

On the Complexity of Connection Games

Édouard Bonnet

edouard.bonnet@lamsade.dauphine.fr
SZTAKI, *Hungarian Academy of Sciences*

Florian Jamain

florian.jamain@lamsade.dauphine.fr
LAMSADE, *Université Paris-Dauphine*

Abdallah Saffidine

abdallahs@cse.unsw.edu.au
CSE, *The University of New South Wales*

Abstract

In this paper, we study three connection games among the most widely played: HAVANNAH, TWIXT, and SLITHER. We show that determining the outcome of an arbitrary input position is PSPACE-complete in all three cases. Our reductions are based on the popular graph problem GENERALIZED GEOGRAPHY and on HEX itself. We also consider the complexity of generalizations of HEX parameterized by the length of the solution and establish that while SHORT GENERALIZED HEX is W[1]-hard, SHORT HEX is FPT. Finally, we prove that the ultra-weak solution to the empty starting position in HEX cannot be fully adapted to any of these three games.

Keywords: Complexity, Havannah, Twixt, Hex, Slither, PSPACE

1. Introduction

Since its independent inventions in 1942 and 1948 by the poet and mathematician Piet Hein and the economist and mathematician John Nash, the game of HEX has acquired a special spot in the heart of abstract game aficionados. Its purity and depth has lead Jack van Rijswijck to conclude his PhD thesis with the following hyperbole [1].

HEX has a Platonic existence, independent of human thought. If ever we find an extraterrestrial civilization at all, they will know HEX, without any doubt.

HEX not only exerts a fascination on players, but it is the root of the field of *connection games* which is being actively explored by game designers and researchers alike [2].

A connection game is a kind of abstract strategy game in which players try to make a specific type of connection with their pieces [2]. In many connection games, the goal is to connect two opposite sides of a board. In these games, players take turns placing or/and moving pieces until they connect the two sides of the board. HEX, Y, and TWIXT are typical examples of this type of game. However, a connection game can also involve completing a loop (HAVANNAH), connecting all the pieces of a color (LINES OF ACTION), or forbidden patterns (SLITHER).

The focuses of research on abstract strategy games and on connection games in particular include the design and programming of strong artificial players and solvers [3, 4], as well as theoretical considerations on aspects specific to connection games such a virtual connections and inferior cells [1, 5, 6, 7].

Developing an artificial player for a strategy game typically involves adapting standard techniques from the game search literature, in particular the classical Alpha-Beta algorithm [8] or the more recent Monte Carlo Tree Search paradigm [9, 3]. These algorithms which explore an exponentially large game tree are meaningful when optimal polynomial time algorithms are impossible or unlikely. For instance, tree search algorithms would not be used for NIM and SHANNON'S EDGE SWITCHING GAME which can be played optimally and solved in polynomial time [10].

Computational complexity is a theoretical tool used to gain formal intuition on families of problems. It can indicate that a problem is unlikely to be solvable in polynomial time and that exponential algorithms might be the best bet. The complexity class PSPACE comprises those problems that can be solved on a Turing machine using an amount of space polynomial in the size of the input. The prototypical example of a PSPACE-complete problem is the Quantified Boolean Formula problem (QBF) which can be seen as a generalization of SAT allowing for variables to be both existentially and universally quantified. Proving that a game is PSPACE-hard shows that a variety of intricate problems can be encoded via positions of this game. Additionally, it is widely believed in complexity theory that if a problem is PSPACE-hard, then it admits no polynomial time algorithms.

For this reason, studying the computational complexity of games is also a popular research topic. The complexity class of CHESS and GO was determined shortly after the very definition of the appropriate classes and other popular games have been classified since then [11, 12]. More recently, we studied the complexity of trick-taking card games which notably include BRIDGE, SKAT, TAROT, and WHIST [13].

Connection games have received less attention. Besides Even and Tarjan's proof that SHANNON'S VERTEX SWITCHING GAME is PSPACE-complete [14] and Reisch's proof that HEX is PSPACE-complete [15], the only complexity results on connection games that we know of are the PSPACE-completeness of virtual connection detection [16] in HEX, the NP-completeness of dominated cell detection in SHANNON'S VERTEX SWITCHING GAME [17], as well as an unpublished note

showing that a problem related to TWIXT is NP-complete [18].¹

The games that we study in this paper rank among the most notable connection games. The game of TWIXT by Alex Randolph [20] was first commercialized in the 1960s and was short-listed for the prestigious *Spiel des Jahres* award in 1979. Notwithstanding HEX, Christian Freeling’s HAVANNAH is the connection game that has attracted the most interest from researchers and programmers. This may be attributed to Freeling challenging programmers to build an AI able to beat him in at least one game in a ten-game match before 2012. Of course, Christian Freeling could not have foreseen the advent of the Monte Carlo Tree Search algorithm and the prize money of the *Havannah Challenge 2012* was awarded after a man-machine match in October 2012.² SLITHER is a more recent addition to the connection game family as it was designed by Corey Clark in 2010. Nevertheless, the introduction of forbidden patterns mechanics in this game has already proved quite influential in connection game design.³ The two-player game SLITHER that we study in this article should not be confused with the Japanese single-player puzzle SLITHER LINK which has been the object of independent papers [21, 22].

The first two games were the main topic of multiple master’s theses and research articles [23, 18, 24, 25, 26, 27, 4] and all three gave rise to competitive play. High-level online competitive play takes place on www.littlegolem.net. Finally, live competitive play can also be observed between human players at the Mind Sports Olympiads where an international TWIXT championship has taken place every year since 1997, as well as between HAVANNAH computer players at the ICGA Computer Olympiad since 2009.⁴

Our main contributions in this paper are diverse.⁵

1. We establish that HAVANNAH, TWIXT, and SLITHER are PSPACE-complete.
2. We study the parameterized complexity of HEX with the length of a solution as parameter and we show that while SHORT GENERALIZED HEX is W[1]-hard, SHORT HEX is FPT.
3. We provide a formal proof that planar SLITHER does not admit draws, as claimed by its designer, but that it heavily depends on the topology of the board.
4. We demonstrate that the John Nash’s ultra-weak solution approach for HEX can only be adapted to TWIXT, HAVANNAH, and SLITHER to limited extent.

¹For a summary in English of Reisch’s reduction, see Maarup’s thesis [19].

²See the press release at <http://mindsports.nl/index.php/arena/havannah/641>.

³See for example the discussion at <https://www.boardgamegeek.com/thread/1081423/new-connection-games-slither-restriction>.

⁴See www.boardability.com/game.php?id=twixt and www.grappa.univ-lille3.fr/icga/game.php?id=37 for details.

⁵This article is based on two conference papers but improves upon them and extends our previous work significantly [28, 29]. The PSPACE-completeness proof for HAVANNAH has been simplified greatly. The parameterized complexity results are new. The discussion on ultra-weak solutions has been detailed and extended to TWIXT and HAVANNAH.

To the best of our knowledge, our result on SHORT HEX constitute the first tractability result in the field of Connection Games since Bruno and Weinberg’s result on the Shannon Switching Game in 1970 [10]. The other games we focus on each add to or change important mechanics of HEX, the quintessential connection game. Indeed, from a connection perspective, TWIXT is played on a non-planar graph⁶; HAVANNAH adds non-edge related winning conditions; and SLITHER allows moving previously placed pieces. Together, these contributions make for a better understanding of the design decisions in connection games and their theoretical implications.

The paper is organized as follows. After describing the rules of GG and HEX, we prove that TWIXT and SLITHER are PSPACE-complete in Section 3 and that HAVANNAH is PSPACE-complete in Section 4. We then address the parameterized complexity of GENERALIZED HEX and HEX (Section 5). After having extensively considered the computational complexity of solving arbitrary positions of these games, Section 6 moves on to the problem of solving their starting, empty, positions. A discussion and final remarks conclude the paper.

2. Previous Work

A staple in proofs of PSPACE-hardness for two-player games, GENERALIZED GEOGRAPHY (GG) is one of the first two-player games to have been proved PSPACE-complete [30]. This problem has been used to show the intractability of games as different as HEX [15], OTHELLO [31], AMAZONS [32], BRIDGE [13], and many more [12]. Our PSPACE-hardness proof for HAVANNAH is a reduction from GG and the results for TWIXT and SLITHER also rely on GENERALIZED GEOGRAPHY, albeit indirectly since we reduce from HEX.

GG is a game played on a graph. In this section, we recall the rules of GG and some of the assumptions that can be made on the input graph while preserving PSPACE-completeness. We also present the rules of HEX.

2.1. Generalized Geography

In GG, players take turns moving a token from vertex to vertex in a given graph. If the token is on a vertex v , then it can be moved to a vertex v' neighboring v provided v' has not been visited yet. A player wins when it is their opponent’s turn and the opponent has no legal moves. An instance of GG is a graph G and an initial vertex v_0 , and asks whether the first player has a winning strategy in the corresponding game.

We denote by $P(v)$ the set of predecessors of the vertex v in G , and $S(v)$ the set of successors of v . A vertex with in-degree i and out-degree o is called (i, o) -vertex. The degree of a vertex is the sum of the in-degree and the out-degree, and the degree of G is the maximal degree among all vertices of G . If V is the

⁶Indeed, planar graphs have average degree smaller than 6, whereas a large board of TWIXT has average degree close to 8.

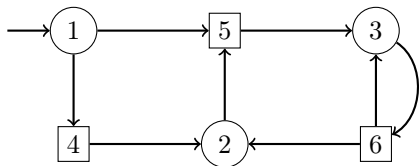


Figure 1: Example of an instance of GG with vertex 1 as initial vertex.

set of vertices of G and V' is a subset of vertices, then $G[V \setminus V']$ is the induced subgraph of G where vertices belonging to V' have been removed.

Lichtenstein and Sipser have proved that the game remained PSPACE-hard even if G was assumed to be bipartite, planar, and of degree at most 3 [33]. We will reduce from such a restriction of GG to show that HAVANNAH is PSPACE-hard. To limit the number of gadgets we need to create, we will also assume a few simplifications detailed below. Naturally, these simplifications do not impact PSPACE-hardness. An example of a simplified instance of GG can be found in Figure 1.

Let (G, v_0) be an instance of GG with G bipartite, planar, and of degree at most 3. We can assume that there is no vertex v with out-degree 0 in G . Indeed, if $v_0 \in P(v)$ then (G, v_0) is trivially winning for Player 1. Else, $(G[V \setminus (\{v\} \cup P(v))], v_0)$ is an equivalent instance, since playing in a predecessor of v is losing.

All edges coming to the initial vertex v_0 can be removed to form an equivalent instance. So, v_0 is a $(0, 1)$ -, a $(0, 2)$ -, or a $(0, 3)$ -vertex. If $S(v_0) = \{v'\}$, then $(G[V \setminus \{v_0\}], v')$ is a strictly smaller instance such that Player 1 is winning in (G, v_0) if and only if Player 1 is losing in $(G[V \setminus \{v_0\}], v')$. If $S(v_0) = \{v', v'', v'''\}$, then Player 1 is winning in (G, v_0) if and only if Player 1 is losing in at least one of the three instances $(G[V \setminus \{v_0\}], v')$, $(G[V \setminus \{v_0\}], v'')$, and $(G[V \setminus \{v_0\}], v''')$. In those three instances v' , v'' , and v''' are not $(0, 3)$ -vertices since they had in-degree at least 1 in G . Therefore, we can also assume that v_0 is $(0, 2)$ -vertex.

Finally, we may assume that there exist a bipartition G_1, G_2 of the graph such that both G_1 and G_2 contain a $(1, 2)$ choice vertex. Indeed, such vertices can always be added to a new component of the graph disconnected from the initial vertex without changing the outcome of the instance.

We call a GG instance with an initial $(0, 2)$ -vertex, at least one $(1, 2)$ -vertex per partition and then only $(1, 1)$ -, $(1, 2)$ -, and $(2, 1)$ -vertices a *simplified* instance.

2.2. Hex

In HEX, two players alternate placing a stone of their color (black or white) in an unoccupied cell of a parallelogram board paved by hexagons. Stones cannot be taken or moved, so the length of any game is bounded by the number of cells. To win, one player must connect together a specified pair of opposite sides of the parallelogram. The other player must connect the other pair of opposite sides. As an example, Figure 2 reproduces a HEX puzzle from the graph theorist Claude Berge, as cited by Hayward and van Rijswijck [34].



Figure 2: A HEX puzzle by Berge. White to play and win. White is trying to connect the bottom left edge to the top right edge and Black is trying to connect the top left edge to the bottom right edge.

In Section 6.1, we will see that a game cannot be drawn. We now formalize what it means to *connect* two (opposite) sides of the board. Two cells are said *neighbors* if they share an edge. In a parallelogram-shaped hexagonal paving, cells have 6 neighbors if they are not on an edge of the parallelogram. On an edge, they may have 2, 3, or 4 neighbors. A black (resp. white) group is a maximal connected component in the graph whose vertices are black (resp. white) stones, and edges represent the *neighbor* relation. Then, connecting two sides mean having a group with at least one stone on each side.

3. Complexity of Twixt and Slither

3.1. Twixt

Alex Randolph's TWIXT is one of the most popular connection games. It was invented around 1960 and was marketed as soon as in 1962 [23]. In his book devoted to connection games, Cameron Browne describes TWIXT as one of the most popular and widely marketed of all connection games [2]. We now briefly describe the rules of TWIXT and refer to Moesker's master's thesis for an introduction and a mathematical approach to the strategy, and the description of a possible implementation [24].

TWIXT is a 2-player connection game played on a GO-like board. At their turn, player White and Black place a pawn of their color in an unoccupied place. Just as in HAVANNAH and HEX, pawns cannot be taken, moved, nor removed. When 2 pawns of the same color are spaced by a knight's move, they are linked by an edge of their color, unless this edge would cross another edge. At each turn, a player can remove some of their edges to allow for new links. The goal for player White (resp. Black) is to link top and bottom (resp. left and right) sides of the board. Note that sometimes, a player could have to choose between two possible edges that intersect each other. The *pencil and paper* version TWIXTPP where the edges of a same color are allowed to cross is also famous and played online.

As an illustration of the game rules, we reproduce here one of the original TWIXT puzzles invented by Alex Randolph. A complete list of puzzles by

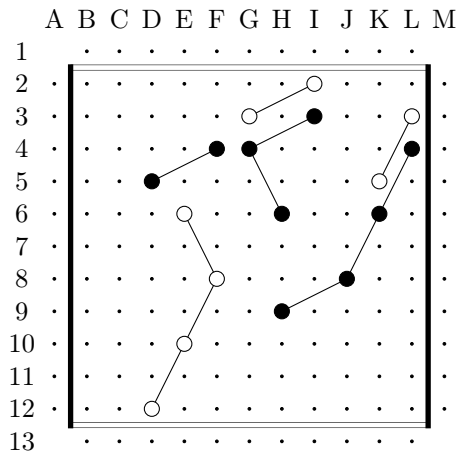


Figure 3: TWIXT puzzle 18 by Alex Randolph. White to play and win.

Randolph supplemented by new puzzles by Alan Hensel can be found on <http://www.ibiblio.org/twixtpuzzles/>. Figure 3 displays puzzle 18 in which White is to play and win. The first natural observation, in this puzzle, is that White has three groups of stones, two of which are connected to the top side, and one is connected to the bottom side. A tentative approach could be to play F5, bringing the top left group closer to the bottom group. Unfortunately that attempt fails when Black answers F6, shutting the top group completely, and the F5 move is wasted. The solution to the puzzle is, instead, for White to start with move G7, which is connected to the bottom group. White threatens both to connect the bottom group to the top left one via F5 and to connect the bottom group to the top right group via I6. There is no way for Black to prevent both threats at once and White wins the game.

As the length of a game of TWIXT is polynomially bounded, exploring the whole tree can be done with polynomial space using a minimax algorithm. Therefore TWIXT is in PSPACE.

Mazzoni and Watkins have shown that 3-SAT could be reduced to single-player TWIXT, thus showing NP-completeness of the variant [18]. While it might be possible to try and adapt their work and obtain a reduction from 3-QBF to standard two-player TWIXT, we propose a simpler approach based on HEX. The PSPACE-completeness of HEX has already been used to show the PSPACE-completeness of AMAZONS, a well-known territory game [32].

We now present how we construct from an instance G of HEX an instance $\phi(G)$ of TWIXT. We can represent a cell of HEX by the 9×9 TWIXT gadgets displayed in Figure 4. Let n be the size of a side of G , Figure 5 shows how a TWIXT board can be paved by n^2 TWIXT cell gadgets to create a HEX board.

It is not hard to see from Figure 4a that in each gadget of Figure 5, move w (resp. b) is dominating for White (resp. Black). That is, playing w is as good for White as any other move of the gadget. We can also see that the moves that are

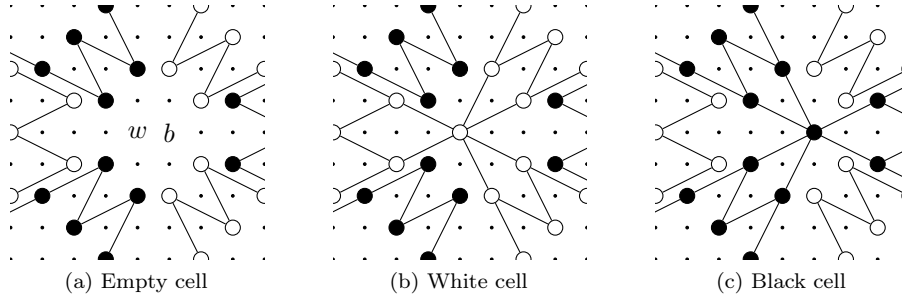


Figure 4: Basic gadgets representing HEX cells in TWIXT.

not part of any gadget in Figure 5 are dominated for both players. As a result, if player Black (resp. White) has a winning strategy in G , then player Black has a winning strategy in $\phi(G)$. Thus, G is won by Black if and only if $\phi(G)$ is won by Black. Therefore determining the winner in TWIXT is at least as hard as in HEX, leading to the desired result.

Theorem 1. *TWIXT is PSPACE-complete.*

Observe that the proposed reduction holds both for the classic version of TWIXT as well as for the *pencil and paper* version TWIXTPP. Indeed, the reduction does not require the losing player to remove any edge, so it also proves that TWIXTPP is PSPACE-hard.

3.2. Slither

Invented in 2010 by Corey Clark, SLITHER is relatively new connection game with an increasing popularity among online board game players. Unlike HEX and HAVANNAH which are played on a hexagonally-paved board, SLITHER is played on a grid and each player is trying to connect a pair of opposite edges corresponding to their color by constructing connected groups of stones. Whereas moves in most other connection games only involve putting down a new element on the board, moves in SLITHER also allow relocating previously played stones. Another important difference between usual connections games and SLITHER is that some stone configurations are forbidden in the latter. Namely, a player is not allowed to play a stone diagonally adjacent to a pre-existing stone of their color unless one of their already placed stones would be mutually adjacent.

3.2.1. Rules

SLITHER is a two-player game starting on an empty n by n grid (or board). Let us call the players Black (or B) and White (or W). Black and White alternate *moves*. Before stating what a move consists of, and what the winning conditions are, we introduce some useful definitions.

As the game proceeds, *squares* of the board can be empty, or contain a black stone, or contain a white stone. We refer to black (resp. white) stones as the

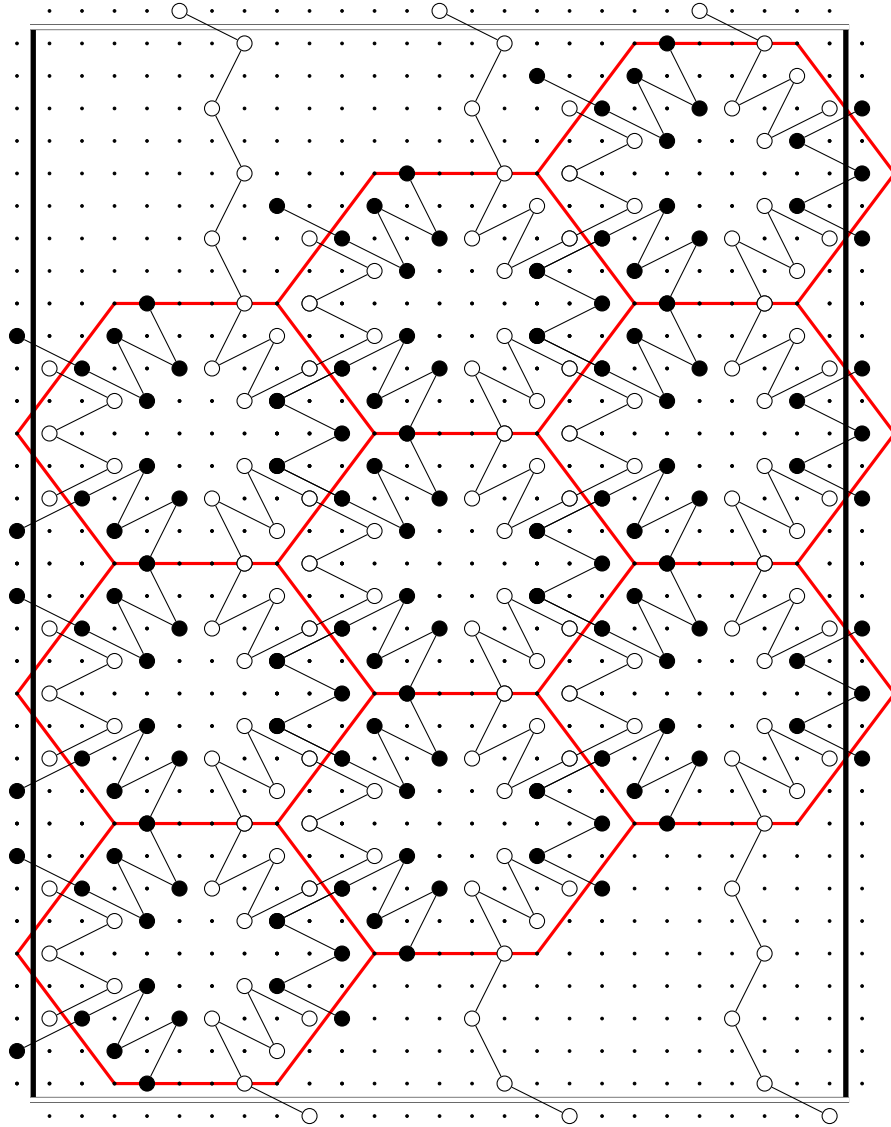


Figure 5: Empty 3×3 HEX board reduced to a TWIXT board.

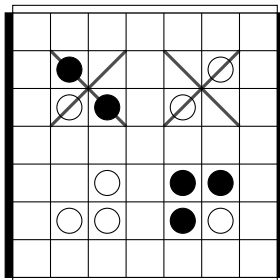


Figure 6: Some examples of allowed and forbidden configurations. Forbidden configurations are crossed.

stones of player B (resp. W). We say that two squares of the board are *adjacent* if they are in the same row and adjacent columns, or in the same column and adjacent rows. They are *king-adjacent* if a chess king can move from one square to the other, and *diagonally-adjacent* if they are king-adjacent but not adjacent. Two stones are *adjacent* (resp. *king-adjacent*, *diagonally-adjacent*) if they are in adjacent (resp. king-adjacent, diagonally-adjacent) squares. For $P \in \{W, B\}$, let G_P be the graph whose vertices are the stones of player P placed in the board, and the edges encode the *adjacent* relation. That is, two vertices are linked by an edge if and only if they represent adjacent stones. Like in the game of GO, a *group* for player P is a maximal connected component in G_P .

A *move* for player P consists of an optional relocation of an existing stone of P on a king-adjacent empty grid square, followed by a mandatory placement a stone of P into an empty grid square (Figure 7). For a move to be legal, the resulting position may not have two diagonally-adjacent stones of P that do not also have an orthogonally-adjacent stone in common (Figure 6). In what follows, we refer to this restrictive rule as the *diagonal rule*.

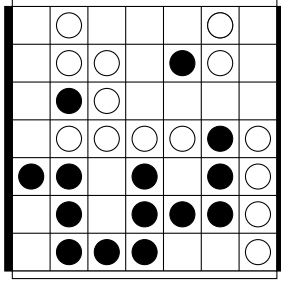
Black wins if they form a group with at least one stone in the first and in the last column. White wins if they form a group with at least one stone in the first and in the last row. Informally, Black wants to connect left-right and White wants to connect top-bottom (Figure 7b).

As in most connection games, a *swap* rule is usually implemented. That is, after the first move, the second player can decide either to play themselves a move and the game goes on normally, or to become first player with that very same move.

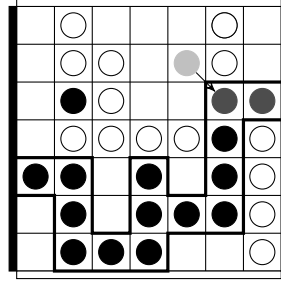
3.2.2. Computational complexity

Here, we show that deciding if one player has a winning strategy from a given position is intractable.

We present a reduction from HEX which is PSPACE-complete [15]. A hexagonal cell of HEX is encoded by the gadgets depicted in Figure 8. More precisely, an empty cell (resp. a cell containing a black stone, resp. containing a white stone) is transformed into the portion of position of Figure 8a (resp. Figure 8b, resp. Figure 8c).

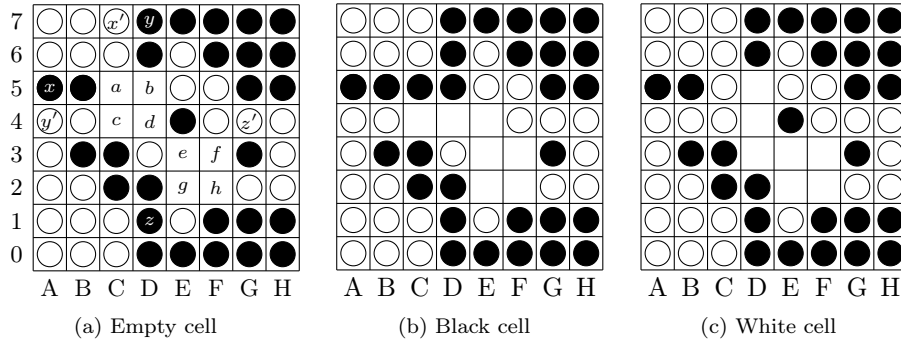


(a) Black to play and win in one move.



(b) The winning move for Black.

Figure 7: Illustration of a move and a winning group on a 7×7 SLITHER board.



(a) Empty cell

(b) Black cell

(c) White cell

Figure 8: Basic gadgets representing HEX cells in SLITHER.

Observation 1. *When a player places a stone in an empty cell gadget (that is, on a square marked with letter a to h), they create a configuration which is forbidden by the diagonal rule. Thus, they should also move one of their stone in the same cell gadget.*

Lemma 1. *In a black cell (Figure 8b), White cannot prevent Black from having a group containing stones in x , y , and z .*

Proof. Black stones on x and y are already in the same group. Because of the diagonal rule, White cannot place a stone in cell C and move a stone in another cell gadget. By Observation 1 and the previous remark, if White moves a stone in cell C , but decides to place a stone in another cell gadget, they can only do so in a white cell, which turns out to be useless. Thus, White might as well place a stone *and* move in cell C . After their move, White should occupy square c ; otherwise, Black places a stone on c and thereby connects their group containing z to their group containing x and y .

There are three ways for W to occupy square c : (1) move stone on B4 to c , or (2) move stone on D3 to c , or (3) place a new stone on c . The first option cannot be extended into a legal move. Indeed, the diagonal rule would impose

that a stone is placed on d , to connect the two diagonally-adjacent white stones on c and $D3$. But then white stones on d and $E5$ would form a forbidden configuration. In the second option, White cannot place a stone on $D3$ nor on d , because of the diagonal rule. And Black's next move would consist of moving the stone on $D2$ to $D3$, and placing a stone on d , which connects z to x and y . Finally, in the third option, White is forced to move their stone in $D3$ to a square other than d . And Black connects in the same manner. \square

Since the cell gadget is symmetric, the following holds similarly.

Lemma 2. *In a white cell (Figure 8c), Black cannot prevent White from having a group containing stones on x' , y' , and z' .*

The cell gadgets are glued together and attached to the edges of the board as described in Figure 9. The empty squares which do not correspond to one of the eight squares designated by letter a to h in Figure 8a can be filled with black stones. For convenience, we do not represent those stones.

The following observation is outlined by Figure 9.

Observation 2. *When playing in a empty cell gadget, Black cannot do more than connecting x , y , and z , and White cannot do more than connecting x' , y' , and z' .*

From the empty cell gadget, Black can move stone $E4$ to b and place a stone on a , resulting in the black cell configuration. By Lemma 1 and Observation 2, it is the optimal play within this cell. Similarly, the optimal play for White in a given empty cell, is to move stone $D3$ to c and place a stone on a , yielding the white cell. Thus, having chosen the cell gadget where to play, the optimal move is to connect six paths going from this cell to the six adjacent cells in a hexagonal paving (Figure 9). Hence, the built SLITHER position simulates a game of HEX, and so, SLITHER is as hard as HEX.

Theorem 2. *It is PSPACE-complete to decide which player has a winning strategy from a given SLITHER position.*

Proof. The membership of this problem to PSPACE boils down to noticing that the length of a game is bounded from above by the number of empty squares. Indeed, at each move, one stone is added to the board. Thus, a minimax depth-first search uses a polynomial amount of space.

The gadgets in Figure 8 and their assembly as in Figure 9 provide a reduction from HEX, a PSPACE-hard problem. \square

4. Complexity of Havannah

HAVANNAH is a 2-player connection game played on a hexagonal board paved by hexagons. White and Black place a stone of their color in turn in an unoccupied cell. Stones cannot be taken, moved nor removed. Two cells are neighbors if they share an edge. A group is a connected component of stones of

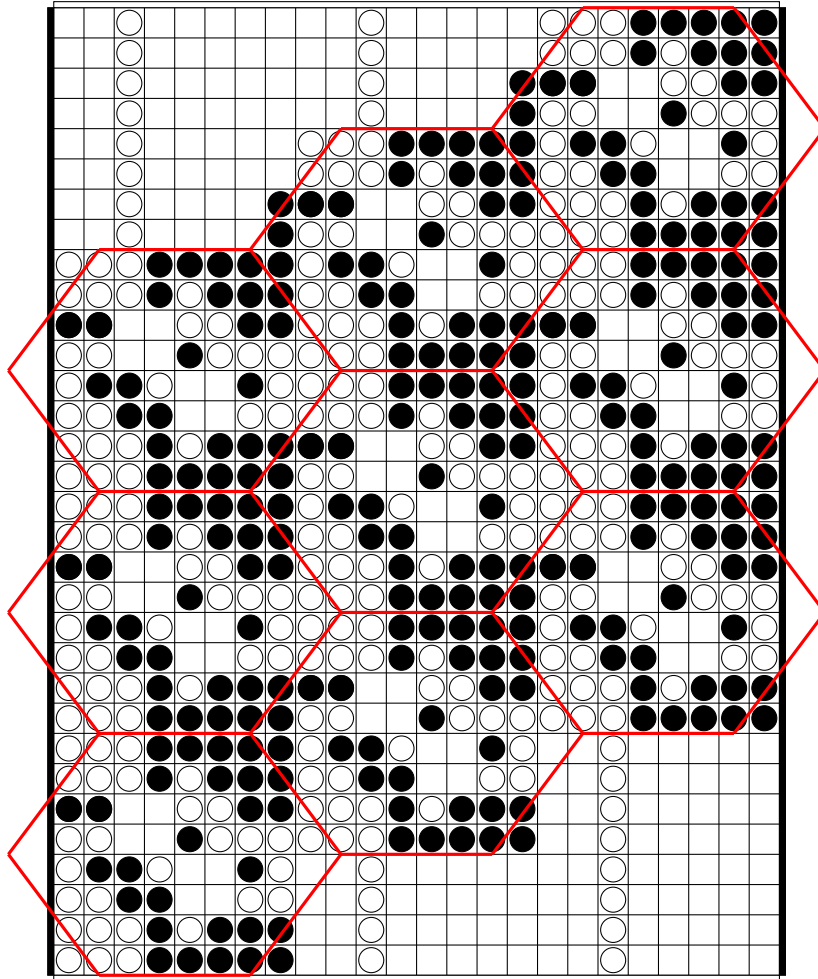


Figure 9: Empty 3×3 HEX board reduced to a SLITHER position.

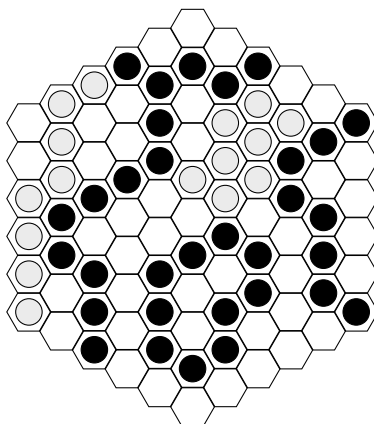


Figure 10: HAVANNAH winning conditions on a size 6 board. Each black group represents a winning pattern: from left to right, *fork*, *ring*, and *bridge*. White groups are not winning, the left group does not constitute a fork because the corner is not part of any edge and so the group only connects two edges; the right group does not constitute a ring because it would need to enclose a non-empty surface.

the same color via the neighbor relation. A player wins if they realize one of the three following different structures: a circular group, called *ring*, with at least one cell, possibly empty, inside; a group linking two corners of the board, called *bridge*; or a group linking three edges of the board, called *fork*. With respect to the *fork* winning condition, corner cells are not considered part of any edge. Figure 10 presents a board in which all three winning conditions are met by different group of stones.

4.1. Preliminary results

As the length of a game of HAVANNAH is polynomially bounded, exploring the whole game tree with Depth-First Search can be done in polynomial space, so HAVANNAH is in PSPACE.

In our reduction, the HAVANNAH board is large enough that the gadgets are far from the edges and the corners. Additionally, the gadgets feature ring threats that are short enough that the bridges and forks winning conditions do not have any influence. Before starting the reduction, we define threats and make two observations that will prove useful in the course of the reduction.

We call *simple threat* a move which threatens to realize a ring on the next move. There are only two kinds of answers to a simple threat: either win on the spot or defend by placing a stone in the cell creating this very threat.

Lemma 3. *If a player is not threatened, playing a simple threat forces the opponent to answer on the cell of the threat.*

Proof. Placing a stone on the cell of the threat wins the game against any other move of the opponent. \square

A *double threat* is a move which threatens to realize a ring on the next move on at least two different cells. We will use *threat* as a generic term to encompass both simple and double threats. To be more concise, we will denote by $W : a_1, a_2; a_3, a_4; \dots; a_{2n-1}, a_{2n}$ a sequence of moves starting with White's move a_1 , Black's answer a_2 , and so on. Similarly, $B : a_1, a_2; a_3, a_4; \dots; a_{2n-1}, a_{2n}$ denotes the corresponding sequence of moves initiated by Black.

Lemma 4. *If a player is not threatened, playing a double threat is winning.*

Proof. The player is not threatened, so their opponent cannot win next turn. Let u and v be two cells of the double threat. If the opponent plays in u , the player wins by playing in v . If the opponent plays anywhere else, the player wins by playing in u . \square

A second-order threat, or *2-threat*, is a move which threatens to realize a double threat on the next move. That is, after a 2-threat is played by a player P , there is a set of cells $\{x, a, b\}$ such that both P playing $\{x, a\}$ as well as P playing $\{x, b\}$ would complete a ring. We call the three cells involved in a 2-threat *carrier* of the threat, and in particular the cell x is called the *exit* of the threat. For example, in Figure 11a, move n is a 2-threat for Black with carrier x, a , and b . Indeed, if Black follows up with x , then Black obtains a ring by playing a or by playing b .

Lemma 5. *If a player makes a 2-threat, then any opponent move that does not belong to the following list is losing. 1) Immediately winning move 2) Simple threat 3) Move inside the carrier of the threat.*

In the following subsections we propose gadgets that encode the different parts of a *simplified* instance of GG, as defined in Section 2.1. These gadgets have starting points and ending points. The gadgets are assembled so that the ending point of a gadget coincides with the starting point of the next one. The resulting instance of HAVANNAH is such that both players must enter in the gadgets by a starting point and leave it by an ending point otherwise they lose. Wires and curves will enable us to encode the edges of the input graph. While figures and lemmas are mostly presented from White's point of view, all the gadgets and lemmas work exactly the same way with colors reversed.

4.2. Vertex gadgets

Recall from Section 2.1 that simplified GG instances only feature four types of vertices: (1, 1)-, (2, 1)-, (0, 2)-, and (1, 2)-vertices. Figure 11 depicts how these four types of vertices are mapped into HAVANNAH gadgets. The diagrams illustrate the White version of the gadgets, corresponding to vertices belonging to the first player in GG. Each gadget can naturally be rotated by multiples of 60° and the colors can be swapped to obtain the Black gadgets.

In Figure 11d and 11c, after Black plays n , White can elect to play c_1 or c_2 . In the former case, Black is forced to reply c_2 which forces White to exit the gadget via x_1 . In the latter case, Black replies c_1 and then White exits the gadget via x_2 .

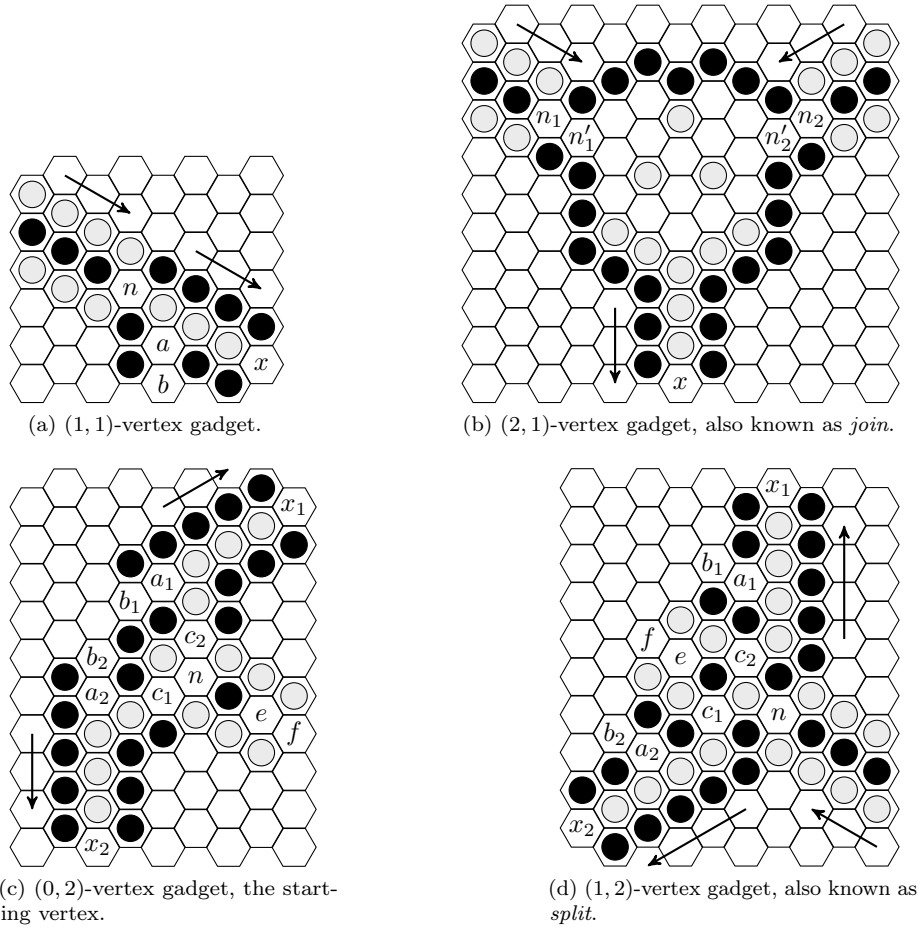


Figure 11: White gadgets before being used. Black enters the gadget by playing n (either n_1 or n_2 in Figure 11b). In each case, optimal play leads White to exit the gadget via x (either x_1 or x_2 in the choice gadgets Figure 11d and 11c). In both choice gadgets, exactly one of the sequences $c_1; c_2$ or $c_2; c_1$ is played between the entry and the exit of the gadget, up to White's decision.

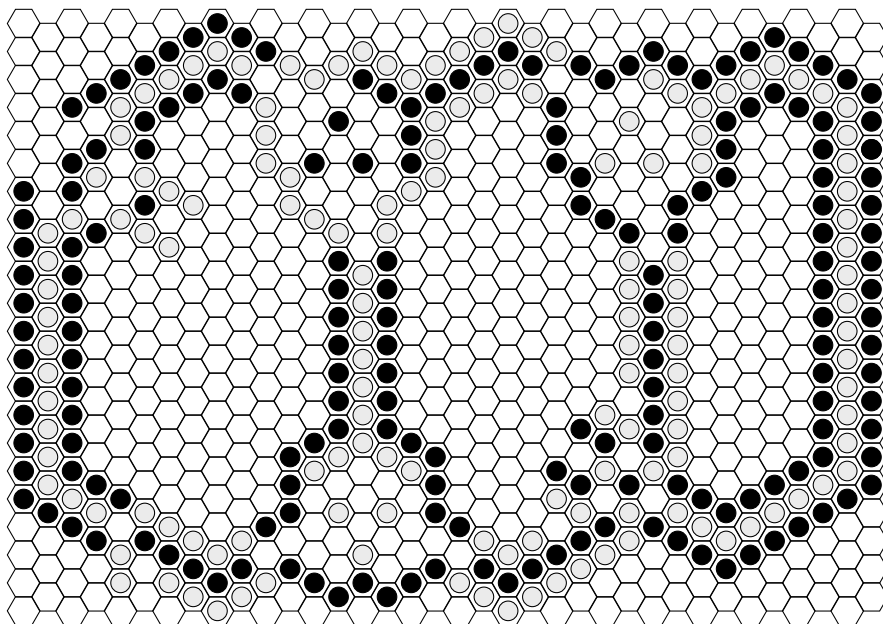


Figure 12: HAVANNAH gadgets representing the GG instance from Figure 1. Black to play first.

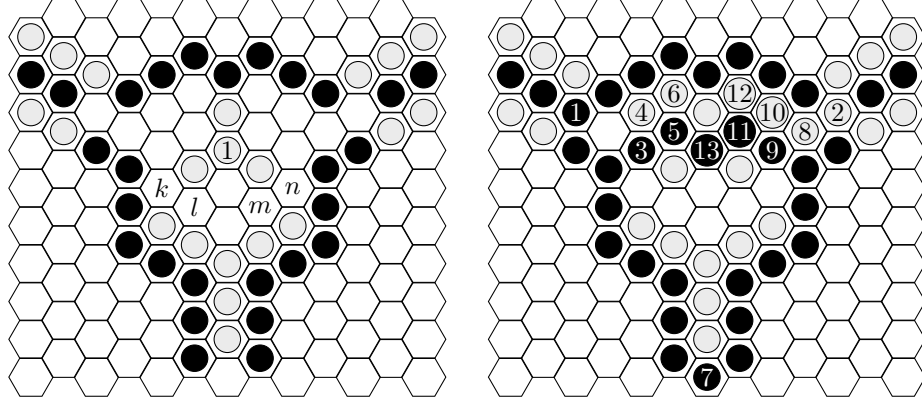
Let (G, v_0) be a simplified instance of GG. G being bipartite, we denote by V_1 the side of the partition containing v_0 , and V_2 the other side. Player 1 moves the token from vertices of V_1 to vertices of V_2 and player 2 moves the token from V_2 to V_1 . We denote by ϕ the reduction from GG to HAVANNAH. Each of Player 1's vertex is encoded by an instance of the corresponding gadget of Figure 11, and each of Player 2's vertex is encoded similarly with colors reversed. Wires and curves are used to connect the gadgets. As an example, we provide the reduction from the GG instance from Figure 1 in Figure 12.

4.3. Properties

We now establish a few properties that will help us prove that Player 1 wins the GG instance if and only if White wins the corresponding HAVANNAH position. The main idea of the reduction is that optimal play in the HAVANNAH position consists of a sequence of White defending a 2-threat by making a 2-threat which Black defends by making their own 2-threat and so on and so forth. Under optimal play, the current 2-threat is in the gadget that corresponds to the token vertex in the GG instance.

Lemma 6. *If White plays the 2-threat in one of their (2,1)-vertex gadgets (Figure 13a) and Black has no winning sequence of simple threats elsewhere on the board, then White can force a win.*

Proof. With move 1 in Figure 13a, Player White makes four distinct 2-threats with respective carriers: $\{k, l, m\}$, $\{k, l, n\}$, $\{k, m, n\}$, and $\{l, m, n\}$. The inter-



(a) White's 2-threats network in the $(2, 1)$ gadget. Black cannot defend against this threat locally and would need to find a winning sequence of simple threats elsewhere on the board.

(b) Main winning line for Black when White does not exit the gadget with x in Figure 11b, assuming Black had entered from n_1 in Figure 11b. Move 2 and 8 are interchangeable for White and are both losing.

Figure 13: Possible developments on a White $(2, 1)$ -vertex gadget.

section of these carriers is empty, so no single Black move is inside the carrier of all threats. By Lemma 5, the only non-losing Black moves would be those satisfying criterion 1) or 2). Since we assumed that Black had no winning sequence of simple threats, no move satisfies criterion 1) nor criterion 2). Therefore, White can force a win. \square

This 2-threat network remains valid even if the $(2, 1)$ gadget has been entered and exited before move 1 is played. Note also that there is no preemptive way for Black to “break” the network: If Black was to play in 1, then White can play the cell just beneath 1 and create a winning 2-threat network. Similarly, move k to n from Black before 1 is played would let White create another 2-threat network in the same gadget.

The dual result is true for Black's $(2, 1)$ gadget.

Corollary 1. *A move by a player P that does maintain the existence of a sequence of winning threats for P is a losing move.*

Proof. Because we are reducing from a *simplified* GG instance, we know that each player has at least one $(2, 1)$ -vertex. Suppose that a player makes a move that leaves the board without a potential sequence of winning threat for them, then the opponent can make the winning 2-threat network in one of their $(2, 1)$ -vertices and force a win by Lemma 6. \square

Corollary 1 ensures that not only should the players defend against the opponent's 2-threats, but that their defense should create 2-threats for the opponent to defend.

Before showing that all the gadgets behave as expected, let us notice that the exchange 3; 4; 5; 6 in Figure 13b can be played in any $(2, 1)$ vertex previously entered from n_1 , using Figure 11b's notation. This exchange (or the symmetrical one 9; 10; 11; 12 when the gadget has been entered from n_2) does not harm Black and the White moves are forced, so we will assume that the exchange always takes place as soon as available, even if White correctly exits the gadget in x .

Lemma 7. *If Black enters a White $(2, 1)$ -vertex gadget via n_1 or n_2 (Figure 11b), and White does not have a threat, then White is forced to play in the exit point x .*

Proof. When Black plays in n_1 , it creates a 2-threat with carrier $\{x, n_2, n'_2\}$. Suppose that White plays in n_2 , then Black has a winning sequence of simple threats as displayed in Figure 13b. If instead White plays in n'_2 , then the same winning sequence is available for Black, simply swapping White move 2 for White move 8 in Figure 13b. Invoking Lemma 5, we know that White needs to play in the carrier, so we can conclude that White needs to play in x .

The proof when Black plays in n_2 is symmetrical. \square

Lemma 8. *If Black enters a White $(1, 1)$ -vertex gadget in n (Figure 11a), and White does not have a threat, then White is forced to play in the exit point x .*

Proof. When Black plays in n_1 , it creates a 2-threat with carrier $\{x, a, b\}$. White playing in a or in b does not create any new threat for Black, so by Corollary 1 these moves are losing. Invoking Lemma 5, we know that White needs to play in the carrier, so we can conclude that White needs to play in x . \square

Lemma 9. *If Black enters a White $(1, 2)$ -vertex gadget in n (Figure 11d), and neither player has a threat available elsewhere on the board, then optimal play dictates that at most two sequences of moves are possible, $c_1; c_2; x_1$ and $c_2; c_1; x_2$, up to White's choice.*

Proof. Let us first show that after Black n and White c_1 , the sequence $c_2; x_1$ is forced. White's c_1 creates a 2-threat with carrier $\{c_2, e, f\}$. Black playing in e or in f does not create any new threat for White, so by Corollary 1 these moves are losing. Invoking Lemma 5, we know that Black needs to play in the carrier, so we can conclude that Black needs to play in c_2 . Black's c_2 creates a 2-threat with carrier $\{x_1, a_1, b_1\}$. White's playing a_1 or b_1 does not create any new threat for black, so by the same reasoning, we obtain that White is forced to play x_1 .

The same style of argumentation shows that if White starts with c_2 , then Black needs to reply c_1 and x_2 is forced for White.

Now suppose that after Black n , White plays a non-threat elsewhere on the board. Black plays in c_1 , forcing White c_2 . Black then plays x_2 creating a double threat in a_2 and b_2 and White has lost. \square

Lemma 10. *When Black initiates in a White $(0, 2)$ -vertex gadget in n (Figure 11c), if neither player has a threat available elsewhere on the board, then optimal play dictates that at most two sequences of moves are possible, $c_1; c_2; x_1$ and $c_2; c_1; x_2$, up to White's choice.*

On the other hand, a player re-entering a $(2, 1)$ gadget in HAVANNAH loses, so it cannot be part of a winning strategy for that player.

As a result, we can map any HAVANNAH winning strategy for White to a GG winning strategy for Player 1 and any HAVANNAH winning strategy for Black to a GG winning strategy for Player 2. Lemmas 7 through 10 together with the “no threat elsewhere invariant” guarantee that exiting the most recently entered gadget is the optimal HAVANNAH play for both players. Since the number of vertices is finite, one player will be forced to re-enter a $(2, 1)$ gadget under optimal play. This means that the theoretical outcome of the translated instance is not a draw, so one player has indeed a winning strategy in HAVANNAH and finding it solves GG. \square

5. Parameterized Complexity of Short Generalized Hex

Generalized Hex. SHANNON’S VERTEX SWITCHING GAME is more commonly called GENERALIZED HEX since it generalizes HEX played on a hexagonal tiling to any (potentially non planar) graphs. A GENERALIZED HEX-instance is a graph G with two specified vertices s and t , each containing a white pebble. White (Player 1) and Black (Player 2) alternate in playing pebbles of their color in unoccupied vertices of the graph. White wins if he manages to create an (s, t) -path where each vertex contains a white pebble. Black wins if he prevents White from doing so. When a black pebble is played in a vertex, one can equivalently imagine that this vertex has been removed from the graph. Any edge having a white pebble at its two endpoints can be equivalently contracted to a single vertex containing a white pebble. That way, White wins if s and t ends up within the same vertex. In all generality, one can assume that the initial instance already contains white pebbles in some other vertices than s and t .

Parameterized complexity. In a nutshell, parameterized complexity aims at solving hard problems (be it NP-hard, PSPACE-hard, etc.) in time $f(k)n^{O(1)}$, called FPT time, (for fixed-parameter tractable), where n is the size of the instance, f is a computable function, and k is a *parameter* of the problem. The parameter can take various forms: *size of the solution* for optimization problems, *treewidth* or *maximum degree* for graph problems, *size of the alphabet* for word problems, to name a few. Assuming the problem we are trying to solve is NP-hard, function f has to be superpolynomial, unless $P = NP$. When such an FPT algorithm exists, f is usually exponential. However, if our parameter k is *small* compared to the size of the instance n , we gain that the exponential blow-up is limited to the small value k . Some problems, like finding a clique of size k in a graph having n vertices, does not admit an algorithm running in time $f(k)n^{O(1)}$ for any computable function f , unless 3-SAT can be solved in subexponential-time which is believed unlikely. Phrased with the terminology of parameterized complexity, CLIQUE is unlikely to be FPT parameterized by the size of the solution. There is a whole hierarchy of problems highly suspected of not being FPT: $W[1] \subseteq W[2] \subseteq \dots \subseteq W[SAT] \subseteq W[P] \subseteq AW[\star]$. For our purpose here, we do not need to define formally all those classes. Instead, we only

say that CLIQUE is $W[1]$ -complete for FPT reductions, where an FPT reduction may blow-up the size of the instance n only polynomially but the new parameter can be any computable function of the old parameter. As an easy exercise, one can check that if there is an FPT reduction from a problem A to a problem B and B admits an FPT algorithm, then so does A . The interested reader is also referred to two recent and extensive textbooks on parameterized complexity [35, 36].

Short games. In their introductory book to parameterized complexity [37], Downey and Fellows devotes a small section to *k-move games*. The *k-move* (or *short*) variant of a two-player game consists of deciding if Player 1 can win in at most k moves. More formally, if one identifies a strategy to a decision tree, one needs to determine if Player 1 has a winning strategy of depth at most k . The problem is parameterized by k . The PhD thesis of Allan Scott [38] is dedicated to short games. The problem SHORT CHESS was shown $AW[*]$ -complete [39]. Pursuit-evasion games (also known as cops and robber games) have been studied within the framework of short games [40] as well as infinite two-player games on graphs [41].

The parameterized complexity of SHORT GENERALIZED HEX was first asked in the book of Downey and Fellows [37] and is still open [42, 36]. We answer partially this question by showing that SHORT GENERALIZED HEX is unlikely to be FPT. More precisely, we prove that SHORT GENERALIZED HEX is $W[1]$ -hard. Nevertheless, this problem could be higher on the intractability hierarchy. It is conjectured $AW[*]$ -complete and Allan proved that it belongs to $AW[*]$. From a purely practical perspective, our partial answer is already satisfactory in asserting that we should not expect to find an FPT algorithm solving SHORT GENERALIZED HEX.

Multicolored clique. In MULTICOLORED CLIQUE, one is given a graph where the vertices are partitioned into k sets called *color classes* and the goal is to find a clique of size k that intersects each color class exactly once. The MULTICOLORED CLIQUE is also $W[1]$ -complete and it is often more convenient to design FPT reductions from this variant than from CLIQUE.

Theorem 4. SHORT GENERALIZED HEX is $W[1]$ -hard.

Proof. From an instance $G = (V = V_1 \uplus V_2 \uplus \dots \uplus V_k, E)$ of the $W[1]$ -hard MULTICOLORED CLIQUE, we build an equivalent instance (H, s, t) of SHORT GENERALIZED HEX in the following way. For each vertex $v \in V$, we add four vertices v^0, v^1, v^2, v^3 to H and a path $sv^0v^1v^2v^3t$. For every $i \in [k]$, we add a vertex t_i , and we link t_i to t and to all the vertices v^2 such that $v \in V_i$. Similarly, we add a vertex s_i that we link to s and to all the vertices v^1 such that $v \in V_i$. That finishes the construction of the selector gadgets (see Figure 15).

Then, we add two copies of an almost complete binary tree (i.e., all levels except the last are completely filled) with $\binom{k}{2}$ leaves, and link both of their root to a new vertex r , itself linked to t . For each leaf of the second to last level, we replace the edge incident to it by two parallel paths of length 2. This has

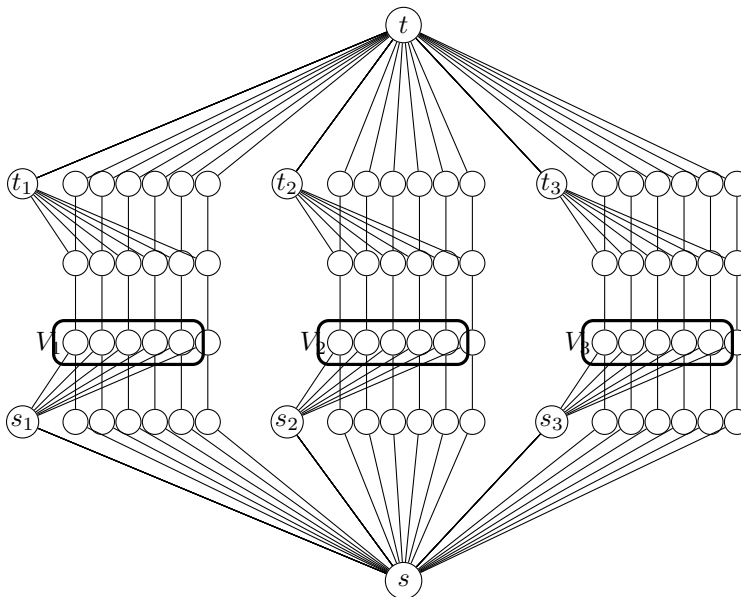


Figure 15: The selector gadgets with $k = 3$.

the same effect of *weighting* the edge by 2 (see Figure 16), so that each leaf is at the *same distance* of the root. We make the slight abuse of still calling the whole structure a tree. Within the b -th copy of the two binary trees, each leaf represents a distinct set $E_{ij} := E(V_i, V_j)$ with $i \neq j$, and is denoted by $l_{i,j}(b)$.

For each edge $e = uv \in E$ and $a, b \in [2]$, we add three vertices $w_e(a, b)$, $w_{e,u}(a, b)$ and $w_{e,v}(a, b)$. We link $w_e(a, b)$ to $w_{e,u}(a, b)$ and $w_{e,v}(a, b)$. We also link $w_{e,u}(a, b)$ to u^1 and $w_{e,v}(a, b)$ to v^1 . Finally, for any $b \in [2]$, we link $l_{i,j}(b)$ to all the vertices $w_e(a, b)$ such that $e \in E_{ij}$ and $a \in [2]$ (see Figure 16). For each $b \in [2]$, we denote by T_b the b -th binary tree together with all the vertices $w_e(a, b)$ and $w_u(a, b)$, for $a \in [2]$, $e \in E$ and $u \in V$. We observe that T_1 and T_2 are disjoint trees. The tree induced by T_1 , T_2 , and r is denoted by T , and is seen as rooted in r .

This ends the construction of H which has $O(|V(G)| + |E(G)| + k^2)$ vertices. We ask if there is a winning strategy for White whose depth is at most $q := 2(2k + \lceil \log \binom{k}{2} \rceil + 4) - 1 = O(k)$. First, if there is a multicolored clique in G , we will exhibit a strategy for White that forces a win in at most q moves. Second, if there is no multicolored clique, we will give a winning strategy for Black.

Normal play. If there is a multicolored clique C in G , then White can win in q moves in H . Here, *move* corresponds to what is usually called a *ply* or a *half-move* in games such as chess (where a move is actually a "move" from White plus a "move" from Black). Let $C = \{c_1, c_2, \dots, c_k\}$ with $c_i \in V_i$ for each $i \in [k]$. The first $4k$ moves are the following. For each $i \in [k]$, White plays in c_i^1 . If Black does not answer in $\{s_i, t_i, c_i^0, c_i^2, c_i^3\}$, White plays in c_i^2 and wins in four more

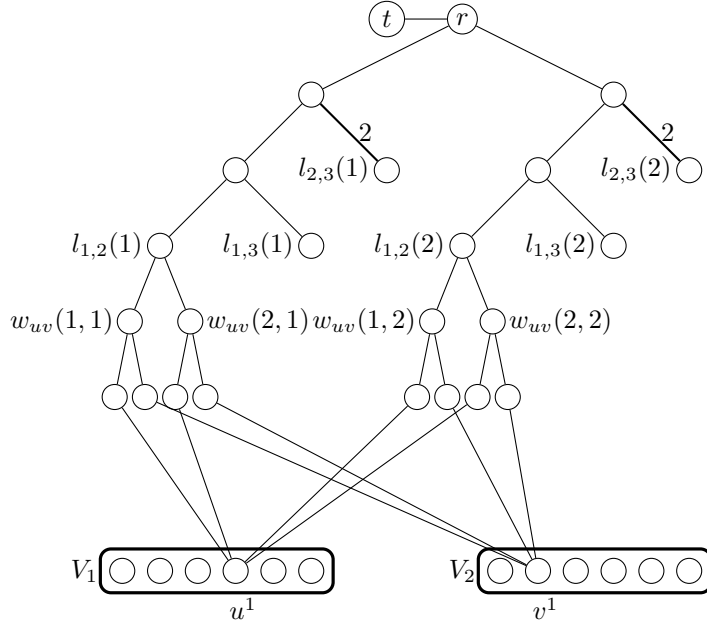


Figure 16: Edge gadgets for $k = 3$ (only one edge $uv \in E_{1,2}$ is represented).

moves; indeed, s and c_i^1 can be connected via c_i^0 or s_i , while c_i^2 and t_i can be connected via c_i^3 or t_i . To win some time, Black plays in s_i and White answers in c_i^0 . Then Black is forced to play in $\{c_i^2, c_i^3, t_i\}$.

On move $4k + 1$, White plays in r and will play the child in T of his last move in a subtree where Black has *not* played until he plays in $l_{i,j}(b)$ for some $i \neq j \in [k]$ and $b \in [2]$. As C is a clique, there is an edge e between c_i and c_j . By construction, the six vertices $w_e(1, b)$, $w_e(2, b)$, $w_{e,c_i}(1, b)$, $w_{e,c_i}(2, b)$, $w_{e,c_j}(1, b)$, and $w_{e,c_j}(2, b)$ are still empty. White's next move is $w_e(a, b)$ where a is such that Black has *not* played in $\{w_e(a, b), w_{e,c_i}(a, b), w_{e,c_j}(a, b)\}$. Then, White's final move is either $w_{e,c_i}(a, b)$ or $w_{e,c_j}(a, b)$, finishing a white path from t all across T to a vertex of the clique to s . One can check that it does not exceed q moves.

Proper defense. If there is no multicolored clique in G , we give a winning strategy for Black: not only White cannot force a win in at most q moves but he cannot win at all. A (y, z) -cut S of a graph J is a set of edges of J whose removal makes that y and z are in two different connected components. We say that S separates y and z . The following lemma presents a situation where Black can easily block his opponent.

Lemma 12. *If there is an (s, t) -cut M of H such that no vertex of M contains a white pebble and M is an induced matching, then Black wins, even if it is White to play.*

Proof. Whenever White plays a vertex u in M , Black answers the other endpoint of the unique edge of M which touches u . \square

If White plays in some vertex v^2 , then Black plays in v_1 and White has not achieved anything. If White plays in some vertex v^0, v^3, s_i , or t_i , then Black plays r and wins by Lemma 12, since the induced matching $\bigcup_{v \in V} \{v^1 v^2\}$ separates s from t . If White plays in a vertex v^1 such that $v \in V_i$, Black plays in s_i . Then, White should play in v^0 (indeed, it is pointless for White to play in u^d for some $d \in \{0, 1\}$ and $u \in V_i$, since Black can answer in u^{1-d}) and Black plays in t_i . Now, if White plays in another vertex u^1 with $u \in V_i$ (for the same i), then Black plays r and wins by Lemma 12 with the separator $\bigcup_{v \in V \setminus \{u\}} \{v^1 v^2\} \cup \{u^2 u^3\}$.

So far, we showed that, for each $i \in [k]$, White can play in at most one vertex of the form v^1 with $v \in V_i$, and then has to play in r . We distinguish two cases: (a) for each $i \in [k]$, White has played in a vertex v^1 with $v \in V_i$ when he plays in r (b) there is some $i \in [k]$ such that White has not yet played in a vertex v^1 with $v \in V_i$ when he plays in r .

We start with case (a). Let $A = \{d_1^1, d_2^1, \dots, d_k^1\}$ be the vertices which now contain a white pebble, with $d_i^1 \in V_i$ for each $i \in [k]$. As, $S := \{d_1, d_2, \dots, d_k\}$ is not a clique in G , there are two vertices d_i and d_j such that $d_i d_j \notin E$. Black plays in the root of T_2 , forcing White to play in the root of T_1 . Then, Black will systematically block the path which does *not* go to the leaf $l_{i,j}(1)$. At this point, $M' := \bigcup_{v \in V \setminus S} \{v^0 v^1\} \cup \bigcup_{v \in S} \{v^2 v^3\} \cup \bigcup_{a \in [2], xy \in E_{ij}, x \in \{d_i, d_j\}} \{w_{xy}(a, 1) w_{xy,x}(a, 1)\}$ separates s and t . Also, there is no white pebble in any of the endpoints of M' . Finally, M' is an induced matching. That observation is based on the key argument that $\bigcup_{a \in [2], xy \in E_{ij}, x \in \{d_i, d_j\}} \{w_{xy}(a, 1) w_{xy,x}(a, 1)\}$ cannot contain both $w_{xy}(a, 1) w_{xy,x}(a, 1)$ and $w_{xy}(a, 1) w_{xy,y}(a, 1)$ since that would imply that $\{x, y\} = \{d_i, d_j\}$, contradicting that $d_i d_j \notin E$. Therefore, Black wins by Lemma 12.

Now, we consider case (b). Black plays s_i where i is such that White has not played in any vertex v^1 with $v \in V_i$ yet. Now, whenever White plays in $u^1 \in V_i$, Black plays in u^0 . This way, White cannot connect any vertices of V_i to s . When White goes back to playing in T , Black forces him to the leaves $l_{i,j}(1)$ (and/or $l_{i,j}(2)$) for some arbitrary $j \in [k]$ and wins by Lemma 12 similarly to case (a). \square

Under the Exponential Time Hypothesis (ETH) that asserts that 3-SAT has no subexponential algorithm, MULTICOLORED CLIQUE cannot be solved in time $f(k)(n+m)^{o(k)}$ for any computable function f [43]. As our reduction linearly preserves the parameter, we have shown that SHORT GENERALIZED HEX cannot be solved in the same running time $f(k)(n+m)^{o(k)}$ unless the ETH fails, where n and m are the number of vertices and edges of the graph. A simple procedure solving SHORT GENERALIZED HEX in time $O^*(n^k)$ would consist of expanding all the continuations of length up to k and decide if Player 1 wins in this game subtree.

A class \mathcal{C} of graphs has *bounded local treewidth* if there is a function $F_{\mathcal{C}}$ such that for any $G \in \mathcal{C}$, $\text{tw}(G) \leq F_{\mathcal{C}}(\text{diam}(G))$ where $\text{tw}(G)$ is the treewidth of G and $\text{diam}(G)$ its diameter.

Theorem 5. SHORT GENERALIZED HEX *is FPT in every class of graphs with bounded local treewidth and closed by edge contractions.*

Proof. As we said above, we can contract the maximal connected components of vertices containing white pebbles (*white connected components*) into a single vertex containing a white pebble. By doing so, we obtain an equivalent graph where initially the white connected components have all size 1. Let W be the set of vertices containing initially a white pebble in this new graph. After at most k moves, the length of a potential (s, t) -path with white pebbles is bounded by $2\lceil \frac{k}{2} \rceil + 1 \leq k + 2$. Indeed, White plays at most $\lceil \frac{k}{2} \rceil$ new pebbles and the path can contain at most $\lceil \frac{k}{2} \rceil + 1$ initially placed white pebbles (every other two). It implies that the problem is expressible in first-order logic by the formula $\bigvee_{k' \leq k} [\exists v_1 \forall v_2 \exists v_3 \forall v_4 \dots \exists v_{k'} \bigwedge_{i \neq j \in [k']} v_i \neq v_j \wedge \bigvee_{l \leq k+2} [\exists u_1, u_2, \dots, u_l \in W \cup \{v_1, v_3, \dots, v_{k'}\} u_1 = s \wedge u_l = t \wedge E(u_1, u_2) \wedge E(u_2, u_3) \wedge \dots \wedge E(u_{l-1}, u_l)]]$. Frick and Grohe showed that any graph problem expressible in first-order logic can be solved in linear time in classes of graph with local bounded treewidth [44]. Yet, this algorithm is far from practical since its dependence in the formula is an exponential tower of height the number of quantifier alternations, so in our case height $\Theta(k)$. \square

Corollary 2. SHORT GENERALIZED HEX *is FPT in planar graphs.*

Proof. Planar graphs are closed by edge contractions and have bounded local treewidth [45]. \square

Corollary 3. SHORT HEX *is FPT.*

Proof. The underlying graph of HEX is planar. \square

6. Ultra-weak Solutions of Connection games

In Allis's game solving hierarchy [46, Section 1.5], a game is said to be *ultra-weakly* solved if the game-theoretic value of the initial position has been determined. HEX is unusual among strategy games in that an ultra-weak solution is known for any board size.

In this section, we first recall how HEX can be ultra-weakly solved to prove that the starting position is a theoretical first-player win. We then consider TWIXT, HAVANNAH, and SLITHER, and show that although some of the arguments carry over, it is impossible to ultra-weakly solve these games with the same method.

6.1. Hex

When John Nash discovered/invented the game HEX, one of his motivations was to find a non-trivial game with a non-constructive proof that the first player has a winning strategy in the initial position. The proof has two steps, prove that the initial position is not a draw, then prove that the initial position is not a second-player win.

The outcome of the initial position is not a game-theoretical because no HEX position can end in a draw. This is a well-known fact but the proof is non-trivial. We refer to Bogomolny [47] for a discussion of the issue and further references.

Lemma 13. *Draws cannot occur in HEX.*

To derive the second step, Nash developed a *strategy-stealing argument* which can be summed up as follows [1]. Suppose for a contradiction that the second player has a winning strategy σ in the initial position. Then a winning strategy for the first player can be obtained as: start with a random move, then apply σ pretending that the initial move did not occur. If σ ever recommends to play on the location of the initial move, then play another random move and carry on. Given that having an additional random stone on the board cannot hurt the first player, then we have developed a winning strategy for the first player too. Since the second and first player cannot both have a winning strategy at the same time, we can conclude that our hypothesis does not hold. Therefore the initial HEX position is not a second-player win when the swap rule is not used.

Lemma 14. *HEX is not a theoretical second-player win.*

The strategy-stealing argument applies to HEX for two reasons. First, it is a symmetrical game, *i.e.*, for any position, there is a map from moves that are legal for the second player to moves that are legal for the first player. Second, there are no *zugzwangs*: having an additional move never hurts.

This leads to the ultra-weak solution for HEX.

Theorem 6. *On any board size, the initial position in HEX is a theoretical first-player win.*

Proof. HEX is a turn-taking finite two-person game of perfect information with no chance. Zermelo's Theorem states that any such game is either a first-player win, a second-player win, or a draw. Lemma 13 and 14 rule out two possibilities and HEX can thus only be a first-player win. \square

6.2. TWIXT and HAVANNAH

TWIXT and HAVANNAH admit draws in theory but they occur very rarely in practice. The fact is relatively well-known in the game community, for instance Ewalds [4, Section 3.4.5] states that draws are possible for HAVANNAH sizes above 3 [4, Section 3.4.5]. Yet, to the best of our knowledge, a proof for arbitrary sizes has not appeared in the related literature. We give here a general construction for both games illustrated in Figure 17.

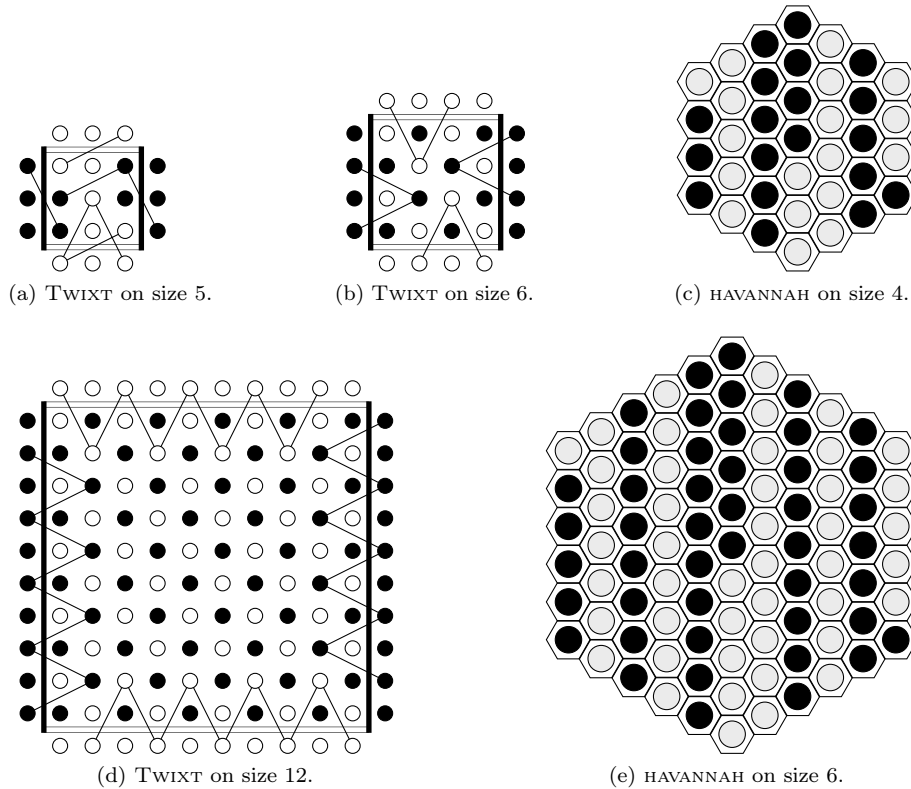


Figure 17: Drawn TWIXT and HAVANNAH positions. The TWIXT positions would also be drawn under the TWIXTPP variant.

Proposition 1. *It is possible to reach a drawn position from an empty TWIXT board of any size, and from an empty HAVANNAH board of any size strictly larger than 3.*

Proof. For TWIXT, any position on size 3 and 4 is a draw, and an example of draw for size 5 is given in Figure 17a. For larger sizes, if both players keep placing pegs in a checkered layout, no connection can ever go past two rows away from any border. This construction ensures a draw on board from size 6. Figure 17b and Figure 17d display the pattern for size 6 and 12 respectively.

In case of HAVANNAH, if both players keep placing pegs in a striped layout, no winning condition is ever met. Figure 17c and 17e displays the pattern for size 4 and 6. \square

Lemma 15. *TWIXT (resp. HAVANNAH) is not a theoretical second-player win.*

Proof. Zugzwangs are not possible in TWIXT (resp. HAVANNAH), and the game is symmetrical, therefore, the strategy-stealing argument applies to TWIXT (resp. HAVANNAH). \square



(a) Mutual zugzwang: no good move for Black and no good move for White.

(b) No move allowed for Black, only White has legal moves.

Figure 18: SLITHER positions with a shortage of moves.

Although the initial positions of TWIXT or HAVANNAH are no second-player wins, draws are possible for board sufficiently large, and so we cannot conclude that either is first-player win.

6.3. Slither

We now present some observations on and properties of SLITHER. Some of the observations made in this section have independently been pointed out earlier in the abstract game community, especially on BoardGameGeek.⁷ However, we prove for the first time that the standard SLITHER variant cannot end in a draw, thereby settling an open-problem often raised in this community.

The concept of *zugzwang* appears in CHESS and denotes a position in which the current player has no desirable move and would rather pass and have the opponent act. A *mutual zugzwang* is a position in which both players would rather have the opponent play. Although zugzwangs are virtually unheard of in typical connection games, where additional moves can never hurt you, things are different in SLITHER.

Proposition 2. *Zugzwangs and mutual zugzwangs can occur in SLITHER.*

Proof. In Figure 18a, if it is White (resp. Black) to play, only one move is available, moving stone on C2 (resp. C4) to *a* and placing a stone on *c* or equivalently moving to *c* and placing on *a*. Then Black (resp. White) wins by placing a stone on C2 (resp. moving stone B5 to C4 and placing a stone on *b*). □

As we have seen, the strategy-stealing argument can be applied to many other games including TWIXT, HAVANNAH, and games of the CONNECT(m, n, k) family [48]. Unfortunately, the argument cannot be applied to SLITHER.

Proposition 3. *Nash's strategy-stealing argument does not apply in SLITHER.*

⁷<http://boardgamegeek.com/thread/692652/what-if-there-no-legal-move>

Proof. Consider the White zugzwang position in Figure 18a. Had the A4 square not been occupied with a White stone, White would have a winning move: move B4 to b and place a stone on c . Since there are positions where having one too many stones on the board can make a player lose the game, Nash’s strategy-stealing argument does not apply. \square

Therefore, there is no theoretical indication yet that SLITHER is *not* a second-player win on an empty board. However, in practice it is a huge advantage to play first, so much that if the swap rule is used, it is recommended to swap no matter where the first move is played, including corner locations. The SLITHER-specific intuition behind this practical advice is that the game is dynamic and a player can bring back a stone from a corner towards the center, moving it closer every turn.

Draws would occur in SLITHER if there were position where no player has a legal move and yet no player connects their own sides. The SLITHER community had indeed identified non-terminal positions in which one of the player had no legal moves, but the opponent always had at least one move possible. In that case, the player with no legal moves would simply skip their turn.

Proposition 4. *There exist positions in which a player has no legal moves.*

Proof. For instance, Black has no legal move in the position Figure 18b. \square

The designer of SLITHER has long claimed that the game did not admit any draw. Since there were no formal proofs, many members of the community were left unconvinced and attempted to find counter-examples. They would submit positions on forums dedicated to SLITHER, and Corey Clark or some SLITHER players would point out that the counter-example was not valid, usually because the diagonal rule was not respected or because a legal move actually existed for one of the players. Before settling this question formally, let us point out that draws can actually occur when the board topology is not restricted to be rectangular.

Proposition 5. *Draws are possible when SLITHER is played on a cylinder or on a torus.*

Proof. In Figure 19a (resp. Figure 19b), if black (resp. black and white) sides are connected, then both players have no legal moves. \square

That draws are possible for some exotic boards should enhance the accidental aspect of our result on classical boards. There is probably no fundamental reasons why the following result is true, and the proof, which we defer to the appendix, consists of a large case analysis on the consequences of forbidding diagonal configurations and the possibility of moving stones.

Theorem 7. *Draws are not possible in SLITHER on rectangular boards.*

Proof. This theorem relies on two properties of SLITHER. First, if a SLITHER board is filled, then at least one player has a winning group. Second, if a board

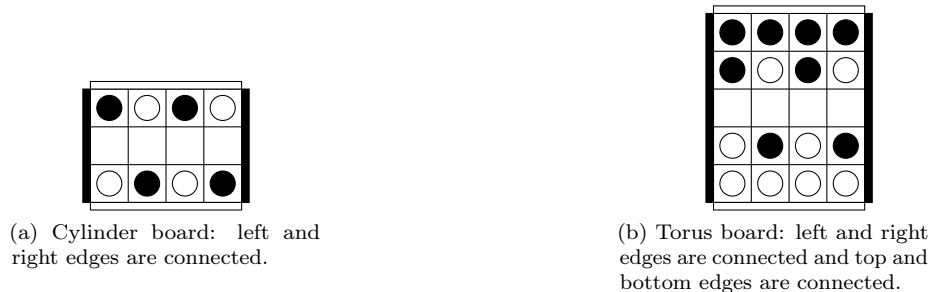


Figure 19: Drawn SLITHER positions on non-planar boards.

is not filled, then at least one player has a legal move. As a result, the game always continues, possibly with a player passing, until one player has a winning group. Since each move adds a stone to an empty intersection, these properties ensure that the game is bound to end after a finite number of turns with a winner. The Appendix proves the two essential properties mentioned above. \square

7. Discussion and Conclusion

Unlike CHESS, CHECKERS, and GO, the game of HEX is not played on a grid board but uses a hexagonal paving. The simplest connection game on a grid, using HEX rules on a checkered board where each cell is connected to the 4 vertically and horizontally adjacent ones, easily leads to draws in almost all cases and is not interesting for players. Designing an interesting connection game played on the usual grid board is challenging and TWIXT and SLITHER can be seen as two different solutions. Other solutions include João Pedro Neto’s GONNECT and Mark Steere’s CROSSWAY, and Browne’s book provides many more examples [2]. Each come with its own set of mechanics and trade-offs.

TWIXT eschews adjacent connections in favor of Knight’s move connections. In a sense, this changes the topology of the board fundamentally as, contrary to HEX and most other connection games, the graph of all possible direct connections is not planar anymore. Another consequence is that TWIXT allows draws, but this difference is mainly of theoretical interest as draws remain extremely rare in competitive play.

SLITHER keeps the grid topology intact but addresses draws by adding forbidden patterns and stone movement. Forbidding patterns alone would not be sufficient to prevent draws: for instance Figure 18a, 18b, and 9 would all be drawn if it was not for the possibility of moving existing pieces. Not only does moving pieces significantly change the dynamics of the game, but it also adds the theoretical possibility of zugzwangs. Again, this remains mainly of theoretical interest as zugzwangs virtually never seem to happen in practice.

HAVANNAH keeps the convenient hexagonal paving of HEX but explores a new dimension in the realm of connection game design by adding the *ring* winning

condition. A board could in principle support a ring or more from both players, except for the fact that the game ends as soon as the first one is completed. This leads to winning races where both players attempt to satisfy their own ring patterns, possibly non-intersecting, while disrupting the opponent's strategy by more immediate threats.

The hexagonal versus grid paving issue is of some importance in game design. It could seem, at first sight, that this difference would make reductions from HEX easier for HAVANNAH than for TWIXT and SLITHER. On the contrary, we only presented a HEX-based reduction for TWIXT and SLITHER, and in both reductions, the size of the resulting game is only linearly larger than the size of the input instance.

As a matter of fact, the ring winning condition in HAVANNAH makes obtaining a PSPACE-hardness proof by direct reduction from HEX quite difficult. Attempts by other authors and ourselves at finding a HEX-based reduction centered around the bridge or the fork winning conditions proved unfruitful.⁸ Instead, disregarding bridges and forks, a sequence of ring threats allowed us to reduce from GENERALIZED GEOGRAPHY, the problem used in Reisch's proof for HEX. On the other hand, if the HAVANNAH rules were changed to drop the ring winning condition, a direct simulation of HEX on the HAVANNAH board would be possible, just like it is for Y, and PSPACE-completeness of this new game would be straightforward.

In terms of future work, settling the complexity of other notable connection games remains a natural direction. In particular, GONNECT and LINES OF ACTION are good candidates. In LINES OF ACTION, each player starts with two groups of pieces and tries to connect all their pieces by moving these pieces and possibly capturing opponent pieces [50]. While the goal of LINES OF ACTION clearly makes it a connection game, the mechanics distinguishes it from more classical connection games as no pieces are added to the board and existing pieces can be moved or removed. GONNECT is best described as a mix between HEX and GO. Each player is trying to connect the opposite sides of a grid board, but captures are possible following the rules of GO. Both these games are quite different from the usual connection games and from the games studied in this paper in that the game length is not polynomially bounded. As a result, these games may very well not be PSPACE-complete.

Future work could also examine HAVANNAH, SLITHER, and TWIXT under a closer eye with the goal of obtaining theoretical results of practical relevance. For instance, a seemingly reasonable heuristic in SLITHER consists of playing the first move of a shortest sequence leading to victory, assuming the opponent passes. We leave as an open question whether such a shortest sequence can be efficiently computed. In fact, determining if such a sequence exists at all might not necessarily be easy and is reminiscent of Mazzoni and Watkins [18] NP-completeness result on single-player TWIXT. Another possible bridge towards actual game playing and solving is provided by parameterized complexity. Our

⁸Fabien and Olivier Teytaud, personal communication and [49].

FPT result on SHORT HEX indicates that solving algorithms need not depend exponentially on the board size, but only on the search depth. Could this result be turned into a practical algorithm for solving HEX endgames on large board sizes?

We find it remarkable that both TWIXT and SLITHER PSPACE-hardness proofs are direct reductions from HEX. It is quite rare indeed that an actually played game constitutes the basis of a computational complexity reduction. This is a new testimony to the special place that HEX occupies among strategy games, it combines being a relatively popular abstract game and a theoretical problem fundamental enough to reduce from.

Acknowledgments

The authors wish to thank Dániel Marx and Radu Curticapean for useful discussions and pointing out the paper [44]. The first author was supported by the ERC grant PARAMTIGHT: “Parameterized complexity and the search for tight complexity results” (no. 280152). The third author was supported by the Australian Research Council (project DE150101351).

References

- [1] J. van Rijswijck, Set colouring games, Ph.D. thesis, University of Alberta, 2006.
- [2] C. Browne, Connection Games: Variations on a Theme, AK Peters, 2005.
- [3] B. Arneson, R. B. Hayward, P. Henderson, Monte Carlo Tree Search in Hex, IEEE Transactions on Computational Intelligence and AI in Games 2 (4) (2010) 251–258, doi:10.1109/TCIAIG.2010.2067212.
- [4] T. Ewalds, Playing and Solving Havannah, Master’s thesis, University of Alberta, 2012.
- [5] P. T. Henderson, Playing and solving the game of Hex, Ph.D. thesis, University of Alberta, 2010.
- [6] A. M. Steane, Threat, support and dead edges in the Shannon game ArXiv e-prints.
- [7] A. M. Steane, Minimal and irreducible links in the Shannon game ArXiv e-prints.
- [8] V. V. Anshelevich, A hierarchical approach to computer Hex, Artificial Intelligence 134 (1-2) (2002) 101–120.
- [9] C. Browne, E. Powley, D. Whitehouse, S. Lucas, P. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakakis, S. Colton, A Survey of Monte Carlo Tree Search Methods, IEEE Transactions on Computational Intelligence and AI in Games 4 (1) (2012) 1–43, ISSN 1943-068X, doi:10.1109/TCIAIG.2012.2186810.

- [10] J. Bruno, L. Weinberg, A constructive graph-theoretic solution of the Shannon switching game, *IEEE Transactions on Circuit Theory* 17 (1) (1970) 74–81.
- [11] E. D. Demaine, R. A. Hearn, Playing games with algorithms: algorithmic combinatorial game theory, in: R. J. Nowakowski (Ed.), *Games of No Chance III*, Cambridge University Press, 3–56, 2009.
- [12] R. A. Hearn, E. D. Demaine, *Games, Puzzles, and Computation*, A K Peters, 2009.
- [13] É. Bonnet, F. Jamain, A. Saffidine, On the Complexity of Trick-Taking Card Games, in: F. Rossi (Ed.), *23rd International Joint Conference on Artificial Intelligence (IJCAI)*, AAAI Press, Beijing, China, 482–488, 2013.
- [14] S. Even, R. E. Tarjan, A combinatorial problem which is complete in polynomial space, *Journal of the ACM (JACM)* 23 (4) (1976) 710–719.
- [15] S. Reisch, Hex ist PSPACE-vollständig, *Acta Informatica* 15 (2) (1981) 167–191.
- [16] S. Kiefer, Die Menge der virtuellen Verbindungen im Spiel Hex ist PSPACE-vollständig, Studienarbeit Nr. 1887, Universität Stuttgart, 2003.
- [17] Y. Björnsson, R. Hayward, M. Johanson, J. van Rijswijck, Dead cell analysis in Hex and the Shannon game, in: A. Bondy, J. Fonlupt, J.-L. Fouquet, J.-C. Fournier, J. L. Ramírez Alfonsín (Eds.), *Graph Theory in Paris*, Springer, 45–59, 2007.
- [18] D. Mazzoni, K. Watkins, Uncrossed Knight Paths is NP-complete, 1997.
- [19] T. Maarup, Hex: Everything You Always Wanted to Know About Hex But Were Afraid to Ask, Master’s thesis, Department of Mathematics and Computer Science, University of Southern Denmark, Odense, Denmark, 2005.
- [20] B. Whitehill, Alex Randolph—A Life of Games, Knucklebones URL <http://thebiggamehunter.com/inventors/game-designers/alex-randolph/>.
- [21] T. Yato, On the NP-completeness of the Slither Link Puzzle, in: *Notes of the 74th meeting of IPSJ SIG Algorithms*, 25–32, 2000.
- [22] R. Yoshinaka, T. Saitoh, J. Kawahara, K. Tsuruma, H. Iwashita, S.-i. Minato, Finding All Solutions and Instances of Numberlink and Slitherlink by ZDDs, *Algorithms* 5 (2) (2012) 176–213, ISSN 1999-4893, doi:10.3390/a5020176.
- [23] J. R. Huber, Computer game playing: the game of Twixt, Master’s thesis, Rochester Institute of Technology, 1983.
- [24] K. Moesker, TwixT: theory, analysis and implementation, Master’s thesis, Maastricht University, 2009.

- [25] J. Uiterwijk, K. Moesker, Mathematical Modelling in TwixT, in: B. Löwe (Ed.), *Logic and the Simulation of Interaction and Reasoning*, 29, 2009.
- [26] F. Teytaud, O. Teytaud, Creating an Upper-Confidence-Tree Program for Havannah, in: H. J. van den Herik, P. Spronck (Eds.), *12th International Conference on Advances in Computer Games (ACG)*, vol. 6048 of *Lecture Notes in Computer Science*, Springer, Pamplona, Spain, ISBN 978-3-642-12992-6, 65–74, 2009.
- [27] R. J. Lorentz, Improving Monte–Carlo Tree Search in Havannah, in: H. J. Herik, H. Iida, A. Plaat (Eds.), *Computers and Games*, vol. 6515 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, ISBN 978-3-642-17927-3, 105–115, 2011.
- [28] É. Bonnet, F. Jamain, A. Saffidine, Havannah and TwixT are PSPACE-complete, in: H. J. van den Herik, H. Iida, A. Plaat (Eds.), *8th International Conference on Computers and Games (CG)*, vol. 8427 of *Lecture Notes in Computer Science*, Springer, Yokohama, Japan, 174–185, 2013.
- [29] E. Bonnet, F. Jamain, A. Saffidine, Draws, Zugzwangs, and PSPACE-Completeness in the Slither Connection Game, in: A. Plaat, H. J. van den Herik, W. A. Kusters (Eds.), *14th International Conference on Advances in Computer Games (ACG)*, vol. 9525 of *Lecture Notes in Computer Science*, Springer, Leiden, The Netherlands, ISBN 978-3-319-27991-6, 160–176, doi:10.1007/978-3-319-27992-3, 2015.
- [30] T. J. Schaefer, On the complexity of some two-person perfect-information games, *Journal of Computer and System Sciences* 16 (2) (1978) 185–225, ISSN 0022-0000, doi:10.1016/0022-0000(78)90045-4.
- [31] S. Iwata, T. Kasai, The Othello game on an $n \times n$ board is PSPACE-complete, *Theoretical Computer Science* 123 (2) (1994) 329–340.
- [32] T. Furtak, M. Kiyomi, T. Uno, M. Buro, Generalized Amazons is PSPACE-complete, in: L. P. Kaelbling, A. Saffiotti (Eds.), *19th International Joint Conference on Artificial Intelligence (IJCAI)*, 132–137, 2005.
- [33] D. Lichtenstein, M. Sipser, Go is polynomial-space hard, *Journal of the ACM (JACM)* 27 (2) (1980) 393–401.
- [34] R. B. Hayward, J. van Rijswijk, Hex and combinatorics, *Discrete Mathematics* 306 (19-20) (2006) 2515–2528, doi:10.1016/j.disc.2006.01.029.
- [35] M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, S. Saurabh, *Parameterized Algorithms*, Springer, ISBN 978-3-319-21274-6, doi:10.1007/978-3-319-21275-3, 2015.
- [36] R. G. Downey, M. R. Fellows, *Fundamentals of Parameterized Complexity*, Texts in Computer Science, Springer, ISBN 978-1-4471-5558-4, doi:10.1007/978-1-4471-5559-1, 2013.

- [37] R. G. Downey, M. R. Fellows, Parameterized Complexity, Monographs in Computer Science, Springer, ISBN 978-1-4612-6798-0, doi:10.1007/978-1-4612-0515-9, 1999.
- [38] A. Scott, On the parameterized complexity of finding short winning strategies in combinatorial games, Ph.D. thesis, University of Victoria, 2009.
- [39] A. Scott, U. Stege, Parameterized Chess, in: M. Grohe, R. Niedermeier (Eds.), 3rd International Workshop on Parameterized and Exact Computation (IWPEC), vol. 5018 of *Lecture Notes in Computer Science*, Springer, Victoria, Canada, ISBN 978-3-540-79722-7, 172–189, 2008.
- [40] A. Scott, U. Stege, Parameterized pursuit-evasion games, *Theoretical Computer Science (TCS)* 411 (43) (2010) 3845–3858, doi:10.1016/j.tcs.2010.07.004.
- [41] H. Björklund, S. Sandberg, S. Vorobyov, On Fixed-Parameter Complexity of Infinite Games, 2003.
- [42] F. V. Fomin, D. Marx, FPT Suspects and Tough Customers: Open Problems of Downey and Fellows, in: H. L. Bodlaender, R. Downey, F. V. Fomin, D. Marx (Eds.), *The Multivariate Algorithmic Revolution and Beyond — Essays Dedicated to Michael R. Fellows on the Occasion of His 60th Birthday*, vol. 7370 of *Lecture Notes in Computer Science*, Springer, ISBN 978-3-642-30890-1, 457–468, doi:10.1007/978-3-642-30891-8, 2012.
- [43] J. Chen, X. Huang, I. A. Kanj, G. Xia, Strong computational lower bounds via parameterized complexity, *Journal of Computer and System Sciences (JCSS)* 72 (8) (2006) 1346–1367.
- [44] M. Frick, M. Grohe, Deciding first-order properties of locally tree-decomposable structures, *Journal of the ACM (JACM)* 48 (6) (2001) 1184–1206, doi:10.1145/504794.504798.
- [45] N. Robertson, P. D. Seymour, Graph minors. III. Planar tree-width, *Journal of Combinatorial Theory, Series B* 36 (1) (1984) 49–64, doi:10.1016/0095-8956(84)90013-3.
- [46] L. V. Allis, Searching for Solutions in Games an Artificial Intelligence, Ph.D. thesis, Vrije Universitat Amsterdam, Department of Computer Science, Rijksuniversiteit Limburg, 1994.
- [47] A. Bogomolny, Hex Can't End in a Draw *from Interactive Mathematics Miscellany and Puzzles*, <http://www.cut-the-knot.org/Curriculum/Games/HexTheory.shtml>, Accessed: 2016-02-06.
- [48] M. Y. Hsieh, S.-C. Tsai, On the fairness and complexity of generalized k-in-a-row games, *Theoretical Computer Science* 385 (1–3) (2007) 88–100, doi:10.1016/j.tcs.2007.05.031.

- [49] F. Teytaud, O. Teytaud, On the huge benefit of decisive moves in Monte-Carlo Tree Search algorithms, in: G. N. Yannakakis, J. Togelius (Eds.), IEEE Conference on Computational Intelligence and Games (CIG), IEEE, Copenhagen, Denmark, ISBN 978-1-4244-6295-7, 359–364, doi:10.1109/ITW.2010.5593334, 2010.
- [50] M. H. M. Winands, Y. Björnsson, J.-T. Saito, Monte Carlo Tree Search in Lines of Action, IEEE Transactions on Computational Intelligence and AI in Games 2 (4) (2010) 239–250, doi:10.1109/TCIAIG.2010.2061050.

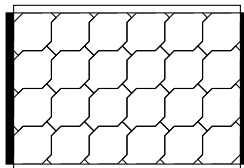


Figure A.20: 6×4 HEX board represented on a rectangular SLITHER board.

Appendix A. Draws are impossible in SLITHER

Stating that SLITHER does not feature any draw actually corresponds to the following three more elementary statements: each filled SLITHER board has winning groups for at least one player, each filled SLITHER board has winning groups for no more than one player, and each non-filled SLITHER board has at least one legal move for one of the two players. The first two statements can be obtained rather directly from the equivalent ones in HEX, thanks to the diagonal rule. The third statement is much more involved and requires a careful case analysis on non-filled boards.

Lemma 16. *If a SLITHER board is filled, then exactly one of the two players wins.*

Proof. Recall that HEX can be played on a rectangular board provided we add a link between each pair of king-adjacent squares along one specified diagonal direction, as in Figure A.20. The forbidden configuration rule ensures that this king-adjacent diagonal connection is respected in SLITHER, although it is indirect. Therefore, any filled $m \times n$ SLITHER board can be mapped onto an equivalent $m \times n$ HEX board such that any pair of SLITHER squares is connected if and only if the corresponding pair of HEX cells is connected. Since any filled HEX board is won by exactly one player, we have the desired SLITHER result. □

Theorem 8. *On a rectangular board, as long as there is at least one empty square, at least one of the two players has a legal move.*

Proof. We adopt the following proof technique.⁹ We assume for a contradiction that we have a non-filled position with no legal moves for any players. We start from an empty square and make deduction concerning its surrounding so as to constrain the occupancy of the nearby squares. Each constraint is deduced based on the established occupancies and from the no legal moves assumption or from the diagonal rule. We may perform a split case analysis on squares that are not constrained enough to have a definite status. In each case, however, we finally arrive at a position with a legal move that cannot be prevented by adding any further constraints.

⁹No part of the argument will rely upon the color of the board edge.

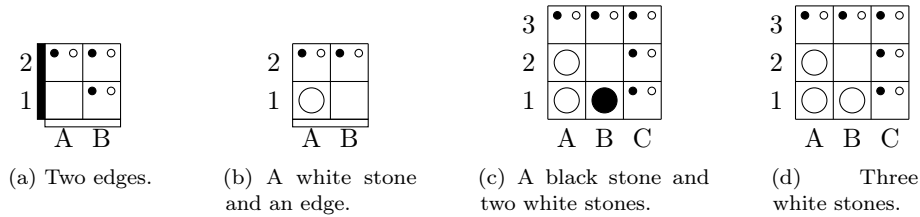


Figure A.21: Case analysis for the bottom left surroundings of the empty square.

As we add constraints to forbid legal moves from either player, we liberally extend the size of the pattern around the empty square. If any such extension was not possible because we would have reached the limit of the board, then it would not be possible to forbid the desired legal move and our case would be proved. We can therefore disregard the possibility of inadvertently meeting an edge of the board as we extend our patterns, at least for the sake of this argument.

In addition to the regular three types of squares, \circ white stone, \bullet black stone, and \square empty, we add the following ones: \square *no constraints yet*, \square *cannot hold a white stone*, and \square *cannot hold a black stone*.

Consider a rectangular board. If there is at least an empty square on the board, then there is at least an empty square s such that one of the following 4 conditions on the bottom and left neighbors of s is satisfied. Either s is in the bottom-left corner (Figure A.21a), or s is on the bottom edge and its left neighbor is occupied (Figure A.21b), or s has three neighboring stones of different colors (Figure A.21c), or these stones are all of the same color (Figure A.21d). We can assume w.l.o.g. that a majority of the bottom-left neighboring stones are white.

The first two cases are treated in Section Appendix B, the third case is treated in Section Appendix C, and the last case in Section Appendix D. In each case, we arrive to the conclusion that at least one player can move. \square

Appendix B. A square in a corner or on the edge of the board

If there is an empty square in a corner, as depicted in Figure A.21a, then placing a stone on that very square is a legal move for at least one player.

If there is an empty square on an edge, we start from the situation in Figure A.21b and use the following reasoning to constrain the surrounding and obtain Figure B.22. C2 needs a white stone to forbid White's move B1, and C1 cannot be white. A2 needs a black stone to forbid Black's move B1. B2 needs to be empty to forbid White's and Black's move B1. C1 cannot be empty to forbid White's move C1. A3 needs a white stone to forbid White's move A1B1-B2, and B3 cannot be white. C3 needs a black stone to forbid Black's move C1B1-B2, and B3 cannot be black. Similarly, A4 needs a black stone, C4 needs a white

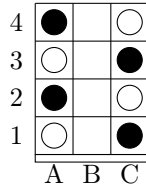


Figure B.22: The case in Figure A.21b with a few deducible constraints filled in.

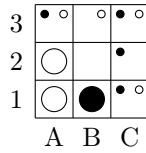


Figure C.23: The case in Figure A.21c with a few deducible constraints filled in.

stone, and B4 needs to be empty, so as to forbid White's move A1B2-B3 and Black's move C1B2-B3.

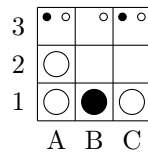
But then, C3B2-B1 is a legal move for Black.

Appendix C. A square with two white stones and a black stone

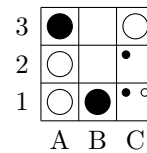
We start from the situation in Figure A.21c and use the following reasoning to constrain the surrounding and obtain Figure C.23. The C2 square cannot contain a white stone, otherwise B2 is a legal move for White. Similarly, B3 cannot contain a black stone, otherwise B2 is a legal move for Black. To forbid White's move B2, there should be a white stone on C1 or on C3 (Figure C.24).

Appendix C.1. Figure C.24a

C2 cannot contain a black stone due to the diagonal rule (since B2 is empty by assumption), and has to be empty. Now, we distinguish two subcases: either B3 is white, or it is empty (Figure C.25).

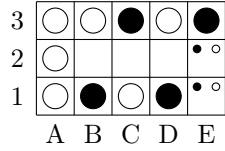


(a) Assume C1 is white.

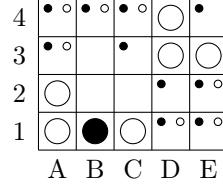


(b) Assume C3 is white.

Figure C.24: Case analysis for Figure C.23, either C1 is white or C3 is white.

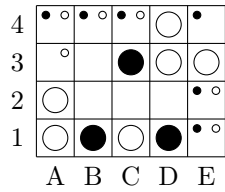


(a) Assume B3 is white.

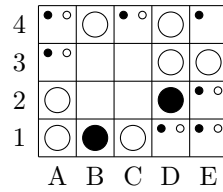


(b) Assume B3 is empty.

Figure C.25: Case analysis for Figure C.24a: B3 is either white or empty.



(a) Assume C3 is black.



(b) Assume C3 is empty.

Figure C.26: Case analysis for Figure C.25b: C3 is either black or empty.

Appendix C.1.1. Figure C.25a

A3 contains a white stone, by the diagonal rule. C3 needs a black stone to forbid Black's move B2. D1 needs a black stone to forbid Black's move B1B2-C2. D3 needs a white stone to forbid White's move C1B2-C2. D2 needs to be empty, by the diagonal rule on stones C1 and C3. E3 needs a black stone to forbid Black's move B1C2-D2.

But, in that situation, D3C2-B2 is a legal move for White.

Appendix C.1.2. Figure C.25b

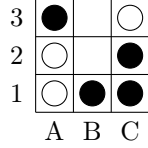
To forbid White's move C2, D3 needs a white stone and C3 and D2 cannot contain a white stone. To forbid White's move D3C2-B2, there should be white stones on E3 and D4 and E4 should not contain a white stone. Now, we distinguish two subcases: either C3 is black, or it is empty (Figure C.26).

Appendix C.1.2.1. Figure C.26a. A3 cannot contain a black stone to forbid Black's move B3. D1 needs a black stone to forbid Black's move C3B2-C2. D2 cannot contain a black stone by the diagonal rule, and has to be empty.

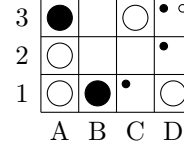
But then, move B1C2-D2 is legal for Black.

Appendix C.1.2.2. Figure C.26b. B4 needs a white stone to forbid White's move C3. D2 needs a black stone to forbid Black's move C3.

But then, B1C2-C3 is legal for Black.

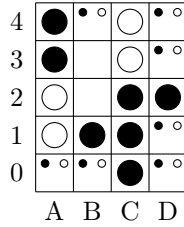


(a) Assume C2 is black.

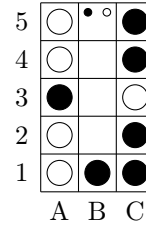


(b) Assume C2 is empty.

Figure C.27: Case analysis for Figure C.24b: C2 is either black or empty.



(a) Assume A4 is black and C4 is white.



(b) Assume A4 is white and C4 is black.

Figure C.28: Case analysis for Figure C.27a: the contents of A4 and C4 is white.

Appendix C.2. Figure C.24b

A3 needs a black stone to forbid Black's move B2. B3 cannot contain a white stone by the diagonal rule (with A2) and has to be empty. Now, we distinguish two subcases: either C2 is black, or it is empty (Figure C.27).

Appendix C.2.1. Figure C.27a

To forbid White's move C3B2-B3 and Black's move A3B2-B3, there should be at least one white stone and one black stone in {A4, C4}. So, we distinguish further between {A4 black, C4 white} and {A4 white, C4 black} (Figure C.28).

Appendix C.2.1.1. Figure C.28a. D2 needs a black stone to forbid Black's move C2B2-B3, and C0 needs a black stone to forbid Black's move C1B2-B3.

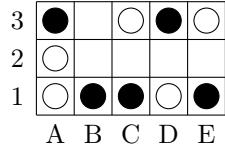
But then, B1B2-B3 is a legal move for Black.

Appendix C.2.1.2. Figure C.28b. By the diagonal rule, square B4 has to be empty. C5 (as well as D4) needs a black stone to forbid Black's move C4B3-B2. Symmetrically, A5 needs a white stone to forbid White's move A4B3-B2.

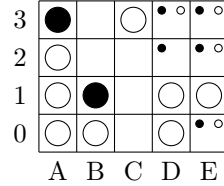
But then, C3B2-B4 is a legal move for White.

Appendix C.2.2. Figure C.27b.

To forbid White's move C2, there should be a white stone on D1 but no white stones on C1 nor D2. We distinguish two subcases: C1 contains a black stone, or it is empty (Figure C.29).



(a) Assume C1 is black.



(b) Assume C1 is empty.

Figure C.29: Case analysis for Figure C.27b: C1 is either black or empty.

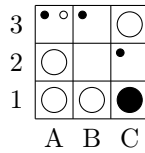


Figure D.30: The case in Figure A.21d with a few deducible constraints filled in.

Appendix C.2.2.1. Figure C.29a.. D3 needs a black stone to forbid Black's move C2 (since B3 is empty). Then, by the diagonal rule, square D2 can only be empty. E3 needs a white stone to forbid White's move C3D2-C2. E1 needs a black stone to forbid Black's move D3C2-D2.

But then, D1C2-B2 is a legal move for White.

Appendix C.2.2.2. Figure C.29b.. The only way to forbid White's move D1C2-B2 is to add two white stones on D0 and on E1. To forbid White's move C1, there should be a white stone on B0 (and a white stone on A0, by the diagonal rule), and no white stones on C0. C0 cannot contain a black stone because of the diagonal rule

But then, C0 is a legal move for White.

Appendix D. A square with three white stones

We start from the situation in Figure A.21d. To forbid White's move B2, there should be a white stone on C3, but no white stones on B3 nor C2. To forbid Black's move B2, there should be a black stone on C1 or A3, say C1 w.l.o.g. (see Figure D.30).

Therefore, B3 and C2 are empty or contain a black stone. They cannot both contain a black stone since B2 is empty. We thus distinguish three cases: B3 and C2 are empty, B3 contains a black stone, and C2 contains a black stone (Figure D.31).

Appendix D.1. Figure D.31a

D3 needs a black stone to forbid Black's move C2. D1 needs a white stone to forbid White's move C3B2-C2. E3 needs a white stone to forbid White's move C3B2-D2. E1 needs a black stone to forbid Black's move D3C2-D2.

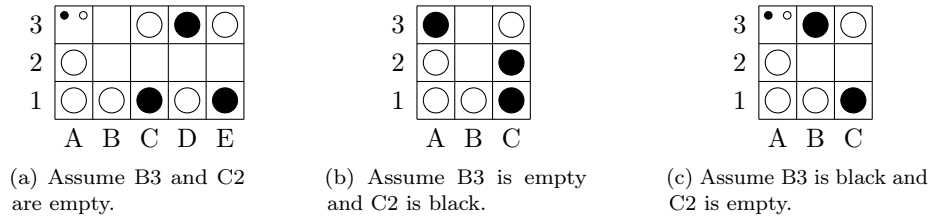


Figure D.31: Case analysis for Fig D.30: the contents of B3 and C2.

But then, D1C2-B2 is legal for White.

Appendix D.2. Figure D.31b

A3 needs a black stone to forbid Black's move B2. This case is equivalent to the case of Figure C.27a under color and spatial symmetry.

Appendix D.3. Figure D.31c

Let us consider cases for the contents of A3. If A3 contains a black stone, then we obtain a position equivalent to Figure C.27a under color and spatial symmetry.

If A3 is empty or white, then a similar proof to Figure D.31a still holds. Indeed, D3 needs a black stone to forbid Black's move B3B2-C2 and D1 needs a white stone to forbid White's move C3B2-C2. The same way, E3 needs a white stone to forbid White's move C3B2-D2, and then E1 needs a black stone to forbid Black's move C1B2-D2.

But then, D1C2-B2 is legal for White.