing pattern for $(17, 9, 9, 1, 1)$.

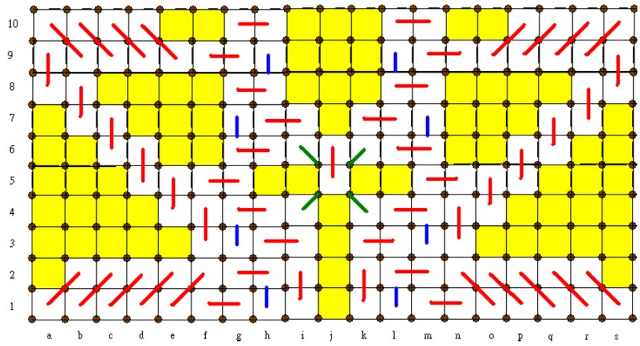

FIGURE 22. The pairing pattern for $(19, 10, 10, 1, 1)$.

In conclusion, $(2k-1, k, k, 1, 1)$ is a forced draw for $k \geq 7$. $\square$

At this point, we have created many pairing patterns for different values of m, n and k. We end this chapter with a simple, yet interesting, theorem.

Theorem 14. *If $n < k$, $(m, n, k, 1, 1)$ is a forced draw for $k \geq 3$.*

Proof. As $n < k$, only horizontal k-in-a-rows can be formed. We pair each square on an odd-numbered column with the square on its right, unless it belongs to the rightmost column. The resulting pairing pattern will look like the one as shown in Figure 23.

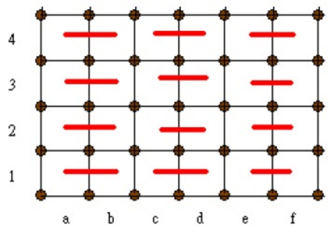

FIGURE 23. Illustration of the pairing pattern described in Theorem 14.

Using this pattern, all horizontal k-in-a-rows contain a set of paired squares for $k \geq 3$. Therefore, if $n < k$, $(m, n, k, 1, 1)$ is a forced draw for $k \geq 3$. $\square$

5. Conclusion

In this project, we have employed two quite different methods, namely, exhaustion and pairing, to solve several (m, n, k, p, q) games. Below is a list of the results we have obtained.

By exhaustion:

- $(m, n, 3, 1, 1)$ is a forced win for B, when $m > 3$ and $n \geq 3$.
- $(5, 5, 4, 2, 2)$ is a forced win for B.
- $(6, 6, 4, 1, 1)$ is a forced win for B.
- $(6, 6, 4, 2, 1)$ is a forced win for W.

By pairing:

- $(k+1, k+1, k, 1, 1)$ is a forced draw for $k \geq 6$.
- $(2k-1, k, k, 1, 1)$ is a forced draw for $k \geq 7$.
- If $n < k$, $(m, n, k, 1, 1)$ is a forced draw for $k \geq 3$.

By exhaustion and pairing, combined:

- $(k, k, k, 1, 1)$ is a forced draw for $k \geq 3$.

Results obtained by exhaustion are often very specific: most of the parameters of the (m, n, k, p, q) game are fixed. Plus, exhaustion can be used to show that a certain game is won, drawn or lost for a certain player.

In contrast, results obtained by pairing can be applied to many board sizes, as long as the relationship between m, n, and k remains unchanged, for example, $n = k$. However, the pairing strategy can only be used to show that a certain game is a forced draw.

While it is hard to use exhaustion to solve more complicated (m, n, k, p, q) games, pairing strategies can prove to be useful in proving certain (m, n, k, p, q) games are drawn, even if the values of m, n, and k are very large.

Other than the main results listed above, we have also found a necessary condition for applying the pairing strategy and, after the discovery of the line

1. e2 . . . f3 2. c2 . . . d2 3. d3 . . . f1,

conjectured that the $(9, 4, 4, 1, 1)$ game is a draw with perfect play by both players.

Since the (m, n, k, p, q) game is not a very popular research topic, we could only find a few websites with relevant results and research progress. Therefore, we have to try very hard to verify the results ourselves.

Our project does not involve complicated mathematical concepts, and the theories behind our research are, in fact, quite simple even for people not familiar with k-in-a-row games. However, we are still delighted to have researched on this interesting topic, and we believe that we have achieved some useful results.

We hope that these results will be able to shed some light on the topic of (m, n, k, p, q) games, and that the results will assist future researchers in their studies on the topic. Even after doing this project, we are still interested in this topic. Thus, we could continue to research on this topic ourselves in the future.

Throughout this project, we have encountered many difficulties, like insufficient time, lack of relevant reference materials, and fruitless exhaustion attempts. Without the help of Ms. Luk, our teacher advisor, we could not have finished this paper, and we would probably obtain no concrete results, other than those obtained by exhaustion. Her advice of exploring methods other than exhaustion led us to the completion of Chapter 4 of this paper, making the project significantly more interesting. Therefore, we would like to thank Ms. Luk for her guidance throughout the year.

Thank you for reading this paper. We sincerely hope that you are satisfied with our work.

6. Suggestions for Further Research

6.1. The centre of the board

Our experience of playing the (m, n, k, p, q) game makes us believe that it is better for both players to 'fight for the centre'. The conclusion seems intuitive, but it may be hard to prove the conjecture mathematically.

6.2. The $(9, 4, 4, 1, 1)$ game

It is still unverified that $(9, 4, 4, 1, 1)$ is a forced win. It may be an interesting, yet exhausting, open problem to work on. Solving the $(9, 4, 4, 1, 1)$ game is a big step closer to solving the family of $(m, n, 4, 1, 1)$ games.

6.3. Pairing strategy for $p \neq 1$ or $q \neq 1$

We suggest that the pairing strategy introduced in Chapter 4 can be modified to cover the case $p \neq 1$ or $q \neq 1$, which, due to time constraints, we were not able to investigate the case in depth.

6.4. Sufficient condition of successful pairing

We have stated a necessary condition of implementing the pairing strategy in Theorem 10. However, we failed to establish a sufficient condition of successful pairing. If criteria of successful pairing are found, all (m, n, k, p, q) games satisfying the criteria will be known as 'draw or better for W'.

REFERENCES

[1] Wikipedia, *Tic-tac-toe*, http://en.wikipedia.org/wiki/Tic-tac-toe

[2] Wikipedia, *M,n,k-game*, http://en.wikipedia.org/wiki/M,n,k-game

[3] I-Chen Wu and Dei-Yen Huang, *A New Family of k-in-a-row Games*, http://www.connect6.org/k-in-a-row.pdf

[4] W.J. Ma. *Generalized Tic-tac-toe*, http://weijima.com/index.php?option=com_conten&view=article&id=11&Itemid=15#more

Appendix A

Analysis of the $(3,3,3,1,1)$ game

1. b2 ... a1

(2. c1 ... a3(F) 3. a2(F) ... c2(F) [4. b1/c3 ... b3] [4. b3 ... b1])

(2. c2 ... a2(F) 3. a3(F) ... c1(F) 4. b1(F) ... b3(F) 5. c3)

(2. c3 ... c1 3. b1(F) ... b3(F) [4. c2/a3 ... a2] [4. a2 ... c2])

2. b1 ... b3(F)

[3. c3 ... c1, *transposing to* 1. b2 ... a1 2. c3 ... c1 3. b1(F) ... b3(F)]

[3. c2 ... a2(F) 4. a3(F) ... c1(F)]

[3. a3 ... c1(F) {4. a2/c3 ... c2} {4. c2 ... a2}]

[3. c1 ... a3(F) 4. c3(F) ... a2#]

[3. a2 ... c2(F) {4. a3/c3 ... c1} {4. c1 ... a3(F)}]

(2. b3, 2. a2, and 2. a3 are symmetrical)

1. a2 ... b2

(2. a3 ... a1(F) 3. c3(F) ... b3(F) 4. b1(F) ... c1)

(2. b3 ... c3 3. a1(F) ... a3(F) 4. c1(F) ... b1(F))

(2. c2 ... c3 3. a1(F) ... a3(F)#)

(2. c3 ... a3 3. c1(F) ... c2(F) [4. b1/b3 ... a1] [4. a1 ... b1])

(2. a1, 2. b1, and 2. c1 are symmetrical)

1. a3 ... b2

(2. b3, *transposing to* 1. b3 ... b2 2. a3)

(2. c2, *transposing to* 1. c2 ... b2 2. a3)

(2. c3 ... b3(F) 3. b1(F) ... a2 4. c2(F) ... c1(F))

(2. c1 ... c2 3. a2(F) ... a1(F) 4. c3(F) ... b3(F))

(2. a1, 2. a2, and 2. c1 are symmetrical)

From the analysis, we can see that the $(3,3,3,1,1)$ game is a forced draw.

Appendix B

Source code of the C++ program for analysis of the $(4, 4, 4, 1, 1)$ game

```
#include<cstdio>
#include<iostream>
#include<cstring>
#include<algorithm>
using namespace std;
//a^b used as 'a to the power of b', not 'a xor b'
//# of states = 3^16 = 43046721

//outcome[state]:
//0 if draw, 1 if B wins, -1 if W wins, -2 if unknown (before
search)/illegal (after search)
//Player 1 (B) aims for maximizing outcome[state]
//Player 2 (W) aims for minimizing outcome[state]
int outcome[44000000];
int pow_3[16];

//board[i][j]: 0 if empty, 1 if occupied by B, 2 if occupied by W
int board[5][5];
bool won(int player){
        for(int i=1;i<=4;i++){
           bool hori_con_4=true;
           for(int j=1;j<= 4;j++){
              if(board[i][j]!=player) hori_con_4=false;
           }
           if(hori_con_4) return true;
        }

        for(int i=1;i<= 4;i++){
           bool vert_con_4=true;
           for(int j=1; j<=4;j++){
              if(board[j][i]!=player) vert_con_4=false;
           }
           if(vert_con_4) return true;
        }

        bool diag1_con_4=true;
        for(int i=1;i<=4;i++){
           if(board[i][i]!=player) diag1_con_4=false;
        }
```

```
        if(diag1_con_4) return true;

        bool diag2_con_4=true;
        for(int i=1;i<=4;i++){
          if(board[i][5-i]!=player) diag2_con_4=false;
        }
        if(diag2_con_4) return true;

        return false;
}
int search(int player, int state,int step){
        if(outcome[state]!=-2){ //the game state has been explored
          return outcome[state];
        }
        if(player ==1) outcome[state]=-1;
        else outcome[state]=1;
        for(int i=1;i<=4;i++){
          for(int j=1;j<=4;j++){
            if(board[i][j] == 0){
              board[i][j] = player;
              //changing board[i][j] from 0 to k = incrementing
state by k* 3^(i*4+j-5)
            if(won(player)){
              if(player == 1) outcome[state]=1;
              else outcome[state]= -1;
            }else if(step==15){
              outcome[state]=0;
            }else{
              if(player==1)
outcome[state]=max(outcome[state],search(2,state+pow_3[i*4+j-5],step+1));
              else
outcome[state]=min(outcome[state],search(1,state+2*pow_3[i*4+j-
5],step+1));
              }
              board[i][j]=0;
              if(player==1 && outcome[state]==1) return 1;
              else if(player==2 && outcome[state]==-1) return -
1;
              else if(step==15) return 0;
            }
          }
      }
```

```
    return outcome[state];
}
int main(){
    for(int i=0;i<43046721;i++) outcome[i]=-2;
    memset(board,0,sizeof(board));
    pow_3[0]=1;
    for(int i=1;i<=15;i++) pow_3[i]=pow_3[i-1]*3;
    printf("% d \ n",search(1,0,0));
    return 0;
}
```

The program gives a single integer, 0, as the output, which means that the the $(4,4,4,1,1)$ game is a forced draw.

Appendix C

Analysis of the $(6,6,4,1,1)$ game

1. d4

(We only consider moves below the a1-f6 diagonal)

(1. . . . a1/b1/e1/f1 2. d3 . . . d2/d5(F) 3. c3#)

(1. . . . c1/d1 2. c4 . . . b4/e4(F) 3. d5#)

(1. . . . b2/f2 2. b4 . . . a4/c4/e4(F) 3. d2#)

(1. . . . c2 2. c4 . . . b4/e4(F) 3. d3#)

(1. . . . d2 2. c4 . . . b4/e4(F) 3. c3#)

(1. . . . e2 2. e4 . . . b4/c4/f4(F) 3. d3#)

(1. . . . c3 2. c4 [2. . . . e4 3. d3#] 2. . . . b4(F) 3. d2 . . . d3(F) 4. f4 . . . e4(F) 5. e3 . . . c1(F) 6.c5#)

(1. . . . d3 2. e3 . . . c5/f2(F) 3. e4#)

(1. . . . e3 2. c3 [2. . . . b2 3. c4#] 2. . . . e5(F) 3. e4 [3. . . . b4 4. c4 . . . f4(F) 5. c2#] [3. . . . f4 4.c4 . . . b4(F) 5. c2#] 3. c4(F) 4. d3#)

(1. . . . f3 2. c4 . . . b4/e4(F) 3. d5#)

(1. …e4 2. c5 [2. …b6/f2 3. d5#] 2. …e3(F) 3. e5# [3. …e2 4. e1(F)#] [3. …e1 4. e2(F)#])

(1. …f4/f5/f6 2 .c4 …b4/e4(F) 3. d3#)

(1. …e5 2. d3 [2. …d2 3. e3#] 2. …d5(F) 3. c5 …b6/e3/f2(F) 4. e4#)

We skip the irrelevant W moves that end in immediate wins for B.

From the analysis, we can see that the $(6, 4, 4, 1, 1)$ game is a forced win.

Appendix D

Analysis of the line *1. e2 …f3 2. c2 …d2 3. d3…f1* of the $(9, 4, 4, 1, 1)$ game

1. e2 …f3 2. c2 …d2 3. d3 …f1

(4. a1/a2/a3/a4/b2/b3/b4/d1/g1/h2/h4/i1/i2/i3 …f4 5. f2 (F) …e3 6. c1 (F) …h3 7. g3 (F) …g2 8. i4 (F) …e4 9. h1 (F) …g4# [9. …d4# if 4. h4])

It is extremely difficult to investigate all possible moves, even with the first three moves given. Therefore, the analysis of a line ends when the line is highly likely to end in a draw by our judgment.

(4. h1/h3/i4 …f4 5. f2(F) …e3 6. c1(F) …d4 [7. b1 …e4(F)#] [7. e4 …b1(F) *draws.*] [7. c3 …c4(F) 8. e4(F) …b1(F) *draws.*] [7. g4 …c4 8. e4(F) b1(F) *draws.*] 7. c4 …c3(F) [8. b1 …e4(F) 9.g4(F) …d1 *draws.*] *From then, W should be able to draw by parrying all immediate threats and occupying b1.)*

(4. g2/g3/g4 …f2 5. f4(F) …e4 *draws.*)

(4. c1 …f4 5. f2(F) …e4 *draws.*)

(4. e1 …f4 5. f2(F) …e3 6. c1(F) …b1 *draws.*)

(4. f2 …e3 *draws.*)

(4. b1 …e4(F) *threatening* 5. …f4 6. f2(F) …d4#. *B has to make a defensive move. After that, W should be able to draw by* 5. …c3.)

(4. c4 …f4 5. f2(F) …c1 6.e3(F) …e1 7. d1(F) …c3 8. b4(F) …d4 *and black can no longer pose meaningful threats.*)

(4. c3 …f4 5. f2(F) …b3 [6. b1 …e4(F) 7. c1 …c4(F) 8. d4 …a1 9. g1 …e3(F) 10. g2 …h2(F) 11. g3 …g4(F) 12. h4(F) …e1(F) *draws.*] [6. e4 …b1(F) 7. e3 …e1 *draws.*])

(4. d4 . . . e4 *draws.*)

(4. e3 . . . e4 *draws.*)

(4.e4 . . . b1(F) 5. e3 . . . e1(F) *draws.*)

(4.f4 . . . e4 *draws.*)

From the analysis, we believe that in the $(9, 4, 4, 1, 1)$ game, the line

1. e2 . . . f3 2. c2 . . . d2 3. d3 . . . f1

leads to a draw.

Reviewer's Comments

In the paper under review, the authors analyse games which generalize the well-known tic-tac-toe, and determine which player has a forced win/draw strategy in various examples. These generalized games, called the (m, n, k, p, q) games, involve two players taking turns to place their pieces on an $m \times n$ square grid, subject to the rule that the first player places q pieces in his/her first turn and each player places p pieces in each subsequent turn. The player who first achieves a k-in-a-row, that is, the game state where his/her k pieces are placed consecutively in a row, column or diagonal, wins the game.

After showing that the second player does not have a forced win if $p \leq q$, the authors analyse the (m, n, k, p, q) games by two methods, namely, exhaustion and pairing strategy. In the first method, all possible sequences of moves are exhausted to determine who has a forced win/draw. The authors employ exhaustion and give definitive results about some examples of (m, n, k, p, q) games where the five parameters are small, and conjecture that the $(9, 4, 4, 1, 1)$ game is a forced draw. On the other hand, the pairing strategy, which is the authors' original contribution in this paper, allows them to determine more efficiently that a large class of (m, n, k, p, q) games are forced draws. The pairing strategy is basically a pair-up of squares in the $m \times n$ square grid such that any k consecutive squares in any row, column and diagonal contain two squares that are paired up. Such a pair-up facilitates a drawing strategy for the second player stated as Algorithm 4.1. In the end of the paper, the authors point out the merits and demerits of the two methods, and draft possible future research directions.

In general, the paper is very well-written with clear explanations of game strategies and algorithms coupled with ample illustrations. In particular, I am impressed by the clarity of the proofs of Theorems 11,12 and 13. Moreover, I appreciate that the authors choose to relegate the tedious analysis of all possible moves of some games to the appendix rather than including it in the body of the paper so as not to obstruct the flow of exposition. The following are some grammatical mistakes I found in the paper and some suggestions for improvement.

1. The reviewer has comments on the wordings, which have been amended in this paper.
2. The authors should define what B and W are, though they define them later in the paper.
3. It is better to say '...which eliminates the possibility...' rather than '...to eliminate the possibility'.
4. It is better to add a note that B and W stand for black and white respectively, as this convention is used in the pictures in later sections.
5. It is better for the authors to point out explicitly how the condition $m > 3$ is used.

6. Change ‘W’ to ‘B’. In fact the authors should write out more moves to explain why B must win.
7. $(9, 4, 4, 1, 1)$ is a forced win, but for whom? Though Theorem 1 implies that it is a forced win for B, still it is better to point that out for the cursory readers.
8. Though the authors point out that the pairing algorithm is not applicable to the game for $k = 5$, it does not imply that the game in this case is not a forced draw. It is better for the authors to indicate explicitly whether they have tried to determine if the game for $k = 5$ is a forced draw or not using other methods, and the difficulties of determining this, if any.
9. Change ‘area’ to ‘square’.