Electronic Colloquium on Computational Complexity, Report No. 97 (2020)

6-Uniform Maker-Breaker Game Is PSPACE-Complete

Md Lutfar Rahman Thomas Watson

University of Memphis

June 29, 2020

Abstract

In a STOC 1976 paper, Schaefer proved that it is PSPACE-complete to determine the winner of the so-called Maker-Breaker game on a given set system, even when every set has size at most 11. Since then, there has been no improvement on this result. We prove that the game remains PSPACE-complete even when every set has size 6.

1 Introduction

The Maker-Breaker game is a perfect-information game played on a set system—a collection of subsets of some finite universe. The two players, called Maker and Breaker, alternate turns. In each turn, the current player claims a previously-unclaimed element of the universe as his own. Maker wins if he claims every element in at least one subset. Breaker wins if he claims at least one element in every subset. There are no draws, and for every set system, one of the players has a strategy that guarantees that he wins. The popular game of Hex can be viewed as a Maker-Breaker game.

Maker-Breaker games were introduced in the influential paper [ES73], which provided a sufficient condition for Breaker to win (and is often considered the forerunner to the method of conditional probabilities). There is a very substantial literature on determining which player has a winning strategy, for various kinds of set systems (and for many generalizations and variants of Maker-Breaker games). We refer to [HKSS14] for a survey. Some cornerstones of this literature are:

- When the universe is the set of edges of an undirected graph with distinguished nodes s and t, and the subsets are s-t paths (this special case is called the "Shannon switching game"), Lehman [Leh64] characterized which player can win, in terms of combinatorial properties of the graph.
- When the universe is the set of edges of a sufficiently large complete undirected graph, and the subsets are Hamiltonian cycles, Chvátal and Erdös [CE78] proved that Maker can win.

Given the effort that has gone into determining the winner for various set systems, it is natural to consider the possibility of automating this process. In other words, let us view this as a computational problem and investigate how efficiently it can be solved.

> *What is the computational complexity of determining which player has a winning strategy in the Maker-Breaker game on a given set system?*

ISSN 1433-8092

In a seminal paper, Schaefer [Sch76, Sch78] proved that the problem is PSPACE-complete, even when the set system has *width* 11, which means each subset in the system has size at most 11. (A simplified proof of PSPACE-completeness for unbounded width was given in [Bys04].) Reductions from this theorem have been used for many other PSPACE-completeness results [FG87, Sla00, Sla02, AS03, Bys04, DH08, Hea09, TDU11, vV13, FGM+15, BDK+16, DGPR18, GHIK19, CPSS19, RW20a].

Since Schaefer's PSPACE-completeness result first appeared in 1976, there has been no improvement on the width 11. We make the first progress in 44 years: Determining the winner of the Maker-Breaker game remains PSPACE-complete even for set systems of width 6. As we note later, this also implies PSPACE-completeness of Maker-Breaker for set systems that are 6-uniform, meaning that every subset has size exactly 6.

1.1 CNF games

In this section, we introduce "CNF games," a broader sense of games that includes Maker-Breaker as a special case.

- In the *ordered* game, the input consists of a conjunctive normal form (CNF) formula φ and an ordered list of variables $\{x_{2n}, x_{2n-1}, \ldots, x_2, x_1\}$ that contains all variables of φ. Player 1 is called T because his goal is to make φ true, and player 2 is called F because his goal is to make φ false. In the first round, T assigns a bit value for x_{2n}, then F assigns a bit value for x_{2n-1}. In the next round, T assigns x_{2n-2}, then F assigns x_{2n-3}, and so on for n rounds. The winner depends on whether φ is satisfied by the resulting assignment. In other words, which player has a winning strategy is determined by whether the following quantified boolean formula is true:

$$(\exists x_{2n})(\forall x_{2n-1}) \cdots (\exists x_2)(\forall x_1) : \varphi(x_1, \ldots, x_{2n})$$

  The problem w-TQBF is to determine which player has a winning strategy, under the restriction that φ has width w (every clause has at most w literals). It is known that 2-TQBF is NL-complete [APT79] and 3-TQBF is PSPACE-complete [SM73].

- In the *unordered* game, the input consists of a CNF φ, a set X of variables that contains all variables of φ (and possibly more), and an indication of which player (T or F) gets the first move. Again, T and F alternate turns assigning bit values to variables, and the winner depends on whether φ is satisfied by the resulting assignment. But now, each turn consists of picking which remaining variable to assign, as well as which bit to assign it. The unordered game more closely resembles real-world games in which the same moves are available to both players. The problem G_w is to determine which player has a winning strategy, under the restriction that φ has width w. The paper [RW20a] originated the G_w notation and showed that G_2 is in L and G_5 is PSPACE-complete.

- The *unordered positive* game is just the unordered game under the restriction that φ must be a positive (a.k.a. monotone) CNF—it only has unnegated literals. In this game, it would never be advantageous for T to assign 0 to a variable, or for F to assign 1 to a variable. Thus we can assume each move consists of T picking a remaining variable and assigning it 1, or F picking a remaining variable and assigning it 0. If we view each clause of φ as a subset of X (the set of variables), then the unordered positive game is equivalent to the Maker-Breaker game on the set system corresponding to (φ, X), where F is Maker (he wants to assign every variable in at least one clause) and T is Breaker (he wants to assign at least one variable in every clause). The problem G_w^+ is the restriction of G_w to positive w-CNFs, i.e., determining whether Maker or Breaker has a winning strategy on a given set system of width w. Thus, Schaefer's theorem [Sch76, Sch78] can be stated as: G_{11}^+ is PSPACE-complete.

Previously, the authors conjectured that G_3^+, and perhaps even G_3, might actually be tractable. These problems have been shown to be tractable—indeed, in L—under various restrictions on the 3-CNF [Kut05, RW20b]. The unordered CNF game seems qualitatively very different from its ordered counterpart. Width 6 might not be optimal for PSPACE-completeness of Maker-Breaker (though it appears to be a barrier for our proof technique), but it is unclear what the optimal width ought to be.

In this paper, we prove the following three results:

Theorem 1. G_6^+ *is* PSPACE*-complete.*

Theorem 2. G_5^+ *is* NL*-hard.*

Theorem 3. G_4 *is* NL*-hard.*

In Table 1 we summarize the state-of-the-art for the ordered, unordered, and unordered positive CNF games.

<table>
<tr><th>$w \rightarrow$</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th></tr>
<tr><td>w-TQBF</td><td>NL-complete
[APT79]</td><td>PSPACE-complete
[SM73]</td><td></td><td></td><td rowspan="2"></td></tr>
<tr><td>G_w</td><td rowspan="2">L
[RW20a]</td><td>L
under restrictions
[RW20b]</td><td>NL-hard
[Theorem 3]</td><td>PSPACE-complete
[RW20a]</td></tr>
<tr><td>G_w^+</td><td>L
under restrictions
[Kut05]</td><td>Unknown</td><td>NL-hard
[Theorem 2]</td><td>PSPACE-complete
[Theorem 1]</td></tr>
</table>

Table 1: Results

Each game has four different patterns for "who has the first move" and "who has the last move." For $a, b \in \{\mathrm{T}, \mathrm{F}\}$ we use the subscript $a \cdots b$ to indicate that player a goes first and b goes last. For example, $G_{6,\mathrm{T}\cdots\mathrm{F}}^+$ is G_6^+ restricted to instances where T has the first move and F has the last move (which necessitates $|X|$ being even). With no such subscript, an instance of G_6^+ must specify which player goes first (and then the parity of $|X|$ determines who goes last). We prove that G_6^+ is PSPACE-complete for each of the four possible patterns, and similarly for G_5^+ being NL-hard, but we are only able to show NL-hardness of G_4 for the patterns $\mathrm{T} \cdots \mathrm{F}$ and $\mathrm{F} \cdots \mathrm{F}$.

Our proof of Theorem 1 follows a similar high-level outline as the proof that G_{11}^+ is PSPACE-complete from [Sch76, Sch78], using a reduction from 3-TQBF. The key is to trade size for width—we develop a gadget for simulating a round of the ordered game, using more variables and clauses but lower width than the gadget from [Sch76, Sch78]. Our correctness analysis also uses a new perspective on the case where T is supposed to win (which is much trickier than the case where F is supposed to win, since T must satisfy every clause whereas F only needs to falsify one clause). To frame T's winning strategy in the event that F "misbehaves," we make use of ideas from the recent paper [RW20b].

The proof of Theorem 1 also yields Theorem 2. Theorem 3 holds by an elementary but new reduction from 2-SAT.

2 Proof of Theorem 1 (and Theorem 2)

We prove Theorem 1 in Section 2.1. In Section 2.2 we provide a streamlined proof of a special case of a lemma from [RW20b], which is needed for the proof of Theorem 1. Then we prove a series of corollaries in Section 2.3, which cover all the patterns for both Theorem 1 and Theorem 2.

2.1 Proof of Theorem 1

We show 3-TQBF $\leqslant \mathrm{G}^{+}_{6,\mathrm{T}\cdots\mathrm{F}}$. Suppose an instance of 3-TQBF is given by

$$(\exists x_{2n})(\forall x_{2n-1})\cdots(\exists x_2)(\forall x_1) : F_1 \wedge F_2 \wedge \cdots \wedge F_m$$

where each F_k is a clause with width $\leqslant 3$. We construct an instance of $\mathrm{G}^{+}_{6,\mathrm{T}\cdots\mathrm{F}}$ as (φ^+, X) where φ^+ is a positive 6-CNF and X is the set of variables in it, such that T has a winning strategy in the 3-TQBF game iff T has a winning strategy in the $\mathrm{G}^{+}_{6,\mathrm{T}\cdots\mathrm{F}}$ game on (φ^+, X).

A 3-TQBF round $(\exists x_i)(\forall x_{i-1})$, where $i \in \{2, 4, 6, \ldots, 2n\}$, will correspond to 16 variables in X and 14 clauses in φ^+. Four of the 16 variables are $\{x_i, \overline{x}_i, x_{i-1}, \overline{x}_{i-1}\}$. Here, $\overline{x}_i$ is the name of an unnegated variable, distinct from the variable x_i. The variables x_i and $\overline{x}_i$ do not necessarily get assigned opposite values. Similarly for x_{i-1} and $\overline{x}_{i-1}$. The other 12 variables associated with a 3-TQBF round $(\exists x_i)(\forall x_{i-1})$ are $\{u_{6i}, u_{6i-1}, \ldots, u_{6i-11}\}$. (This variable naming scheme is borrowed from [Sch76, Sch78].) In the $\mathrm{G}^{+}_{6,\mathrm{T}\cdots\mathrm{F}}$ game, we define "legitimate" gameplay corresponding to a 3-TQBF round $(\exists x_i)(\forall x_{i-1})$ as follows:

1. T plays one of x_i, $\overline{x}_i$
2. F plays the remaining variable in the pair x_i, $\overline{x}_i$
3. T plays u_{6i}
4. F plays u_{6i-1}
5. T plays u_{6i-2}
6. F plays u_{6i-3}
7. T plays u_{6i-4}
8. F plays one of x_{i-1}, $\overline{x}_{i-1}$
9. T plays the remaining variable in the pair x_{i-1}, $\overline{x}_{i-1}$
10. F plays u_{6i-5}
11. T plays u_{6i-6}
12. F plays u_{6i-7}
13. T plays u_{6i-8}
14. F plays u_{6i-9}
15. T plays u_{6i-10}
16. F plays u_{6i-11}

In the $G^+_{6,T\cdots F}$ game, T always assigns 1 and F always assigns 0 to variables. In a legitimate gameplay, T choosing x_i or $\overline{x}_i$ to assign 1 is like T choosing to assign $x_i = 1$ or $x_i = 0$ (respectively) in the 3-TQBF game. Similarly, F choosing x_{i-1} or $\overline{x}_{i-1}$ to assign 0 is like F choosing to assign $x_{i-1} = 0$ or $x_{i-1} = 1$ (respectively) in the 3-TQBF game.

We say the gameplay for the entire $G^+_{6,T\cdots F}$ game is legitimate when it consists of legitimate gameplay for the $(\exists x_{2n})(\forall x_{2n-1})$ round, followed by legitimate gameplay for the $(\exists x_{2n-2})(\forall x_{2n-3})$ round, followed by legitimate gameplay for the $(\exists x_{2n-4})(\forall x_{2n-5})$ round, and so on. Legitimate gameplay mimics the 3-TQBF gameplay in a natural way. We will design the clauses so that any player who plays illegitimately either outright loses, or at least gains no advantage by deviating from legitimate gameplay.

The 14 clauses associated with the 3-TQBF round $(\exists x_i)(\forall x_{i-1})$ are:

$$\begin{aligned}
A_i &= x_i \vee \overline{x}_i \vee u_{6i+1} \vee u_{6i+3} \vee u_{6i+5} \\
C_{6i} &= u_{6i} \vee u_{6i+1} \vee u_{6i+3} \vee u_{6i+5} \vee (x_i \wedge \overline{x}_i) \\
C_{6i-2} &= u_{6i-2} \vee u_{6i-1} \vee u_{6i+1} \vee u_{6i+3} \vee (x_i \wedge \overline{x}_i) \\
C_{6i-4} &= u_{6i-4} \vee u_{6i-3} \vee u_{6i-1} \vee u_{6i+1} \vee (x_i \wedge \overline{x}_i) \\
B_i &= x_{i-1} \vee \overline{x}_{i-1} \vee u_{6i-3} \vee u_{6i-1} \\
C_{6i-6} &= u_{6i-6} \vee u_{6i-5} \vee u_{6i-3} \vee u_{6i-1} \vee (x_{i-1} \wedge \overline{x}_{i-1}) \\
C_{6i-8} &= u_{6i-8} \vee u_{6i-7} \vee u_{6i-5} \vee u_{6i-3} \vee (x_{i-1} \wedge \overline{x}_{i-1}) \\
C_{6i-10} &= u_{6i-10} \vee u_{6i-9} \vee u_{6i-7} \vee u_{6i-5} \vee (x_{i-1} \wedge \overline{x}_{i-1})
\end{aligned}$$

As we note later, each C_j is not really a clause, since it contains a conjunction, but it is equivalent to a pair of clauses. Thus the six C_j's correspond to 12 clauses, but we often refer to C_j as "a clause" anyway. Note that each C_j contains one even-index u variable and the three previous odd-index u variables. For any clause that appears to contain some u_j variable where $j > 12n$, that non-existent variable is actually not present in the clause. Intuitively, the variables x_i and $\overline{x}_i$ in A_i, and x_{i-1} and $\overline{x}_{i-1}$ in B_i, and u_j in C_j (which we wrote first in the clauses) enable F to threaten T with defeat if T plays illegitimately, and the other variables in the clauses enable T to threaten F with defeat if F plays illegitimately.

For each clause F_k in the 3-TQBF game we introduce a clause

$$D_k = F'_k \vee u_1 \vee u_3 \vee u_5$$

where F'_k is the clause which results from replacing each negated variable $\neg x_i$ by the unnegated variable $\overline{x}_i$ throughout the clause F_k. For example, if $F_k = (x_1 \vee \neg x_2 \vee \neg x_3)$ then $F'_k = (x_1 \vee \overline{x}_2 \vee \overline{x}_3)$, where x_2, $\overline{x}_2$, x_3, $\overline{x}_3$ are separate variables.

In summary, the formal construction is as follows:

$$\begin{aligned}
X &= \{x_1, \overline{x}_1, x_2, \overline{x}_2, \ldots, x_{2n}, \overline{x}_{2n}\} \cup \{u_1, u_2, \ldots, u_{12n}\} \\
&= \bigcup_{i=2,4,6,\ldots,2n} \{x_i, \overline{x}_i, x_{i-1}, \overline{x}_{i-1}, u_{6i}, u_{6i-1}, \ldots, u_{6i-11}\} \\
\varphi^+ &= \bigwedge_{i=2,4,6,\ldots,2n} (A_i \wedge B_i) \wedge \bigwedge_{j=2,4,6,\ldots,12n} (C_j) \wedge \bigwedge_{k=1,2,3,\ldots,m} (D_k)
\end{aligned}$$

where

$$\begin{aligned}
A_i &= x_i \vee \overline{x}_i \vee u_{6i+1} \vee u_{6i+3} \vee u_{6i+5} \\
B_i &= x_{i-1} \vee \overline{x}_{i-1} \vee u_{6i-3} \vee u_{6i-1} \\
C_j &= u_j \vee u_{j+1} \vee u_{j+3} \vee u_{j+5} \vee (x_{\lceil j/6 \rceil} \wedge \overline{x}_{\lceil j/6 \rceil}) \\
D_k &= F'_k \vee u_1 \vee u_3 \vee u_5
\end{aligned}$$

Any occurrence of a non-existent variable u_j (where $j > 12n$) is omitted from the clauses. For example, A_{2n} is simply the clause $x_{2n} \vee \overline{x}_{2n}$. Now:

$$C_j = (u_j \vee u_{j+1} \vee u_{j+3} \vee u_{j+5} \vee x_{\lceil j/6 \rceil}) \wedge (u_j \vee u_{j+1} \vee u_{j+3} \vee u_{j+5} \vee \overline{x}_{\lceil j/6 \rceil})$$

So C_j contains two clauses with width $\leqslant 5$, and A_i, B_i, and D_k are individual clauses with widths $\leqslant 5$, $\leqslant 4$, and $\leqslant 6$ respectively. Therefore, φ^+ is a positive 6-CNF with $16n$ variables and $14n + m$ clauses. Though C_j contains two clauses we often treat C_j as a clause in the proof. The construction is now complete. Furthermore, (φ^+, X) can be constructed in logarithmic space.

Now we claim T has a winning strategy in the 3-TQBF game iff T has a winning strategy in the $\mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{F}}$ game (φ^+, X).

First we prove in Lemma 1 that the claim holds if the gameplay is restricted to be legitimate. Then we prove that the claim still holds even if the gameplay is not legitimate. In Lemma 2 we show if T plays illegitimately then either the game will be restored to a legitimate situation with no advantage to T, or F will win immediately. In Lemma 3 we show if F plays illegitimately then either the game will be restored to a legitimate situation with no advantage to F, or a chain reaction will be started that enables T to win eventually.

Lemma 1. T *has a winning strategy in the* 3-TQBF *game iff* T *has a winning strategy in the* $\mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{F}}$ *game* (φ^+, X) *when gameplay is restricted to be legitimate.*

Proof. A legitimate gameplay satisfies all A_i, B_i, C_j since A_i is satisfied by one of x_i or $\overline{x}_i$, B_i is satisfied by one of x_{i-1} or $\overline{x}_{i-1}$, and C_j is satisfied by u_j where j is even because they have been played by T. Since F plays all u_1, u_3, u_5 we know that D_k gets satisfied iff F'_k gets satisfied. Furthermore, F'_k gets satisfied iff F_k gets satisfied by the assignment to the x_i variables (ignoring the $\overline{x}_i$ variables), because of the definition of F'_k and the fact that x_i and $\overline{x}_i$ get opposite values. In summary, a legitimate gameplay satisfies φ^+ iff $F_1 \wedge F_2 \wedge \cdots \wedge F_m$ gets satisfied by the assignment to the x_i variables.

Suppose F has a winning strategy in the 3-TQBF game. We describe F's winning strategy in (φ^+, X). F can use the same strategy to pick one from $x_{i-1}, \overline{x}_{i-1}$ where F picking x_{i-1} or $\overline{x}_{i-1}$ is equivalent to assigning $x_{i-1} = 0$ or $x_{i-1} = 1$ respectively in the 3-TQBF game. F wins since this strategy makes the assignment to all the x_i variables match F's strategy in the 3-TQBF game, which ensures $F_1 \wedge \cdots \wedge F_m$ is unsatisfied and hence φ^+ is unsatisfied.

Suppose T has a winning strategy in the 3-TQBF game. We describe T's winning strategy in (φ^+, X). T can use the same strategy to pick one from $x_i, \overline{x}_i$ where T picking x_i or $\overline{x}_i$ is equivalent to assigning $x_i = 1$ or $x_i = 0$ in the 3-TQBF game respectively. T wins since this strategy makes the assignment to all the x_i variables match T's strategy in the 3-TQBF game, which ensures $F_1 \wedge \cdots \wedge F_m$ is satisfied and hence φ^+ is satisfied. ☐

Lemma 2. *If* F *has a winning strategy in the* 3-TQBF *game then* F *has a winning strategy in the* $\mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{F}}$ *game* (φ^+, X) *even if the gameplay does not progress legitimately.*

Proof. Suppose F has a winning strategy in the 3-TQBF game. In the game (φ^+, X), F can follow his strategy from Lemma 1 until T plays illegitimately on move p (p is odd and $1 \leqslant p \leqslant 16$) at round $(\exists x_i)(\forall x_{i-1})$. We consider all the different cases of p:

- $p = 1$: F already played u_{6i+1}, u_{6i+3}, u_{6i+5} (or these variables do not exist if $i = 2n$) due to legitimate gameplay before this move. T was supposed to play x_i or $\overline{x}_i$ but T did not do so. There are two possibilities:
  - If T also did not play u_{6i}, then F plays u_{6i}. Then whatever T plays, F plays one of x_i, $\overline{x}_i$. F wins since C_{6i} is unsatisfied.
  - If T played u_{6i}, then F plays one of x_i or $\overline{x}_i$ (it does not matter which one). Now it is T's move. If T plays the other from x_i, $\overline{x}_i$ then the game comes back to a legitimate situation at move 4, where F has no disadvantage since T effectively let F make the choice of x_i or $\overline{x}_i$ for him. If T does not play the other from x_i, $\overline{x}_i$ then F plays it and wins since A_i is unsatisfied.
- $p = 9$: F already played u_{6i-3}, u_{6i-1} and one of x_{i-1}, $\overline{x}_{i-1}$ due to legitimate gameplay before this move. T was supposed to play the other one from x_{i-1}, $\overline{x}_{i-1}$ but T did not do so. F plays it and wins since B_i is unsatisfied.
- Other p: T was supposed to play u_j where j is even, but T did not do so. F already played u_{j+1}, u_{j+3}, u_{j+5} and one of $x_{\lceil j/6 \rceil}$, $\overline{x}_{\lceil j/6 \rceil}$ due to legitimate gameplay before this move. Then F plays u_j and wins since C_j is unsatisfied. □

Lemma 3. *If* T *has a winning strategy in the* 3-TQBF *game then* T *has a winning strategy in the* $\mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{F}}$ *game* (φ^+, X) *even if the gameplay does not progress legitimately.*

Definition 1. *We define an* ***order on all the clauses****:* A_i, C_{6i}, C_{6i-2}, C_{6i-4}, B_i, C_{6i-6}, C_{6i-8}, C_{6i-10} *for* $i = 2n$ *then the same for* $i = 2n-2$*, and so on. Finally all* D_k *at the end ordered by* k *increasing. To represent an* ***interval of clauses*** *from this order, we use analogous mathematical notations "(", ")", "[", "]". For example,* $[A_{2n}, C_t)$ *means all the clauses from* A_{2n} *(inclusive) to* C_t *(exclusive). Let* V_t *be all the variables that occur at least once in* $(C_t, C_2]$ *along with* $\{u_1, u_3, u_5\}$*. For example,* $V_2 = \{u_1, u_3, u_5\}$ *and* $V_4 = \{u_1, u_2, u_3, u_5, u_7\}$*.*

Lemma 4. *If* $[A_{2n}, C_t]$ *are already satisfied where* $t \leqslant 12n-4$ *and* F *has already played at most one variable in* V_t*, then* T *has a strategy to satisfy* $(C_t, D_m]$ *even if it is* F*'s turn.*

Before proving Lemma 4, we use it to prove Lemma 3.

Proof of Lemma 3. Suppose T has a winning strategy in the 3-TQBF game. In the game (φ^+, X), T can follow his strategy from Lemma 1 until F plays illegitimately on move p (p is even and $1 \leqslant p \leqslant 16$) at round $(\exists x_i)(\forall x_{i-1})$. The outline of the argument is: The legitimate gameplay so far will have satisfied an interval of clauses, from A_{2n} through some clause associated with round $(\exists x_i)(\forall x_{i-1})$. After the illegitimate move by F, there might be another opportunity for F to restore the gameplay to a legitimate situation with no disadvantage to T. If that opportunity does not exist, or if F fails to get the gameplay "back on track," then T will have a move that satisfies the next few clauses. Then for some t (t stands for "threshold"), $[A_{2n}, C_t]$ will be satisfied, and it will be F's turn and T will satisfy the rest of the clauses (and hence win) by Lemma 4. The illegitimate move by F could have happened in V_t or somewhere else, and none of the other prior moves happened in V_t.

We consider all the different cases of p:

- $p = 2$: $[A_{2n}, C_{6i})$ are already satisfied due to legitimate gameplay before this move. F was supposed to play the other one from x_i, $\overline{x}_i$ but F did not do so. Then T plays that and that satisfies $[C_{6i}, C_{6i-4}]$. Now it is F's turn and T wins by Lemma 4 with $t = 6i - 4$.

- $p = 8$: $[A_{2n}, B_i)$ are already satisfied due to legitimate gameplay before this move. F was supposed to play one from x_{i-1}, $\overline{x}_{i-1}$ but F did not do so. There are two possibilities:

  - If F played u_{6i-5}, then T plays one of x_{i-1} or $\overline{x}_{i-1}$ (it does not matter which one). Now it is F's move. If F plays the other from x_{i-1}, $\overline{x}_{i-1}$ then the game comes back to a legitimate situation at move 11, where T has no disadvantage since F effectively let T make the choice of x_{i-1} or $\overline{x}_{i-1}$ for him. If F does not play the other from x_{i-1}, $\overline{x}_{i-1}$ then T plays it and that satisfies $[B_i, C_{6i-10}]$, so now it is F's turn and T wins by Lemma 4 with $t = 6i - 10$.
  - If F did not play u_{6i-5}, then T plays u_{6i-5} and that satisfies $[C_{6i-6}, C_{6i-10}]$. Let us pretend, for a moment, that one of x_{i-1} or $\overline{x}_{i-1}$ has already been played by T and the other has already been played by F (though in reality, neither has been played yet). Then B_i and hence all of $[A_{2n}, C_{6i-10}]$ are satisfied, and F's illegitimate move was the only variable that may have been played so far among V_{6i-10}, and it is F's turn, so T would win by Lemma 4 with $t = 6i - 10$. In reality, T can use that strategy from Lemma 4, and whenever F plays one of x_{i-1} or $\overline{x}_{i-1}$, T responds by playing the other, then resumes the strategy from Lemma 4. (Or, if F never plays x_{i-1} or $\overline{x}_{i-1}$, then T will play one of them after concluding his strategy from Lemma 4, and F will have to play the other as the final move.) Then B_i gets satisfied along with $(C_{6i-10}, D_m]$, so T wins.

- $p = 16$: $[A_{2n}, C_{6i-10}]$ are already satisfied due to legitimate gameplay before this move. F was supposed to play u_{6i-11} but F did not do so. Here $i > 2$ since if $i = 2$ then $u_{6i-11} = u_1$, which will be the only leftover variable to play and F must play it. So we only consider $i > 2$. Then T plays u_{6i-11} (which is $u_{6(i-2)+1}$) and that satisfies $[A_{i-2}, C_{6(i-2)-4}]$. Now it is F's turn and T wins by Lemma 4 with $t = 6(i-2) - 4$.

- Other p: F was supposed to play u_{j+1} (2nd variable in C_j and j is even) but F did not do so. $[A_{2n}, C_j)$ are already satisfied due to legitimate gameplay before this move. Then T plays u_{j+1}. There are two possibilities of j:

  - $j \leqslant 4$: T's move u_{j+1} satisfies $[C_j, D_m]$ since all D_k are satisfied by u_{j+1} (which is either u_3 or u_5). Therefore T wins.
  - $j > 4$: T's move u_{j+1} satisfies $[C_j, C_{j-4}]$. Now it is F's turn and T wins by Lemma 4 with $t = j - 4$. □

To prove Lemma 4, we need Lemma 5, which concerns "tree-like" positive 3-CNFs. Lemma 5 follows from [RW20b], but for completeness we provide a streamlined, self-contained proof in Section 2.2.

Definition 2. *A positive* 3*-CNF is a* ***tree*** *if each of the following holds:*

(1) Each clause has width exactly 3*, so the formula can be viewed as a* 3*-uniform hypergraph where variables are nodes and clauses are hyperedges.*
(2) Each clause has at least one "spare variable" that occurs in no other clauses.
(3) Any two clauses share at most one variable.
*(4) If we delete a spare variable from every clause, the resulting graph (*2*-uniform hypergraph) would be a tree (i.e., connected and no cycles).*

When we say F can use pass moves, this means F has the option of forgoing any turn, thus forcing T to play multiple variables in a row.

Lemma 5. *For every tree,* T *has a winning strategy even if* F *gets to play the first two moves and* F *can use pass moves.*

Proof of Lemma 4. Shrink the clauses $(C_t, D_m]$ by removing some variables from them as follows:

$$
\begin{aligned}
A'_i &= x_i \vee \overline{x}_i \vee u_{6i+3} \\
B'_i &= x_{i-1} \vee \overline{x}_{i-1} \vee u_{6i-3} \\
C'_j &= u_j \vee u_{j+3} \vee u_{j+5} && \text{(previously two clauses, now only one)} \\
D' &= u_1 \vee u_3 \vee u_5 && \text{(all } D'_k \text{ are the same, we call it just } D'\text{)}
\end{aligned}
$$

All these clauses form a positive 3-CNF ψ. The hypergraph for ψ has been illustrated in Figure 1. We argue that ψ is a tree. We show it satisfies each of the four properties of a tree as described in Definition 2.

- Tree property (1) holds since each of A'_i, B'_i, C'_j, D' has exactly 3 variables. The variables u_{6i+3} in A'_i, and u_{j+3} and u_{j+5} in C'_j, are guaranteed to exist since $t \leqslant 12n-4$.
- Tree property (2) holds since x_i, x_{i-1}, u_j, u_1 only occur in A'_i, B'_i, C'_j, D' respectively.
- Tree property (3) holds since:
  - C'_j and A'_i share only u_{6i+3} if $j = 6i$ or $j = 6i-2$.
  - C'_j and B'_i share only u_{6i-3} if $j = 6i-6$ or $j = 6i-8$.
  - C'_j and C'_{j-2} share only u_{j+3}.
  - C'_2 and D' share only u_5.
  - Other pairs do not share a variable.
- Tree property (4) holds since deleting x_i, x_{i-1}, u_j, u_1 (which are spare variables) from A'_i, B'_i, C'_j, D' respectively creates a 2-uniform hypergraph as shown in Figure 2 which is clearly a tree.

Therefore ψ is a tree.

By Lemma 5, T has a winning strategy on the tree ψ even if F has the first two moves (and subsequently T and F play alternately) and F can use pass moves. Now we claim that T has a strategy to satisfy $(C_t, D_m]$ in φ^+ assuming F has already played at most one variable in V_t and it is F's turn (and F cannot use pass moves). Because every variable in ψ is also in V_t, we can say F has already played at most one variable of ψ. Because it is F's turn in φ^+, that's like allowing F to have the second move in ψ as well. After that, T's strategy for φ^+ is the same as T's winning

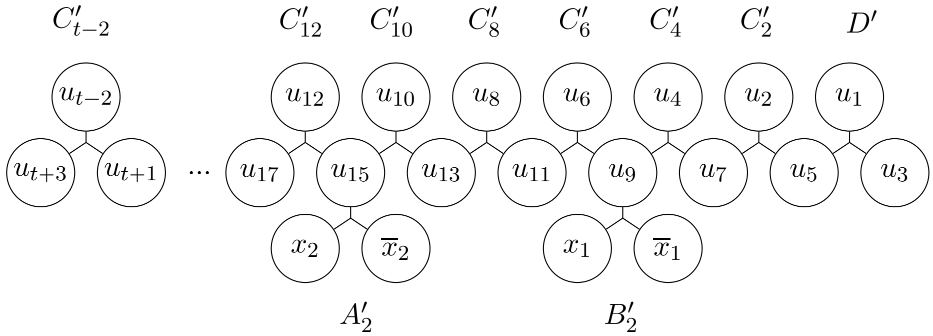

Figure 1: Hypergraph for ψ

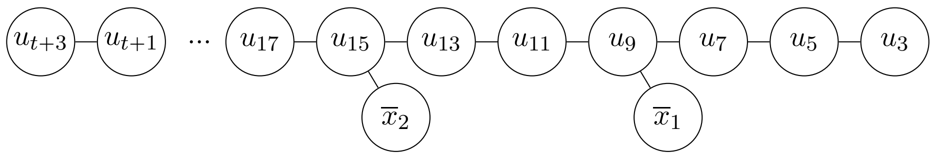

Figure 2: Hypergraph after deleting a spare variable from each clause in ψ

strategy for ψ, except that whenever F plays a variable of φ^+ that's not in ψ, T interprets it as a pass move by F and continues with his strategy for ψ. Since this strategy ensures that ψ gets satisfied, it also ensures that $(C_t, D_m]$ and hence all of φ^+ gets satisfied. $\square$

2.2 Trees

In order to prove Lemma 5, we need Lemma 6 and Lemma 7. First we outline some definitions.

Definition 3. *We henceforth refer to a tree as a* ***single tree****. A* ***married tree*** *is a formula consisting of two disjoint single trees ("spouses") and a width-2 clause with one endpoint in each spouse (and every width-3 clause has a spare variable even after the inclusion of the width-2 clause). The endpoints of the width-2 clause in a married tree are considered roots of the spouses. A* ***win-forest*** *is a formula where each connected component is either a single tree or a married tree.*

After any move by T or F, a formula changes to a residual formula where the variable that got played is removed, and if T played then any clause containing the variable disappears (since it is satisfied), and if F played then any clause containing the variable shrinks (since a false literal might as well not be there).

Lemma 6. *Any move by* F *on a single tree results in a win-forest.*

Lemma 7. T *can ensure that a win-forest remains a win-forest after an* F-T *round even if* F *can use pass moves.*

Before proving Lemma 6 and Lemma 7, we use them to prove Lemma 5.

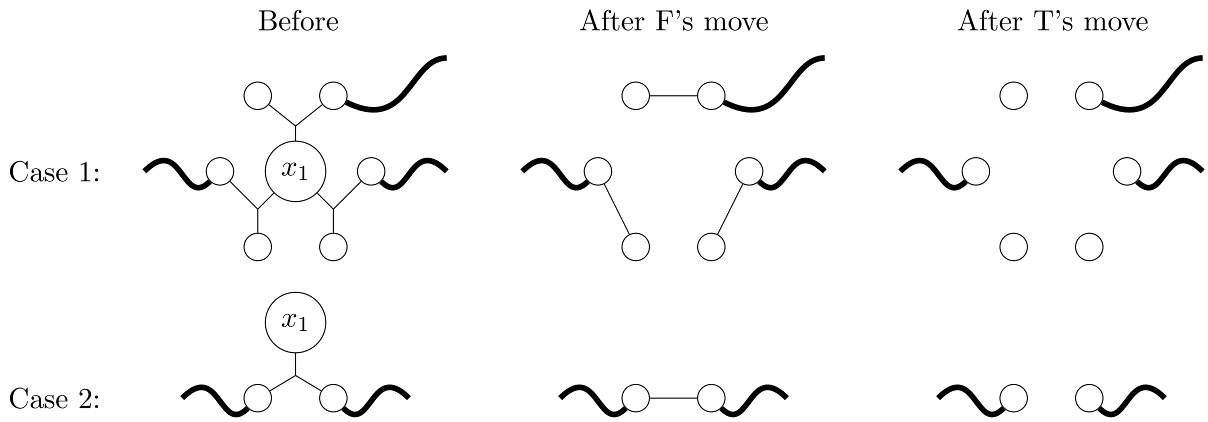

Figure 3: F's move and T's move on x_1 and its effect on formulas

Proof of Lemma 5. The tree ψ is a single tree. By Lemma 6, F's first move on ψ results in a win-forest. Then we prove T can win a $\mathrm{G}^{+}_{3,\mathrm{F}\ldots}$ game on that win-forest even if F can use pass moves. We prove this by induction on the number of variables.

Base case: The formula is a win-forest with one or two variables. In case of one variable the only possibility is an isolated variable with no clauses. T has already won in this case. In case of two variables there exists either two isolated variables where T has already won or a width-2 clause which T can satisfy in one move.

Induction step: The formula is a win-forest with at least three variables. Whatever F plays, T has a response to ensure the residual formula is again a win-forest by Lemma 7. By the induction hypothesis, T can win the rest of the game. □

Any move by T or F can occur in two different ways as illustrated in Figure 3. Specifically, Case 1 is a move on a non-spare variable, and Case 2 is a move on a spare variable.

Proof of Lemma 6. The formula is a single tree. If F's move is a pass move then that results in a win-forest with only one single tree. If F's move is an actual move then it creates some married trees in which one spouse is just a single variable (Case 1 with F) or only one married tree (Case 2 with F). Then that results in a win-forest with only married trees. □

Proof of Lemma 7. The argument will show that whatever F plays, whether a pass move or an actual move in a single tree or married tree, T has a response such that each component of the residual formula is again either a single tree or a married tree; therefore the residual formula is again a win-forest.

Suppose F played a pass move. T can play any remaining variable in the win-forest. If that variable is an isolated variable then it just removes the isolated variable. Otherwise it satisfies some clauses in a component by Case 1 or Case 2 with T. Consequently the component is broken down into some single trees and possibly one married tree (if the component was a married tree). This preserves the win-forest property.

Suppose F played in a single tree. Then by Lemma 6 the residual formula is a win-forest. Then T can pretend F just played a pass move on this win-forest, and T can respond as explained in the previous paragraph. This preserves the win-forest property.

Suppose F played in a married tree. F's move happened in one of the two single trees that got married. T can play the root of the other spouse (where F has not played) and satisfy the width-2 clause. This means the two single trees get separated by T's move and it also breaks T's single tree at the root by Case 1 with T. Furthermore, F's move in his single tree also preserves the win-forest property by Lemma 6. This preserves the win-forest property. ☐

2.3 Corollaries

In this section, we investigate corollaries for G_6^+ in Section 2.3.1, G_6 in Section 2.3.2, G_5^+ in Section 2.3.3, and G_5 in Section 2.3.4.

2.3.1 G_6^+

Our proof of Theorem 1 in Section 2.1 showed that $\mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{F}}$ is PSPACE-complete. Now we show that $\mathrm{G}^+_{6,\mathrm{F}\cdots\mathrm{F}}$, $\mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{T}}$, and $\mathrm{G}^+_{6,\mathrm{F}\cdots\mathrm{T}}$ are also PSPACE-complete.

Corollary 1. $\mathrm{G}^+_{6,\mathrm{F}\cdots\mathrm{F}}$ *is* PSPACE*-complete.*

Proof. The reduction is 3-TQBF $\leqslant \mathrm{G}^+_{6,\mathrm{F}\cdots\mathrm{F}}$. The idea is similar to 3-TQBF $\leqslant \mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{F}}$ from the proof of Theorem 1 in Section 2.1. We introduce one more variable z to X and add z to the first four clauses of φ^+: A_{2n}, C_{12n}, C_{12n-2}, and C_{12n-4}, increasing their widths by one, from $2, 2, 3, 4$ to $3, 3, 4, 5$ respectively. So φ^+ is a 6-CNF.

Now the claim is that T has a winning strategy in the 3-TQBF game iff T has a winning strategy in the $\mathrm{G}^+_{6,\mathrm{F}\cdots\mathrm{F}}$ game (φ^+, X).

Suppose F has a winning strategy in the 3-TQBF game. Then F can play z as the first move. Then F wins by the same argument as in Section 2.1.

Suppose T has a winning strategy in the 3-TQBF game. If F plays z as the first move then T wins by the same argument as in Section 2.1. If F does not play z as the first move then T plays z and satisfies A_{2n}, C_{12n}, C_{12n-2}, and C_{12n-4}. Then T wins by Lemma 4 with $t = 12n - 4$. ☐

Corollary 2. $\mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{T}}$ *is* PSPACE*-complete.*

Proof. The reduction is $\mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{F}} \leqslant \mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{T}}$. Suppose an instance of $\mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{F}}$ is (φ^+, X). We simply introduce a dummy variable z that does not appear in φ^+ and use $Y = X \cup \{z\}$. We claim that T has a winning strategy in the $\mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{F}}$ game (φ^+, X) iff T has a winning strategy in the $\mathrm{G}^+_{6,\mathrm{T}\cdots\mathrm{T}}$ game (φ^+, Y). We repeat an argument from [RW20a] that shows this.

Suppose T has a winning strategy on (φ^+, X). We show T's winning strategy on (φ^+, Y). T can start by the same strategy as in (φ^+, X) and continue as long as F does not play z. If F never plays z, then T plays z at the end and wins as in (φ^+, X). If F plays z then T can respond by playing any remaining variable $x_i = 1$, then T resumes his strategy from (φ^+, X) until that strategy tells him to play x_i. At this time, T again picks any other remaining variable and assigns it 1. Then T again resumes his strategy from (φ^+, X). The game goes on like this in phases. At the end, T has played all the variables he would have played in the (φ^+, X) game and possibly one more. Since φ^+ is positive, it must still be satisfied when one of the variables is 1 instead of 0.

Suppose F has a winning strategy on (φ^+, X). Then F's winning strategy on (φ^+, Y) is analogous to T's strategy in the previous paragraph. ☐

Corollary 3. $G^+_{6,F\cdots T}$ *is* PSPACE*-complete.*

Proof. $G^+_{6,F\cdots F}$ is PSPACE-complete by Corollary 1. The reduction is $G^+_{6,F\cdots F} \leqslant G^+_{6,F\cdots T}$. The technique is identical to Corollary 2. ☐

Therefore we found PSPACE-completeness of all patterns of G^+_6 games.

Corollary 4. $G^+_{6,T\cdots F}, G^+_{6,F\cdots F}, G^+_{6,T\cdots T}, G^+_{6,F\cdots T}$ *remain* PSPACE*-complete even when every clause has exactly* 6 *variables.*

Proof. For any pattern $a \cdots b$ where $a, b \in \{T, F\}$, we reduce from $G^+_{6,a\cdots b}$ to the restricted version where every clause has exactly 6 variables. We argue that any clause C with width < 6 can be resized to a set of width-6 clauses without changing the outcome. We introduce two variables x, x' and clause C is written as $(C \vee x) \wedge (C \vee x')$, thus increasing C's width by 1. Whichever player has a winning strategy in the original formula, they can follow the same strategy in the modified formula until the other player plays x or x' and then respond by playing the other. (Or, if the other player never plays x or x', then it does not matter which one the winning player plays as the 2nd-to-last move in the game.) So it is possible to increase any clause's width without changing the outcome. We can repeatedly do this process until all clauses have width exactly 6. This increases the size of the formula by at most a constant factor. ☐

2.3.2 G_6

We already know that $G_{5,T\cdots F}$ and $G_{5,F\cdots F}$ are PSPACE-complete [RW20a]. But any completeness result for $G_{5,T\cdots T}$ and $G_{5,F\cdots T}$ is unknown. Not only that, but also the complexities of $G_{6,T\cdots T}$ and $G_{6,F\cdots T}$ were unknown. Due to Corollary 2 and Corollary 3 we now know that $G_{6,T\cdots T}$ and $G_{6,F\cdots T}$ are also PSPACE-complete.

2.3.3 G^+_5

Now we show that $G^+_{5,T\cdots F}$, $G^+_{5,F\cdots F}$, $G^+_{5,T\cdots T}$, and $G^+_{5,F\cdots T}$ are all NL-hard. Each of these results implies Theorem 2.

Corollary 5. $G^+_{5,T\cdots F}$ *is* NL*-hard.*

Proof. It is well-known that 2-SAT is NL-complete, and trivially 2-SAT $\leqslant$ 2-TQBF. The reduction is 2-TQBF $\leqslant G^+_{5,T\cdots F}$. The technique is identical to 3-TQBF $\leqslant G^+_{6,T\cdots F}$ in Theorem 1 where the widths of A_i, B_i, C_j, D_k were $5, 4, 5, 6$ respectively. Since each F_k is now a width-2 clause, D_k becomes a width-5 clause. Therefore φ^+ becomes a 5-CNF. ☐

Corollary 6. $G^+_{5,F\cdots F}$ *is* NL*-hard.*

Proof. The reduction is 2-TQBF $\leqslant G^+_{5,F\cdots F}$. The technique is identical to Corollary 1. ☐

Corollary 7. $G^+_{5,T\cdots T}$ *is* NL*-hard.*

Proof. $G^+_{5,T\cdots F}$ is NL-hard by Corollary 5. The reduction is $G^+_{5,T\cdots F} \leqslant G^+_{5,T\cdots T}$. The technique is identical to Corollary 2. ☐

Corollary 8. $\mathrm{G}^+_{5,\mathrm{F}\cdots\mathrm{T}}$ *is* NL*-hard.*

Proof. $\mathrm{G}^+_{5,\mathrm{F}\cdots\mathrm{F}}$ is NL-hard by Corollary 6. The reduction is $\mathrm{G}^+_{5,\mathrm{F}\cdots\mathrm{F}} \leqslant \mathrm{G}^+_{5,\mathrm{F}\cdots\mathrm{T}}$. The technique is identical to Corollary 2. □

Therefore we found NL-hardness of all patterns of G^+_5 games. But any completeness result for any pattern still remains open.

Corollary 9. $\mathrm{G}^+_{5,\mathrm{T}\cdots\mathrm{F}}, \mathrm{G}^+_{5,\mathrm{F}\cdots\mathrm{F}}, \mathrm{G}^+_{5,\mathrm{T}\cdots\mathrm{T}}, \mathrm{G}^+_{5,\mathrm{F}\cdots\mathrm{T}}$ *remain* NL*-hard even when every clause has exactly* 5 *variables.*

Proof. The technique is identical to Corollary 4. □

2.3.4 G_5

We already know that $\mathrm{G}_{5,\mathrm{T}\cdots\mathrm{F}}$ and $\mathrm{G}_{5,\mathrm{F}\cdots\mathrm{F}}$ are PSPACE-complete [RW20a]. But nothing was known for $\mathrm{G}_{5,\mathrm{T}\cdots\mathrm{T}}$ and $\mathrm{G}_{5,\mathrm{F}\cdots\mathrm{T}}$. Due to Corollary 7 and Corollary 8 we now know that $\mathrm{G}_{5,\mathrm{T}\cdots\mathrm{T}}$ and $\mathrm{G}_{5,\mathrm{F}\cdots\mathrm{T}}$ are also NL-hard. But any completeness result for $\mathrm{G}_{5,\mathrm{T}\cdots\mathrm{T}}$ and $\mathrm{G}_{5,\mathrm{F}\cdots\mathrm{T}}$ still remains open.

3 Proof of Theorem 3

In this section, we show 2-SAT $\leqslant \mathrm{G}_{4,\mathrm{T}\cdots\mathrm{F}}$ and 2-SAT $\leqslant \mathrm{G}_{4,\mathrm{F}\cdots\mathrm{F}}$, each of which implies Theorem 3.

Lemma 8. $\mathrm{G}_{4,\mathrm{T}\cdots\mathrm{F}}$ *is* NL*-hard.*

Proof. 2-SAT is a well-known NL-complete problem. We show 2-SAT $\leqslant \mathrm{G}_{4,\mathrm{T}\cdots\mathrm{F}}$ under a logarithmic space reduction. Suppose an instance of 2-SAT is (φ, X) where φ is a 2-CNF and X is the set of boolean variables that occur in φ. We construct an instance of $\mathrm{G}_{4,\mathrm{T}\cdots\mathrm{F}}$ as $(\psi, X \cup Y)$ where ψ is a 4-CNF and $X \cup Y$ is the set of boolean variables that occur in ψ. The reduction will show that φ has a satisfying assignment iff T has a winning strategy in the $\mathrm{G}_{4,\mathrm{T}\cdots\mathrm{F}}$ game $(\psi, X \cup Y)$. The construction is as follows:

Suppose $X = \{x_1, x_2, \ldots, x_n\}$ and let $Y = \{y_1, y_2, \ldots, y_n\}$. To construct ψ, take φ and replace each occurrence of x_i with $(x_i \vee y_i)$ and replace each occurrence of $\neg x_i$ with $(\neg x_i \vee \neg y_i)$.

Suppose φ does not have a satisfying assignment. We show a winning strategy for F in $(\psi, X \cup Y)$. The strategy is whenever T plays in a fresh pair x_i, y_i then F can immediately play the other variable from the pair to make $x_i = y_i$. The strategy works since making $x_i = y_i$ for all i makes ψ equivalent to φ where T gets to assign all variables. Since φ does not have a satisfying assignment, F wins.

Suppose φ has a satisfying assignment and fix one such assignment. We show a winning strategy for T in $(\psi, X \cup Y)$. T starts by picking a fresh pair x_i, y_i and assigns x_i according to x_i's value in φ's satisfying assignment. If F immediately replies with y_i then T picks another fresh pair and so on. If F does not play y_i but in some fresh pair x_j, y_j then T immediately plays the other variable from the pair x_j, y_j according to x_j's value in φ's satisfying assignment. T keeps chasing F like this until F plays y_i. After F eventually plays y_i, T continues by playing x_k in any other fresh pair x_k, y_k and chasing F until F plays y_k. The strategy works since T is able to assign a variable from each pair x_i, y_i according to the satisfying assignment in φ. Therefore T wins since ψ gets satisfied. □

Lemma 9. $\mathrm{G}_{4,\mathrm{F}\cdots\mathrm{F}}$ *is* NL-*hard.*

Proof. We show 2-SAT $\leqslant \mathrm{G}_{4,\mathrm{F}\cdots\mathrm{F}}$. The argument is almost identical to Lemma 8 except some minor changes that need to be explicitly addressed. We introduce a dummy variable d to have $X \cup Y \cup \{d\}$ and no changes to ψ in Lemma 8's construction. The idea is to make F play that d to ultimately get $\mathrm{G}_{4,\mathrm{T}\cdots\mathrm{F}}$ then we will be done with the rest. We claim φ has a satisfying assignment iff T has a winning strategy in the $\mathrm{G}_{4,\mathrm{F}\cdots\mathrm{F}}$ game $(\psi, X \cup Y \cup \{d\})$.

Suppose φ does not have a satisfying assignment. We argue that F has a winning strategy. F can start by playing d. Then we are left with a $\mathrm{G}_{4,\mathrm{T}\cdots\mathrm{F}}$ game where F wins by the same argument as in Lemma 8.

Suppose φ has a satisfying assignment and fix one such assignment. We argue that T has a winning strategy. If F plays d as the first move then the rest of the strategy is identical to Lemma 8 and T wins. If F does not play d at the beginning but plays in a fresh pair x_i, y_i then T can immediately respond by playing the other variable and assign it according to x_i's value in φ's satisfying assignment. T can chase F like this until F plays d. This way T can play exactly one variable from each pair until d is played. After d is played by F, the game remains as $\mathrm{G}_{4,\mathrm{T}\cdots\mathrm{F}}$ where some rounds already happened as if d never existed at all. The same argument works and T wins. ☐

The reductions in the proofs of Lemma 8 and Lemma 9 produce 4-CNFs where every clause has exactly 4 literals, so $\mathrm{G}_{4,\mathrm{T}\cdots\mathrm{F}}$ and $\mathrm{G}_{4,\mathrm{F}\cdots\mathrm{F}}$ remain NL-hard under this restriction. It remains open to show that $\mathrm{G}_{4,\mathrm{T}\cdots\mathrm{T}}$, $\mathrm{G}_{4,\mathrm{F}\cdots\mathrm{T}}$, and G_4^+ are NL-hard.

4 Summary

In Table 2 we summarize the status of the complexity of G_w^+ for all widths w and all patterns. We conjecture that G_3^+ may be tractable, but the only known general upper bound is PSPACE. For G_5^+, it would be interesting to improve the NL-hardness to P-hardness. For G_4^+, any nontrivial result would be interesting (such as NL-hardness, or improving the PSPACE upper bound even under restrictions on the formula).

In Table 3 we summarize the status of the complexity of G_w for all widths w and all patterns. We conjecture that even G_3 might be tractable, but again the only known general upper bound is PSPACE. For $\mathrm{G}_{4,\mathrm{T}\cdots\mathrm{F}}$, $\mathrm{G}_{4,\mathrm{F}\cdots\mathrm{F}}$, $\mathrm{G}_{5,\mathrm{T}\cdots\mathrm{T}}$, and $\mathrm{G}_{5,\mathrm{F}\cdots\mathrm{T}}$, it would be interesting to improve the NL-hardness to P-hardness. For $\mathrm{G}_{4,\mathrm{T}\cdots\mathrm{T}}$ and $\mathrm{G}_{4,\mathrm{F}\cdots\mathrm{T}}$, any nontrivial result would be interesting.

It would also be interesting to see if Theorem 1 can be used to improve any parameters in some of the many PSPACE-completeness results that have been shown by reduction from Schaefer's theorem for width 11.

Acknowledgments

We thank Florian Galliot for useful exchanges. This work was supported by NSF grant CCF-1657377.

<table>
<tr><th>$w \rightarrow$</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th></tr>
<tr><td>T · · · F</td><td rowspan="4">L
[RW20a]</td><td rowspan="4">L
under restrictions
[Kut05]</td><td rowspan="4">Unknown</td><td>NL-hard
[Corollary 5]</td><td>PSPACE-complete
[Theorem 1]</td></tr>
<tr><td>F · · · F</td><td>NL-hard
[Corollary 6]</td><td>PSPACE-complete
[Corollary 1]</td></tr>
<tr><td>T · · · T</td><td>NL-hard
[Corollary 7]</td><td>PSPACE-complete
[Corollary 2]</td></tr>
<tr><td>F · · · T</td><td>NL-hard
[Corollary 8]</td><td>PSPACE-complete
[Corollary 3]</td></tr>
</table>

Table 2: G_w^+ results

<table>
<tr><th>$w \rightarrow$</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th></tr>
<tr><td>T · · · F</td><td rowspan="4">L
[RW20a]</td><td rowspan="4">L
under restrictions
[RW20b]</td><td>NL-hard
[Lemma 8]</td><td rowspan="2">PSPACE-complete
[RW20a]</td><td rowspan="2"></td></tr>
<tr><td>F · · · F</td><td>NL-hard
[Lemma 9]</td></tr>
<tr><td>T · · · T</td><td rowspan="2">Unknown</td><td>NL-hard
[Corollary 7]</td><td>PSPACE-complete
[Corollary 2]</td></tr>
<tr><td>F · · · T</td><td>NL-hard
[Corollary 8]</td><td>PSPACE-complete
[Corollary 3]</td></tr>
</table>

Table 3: G_w results

References

[APT79] Bengt Aspvall, Michael Plass, and Robert Tarjan. A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Information Processing Letters*, 8(3):121–123, 1979.

[AS03] Argimiro Arratia and Iain Stewart. A note on first-order projections and games. *Theoretical Computer Science*, 290(3):2085–2093, 2003.

[BDK+16] Boštjan Brešar, Paul Dorbec, Sandi Klavžar, Gašper Košmrlj, and Gabriel Renault. Complexity of the game domination problem. *Theoretical Computer Science*, 648:1–7, 2016.

[Bys04] Jesper Byskov. Maker-Maker and Maker-Breaker games are PSPACE-complete. Technical Report RS-04-14, BRICS, Department of Computer Science, Aarhus University, 2004.

[CE78] Vasek Chvátal and Paul Erdös. Biased positional games. *Annals of Discrete Mathematics*, 2:221–229, 1978.

[CPSS19] Eurinardo Costa, Victor Lage Pessoa, Rudini Menezes Sampaio, and Ronan Soares. PSPACE-hardness of two graph coloring games. In *Proceedings of the 10th Latin and American Algorithms, Graphs, and Optimization Symposium (LAGOS)*, pages 333–344. Elsevier, 2019.

[DGPR18] Éric Duchêne, Valentin Gledel, Aline Parreau, and Gabriel Renault. Maker-Breaker domination game. Technical Report 1807.09479, arXiv, 2018.

[DH08] Erik Demaine and Robert Hearn. Constraint logic: A uniform framework for modeling computation as games. In *Proceedings of the 23rd Conference on Computational Complexity (CCC)*, pages 149–162. IEEE, 2008.

[ES73] Paul Erdös and John Selfridge. On a combinatorial game. *Journal of Combinatorial Theory, Series A*, 14(3), 1973.

[FG87] Aviezri Fraenkel and Elisheva Goldschmidt. PSPACE-hardness of some combinatorial games. *Journal of Combinatorial Theory, Series A*, 46(1):21–38, 1987.

[FGM+15] Stephen Fenner, Daniel Grier, Jochen Messner, Luke Schaeffer, and Thomas Thierauf. Game values and computational complexity: An analysis via black-white combinatorial games. In *Proceedings of the 26th International Symposium on Algorithms and Computation (ISAAC)*, pages 689–699. Springer, 2015.

[GHIK19] Valentin Gledel, Michael Henning, Vesna Iršič, and Sandi Klavžar. Maker-Breaker total domination game. Technical Report 1902.00204, arXiv, 2019.

[Hea09] Robert Hearn. Amazons, Konane, and Cross Purposes are PSPACE-complete. In *Games of No Chance 3*, Mathematical Sciences Research Institute Publications, pages 287–306. Cambridge University Press, 2009.

[HKSS14] Dan Hefetz, Michael Krivelevich, Miloš Stojaković, and Tibor Szabó. *Positional Games*. Birkhäuser Basel (Springer), 2014.

[Kut05] Martin Kutz. Weak positional games on hypergraphs of rank three. In *Proceedings of the 3rd European Conference on Combinatorics, Graph Theory, and Applications (EuroComb)*, pages 31–36. Discrete Mathematics & Theoretical Computer Science, 2005.

[Leh64] Alfred Lehman. A solution of the Shannon switching game. *Journal of the Society for Industrial and Applied Mathematics*, 12(4):687–725, 1964.

[RW20a] Md Lutfar Rahman and Thomas Watson. Complexity of unordered CNF games. *ACM Transactions on Computation Theory*, 12(3):18:1–18:18, 2020.

[RW20b] Md Lutfar Rahman and Thomas Watson. Tractable unordered 3-CNF games. In *Proceedings of the 14th Latin American Theoretical Informatics Symposium (LATIN)*. Springer, 2020. To appear.

[Sch76] Thomas Schaefer. Complexity of decision problems based on finite two-person perfect-information games. In *Proceedings of the 8th Symposium on Theory of Computing (STOC)*, pages 41–49. ACM, 1976.

[Sch78] Thomas Schaefer. On the complexity of some two-person perfect-information games. *Journal of Computer and System Sciences*, 16(2):185–225, 1978.

[Sla00] Wolfgang Slany. The complexity of graph Ramsey games. In *Proceedings of the 2nd International Conference on Computers and Games (CG)*, pages 186–203. Springer, 2000.

[Sla02] Wolfgang Slany. Endgame problems of Sim-like graph Ramsey avoidance games are PSPACE-complete. *Theoretical Computer Science*, 289(1):829–843, 2002.

[SM73] Larry Stockmeyer and Albert Meyer. Word problems requiring exponential time. In *Proceedings of the 5th Symposium on Theory of Computing (STOC)*, pages 1–9. ACM, 1973.

[TDU11] Sachio Teramoto, Erik Demaine, and Ryuhei Uehara. The Voronoi game on graphs and its complexity. *Journal of Graph Algorithms and Applications*, 15(4):485–501, 2011.

[vV13] Jan van Rijn and Jonathan Vis. Complexity and retrograde analysis of the game Dou Shou Qi. In *Proceedings of the 25th Benelux Conference on Artificial Intelligence (BNAIC)*, 2013.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il