CMSC 858M: Fun with Hardness
Spring 2021

Instructor: Mohammad T. Hajiaghayi
Scribe: Erika Melder

April 30, 2021

1 Overview

In this chapter, we analyze Viglietta's results on games. First, we define the class PSPACE and see how it compares to other classes. We define the problem QSAT and determine that certain variants are PSPACE-hard. We then describe three metatheorems for proving hardness results. We use these metatheorems to analyze various popular games, and determine the complexity of solving arbitrarily arranged versions of those games. Finally, we define stochastic games and apply these metatheorems to various formulations of that model.

2 The Class PSPACE

Define the class PSPACE as follows:

Definition 1 *A problem* A *is in PSPACE if* A *can be solved in polynomial space.*

We know NP $\subseteq$ PSPACE $\subseteq$ EXP because every branching path has polynomially-many polynomially-sized stack frames, and there are exponentially-many total states in a polynomial-space machine. We also know PSPACE $=$ NPSPACE by Savitch's theorem - there is quadratic blowup when moving from a nondeterministic polyspace algorithm to a deterministic polyspace one.

3 QSAT and the QSAT Dichotomy

The problem QSAT (**Q**uantified **SAT**isfiability) is as follows:

Definition 2 *Consider a fully quantified Boolean formula* φ*, as in*

$$\varphi = Q_1 Q_2 \dots Q_n \psi(x_1, x_2, \dots, x_n)$$

where the Q_i *are alternating* $\forall/\exists$ *and* ψ *is a Boolean formula on* n *variables, and each variable has exactly one quantifier associated with it.*

Then φ *is in QSAT if it is satisfiable.*

We can refine this further by defining QkSAT (as in Q2SAT, Q3SAT, etc) to be QSAT on a formula with k variables and quantifiers. Additionally, similar to how Schaefer's Dichotomy Theorem classifies the hardness of various forms of 3SAT, there is a theorem for various forms of QSAT as well:

Theorem 1 *Horn-QSAT, Dual Horn-QSAT, Q2SAT, and X(N)OR-QSAT are all in P. All other forms of QSAT are PSPACE-complete.*

This extends to the planar versions - Planar 1-in-3 QSAT is hard while Planar NAE Q3SAT is easy.

4 Metatheorem 3 and its Consequences

Metatheorem 3 is as follows:

Theorem 2 *Define a* door *of a planar graph to be an edge that exists only if a specific Boolean variable is set to True.*
Define a pressure plate *to be a vertex that, when visited, will change a single Boolean variable to True or False (fixed on a per-pressure plate basis).*
A game where the player must traverse a planar graph from start to finish, that has door and pressure plate objects, is PSPACE-complete.

The reduction is from Q3SAT, and involves setting up a path that allows fixing variables quantified by $\exists$ while branching on variables quantified by $\forall$. This structure allows us to prove that many video games are PSPACE-complete, by describing how to build these door and pressure plate objects in their engines.

- **FPS Games:** Generally easy, since these games natively contain doors and pressure plates. (Quake is one such game.)
- **RPG Games:** Also generally easy, since the games contain the native structures.
- **SCUMM Engine Games:** These games rely on a similar logical structure, too.
- **Prince of Persia:** By placing pressure plates on the top of a wall at the height of the player, we can enforce that they are pressed even with the ability to jump, and prove the game PSPACE-complete.

5 Metatheorem 4 and its Consequences

Metatheorem 4 is as follows:

Theorem 3 *Define a* button *of a planar graph to be an optional event that may occur at a vertex that allows 3 doors to be opened (or closed) at once.*
A game where the player must traverse a planar graph from start to finish, that has door and button objects, is PSPACE-complete.

This can be obtained by treating a button as 3 pressure plates put together. This causes several more games to become PSPACE-complete, such as Sonic the Hedgehog.

6 Metatheorem 5 and its Consequences

Metatheorem 5 is a further generalization as follows:

Theorem 4 *Define a* generalized door *to include three distinct paths: a* traverse path *that requires the door to be opened, a* open path *which allows (but does not require) the player to open the door, and a* close path *which forces the player to close the door.*
A game where the player must traverse a planar graph from start to finish, that has generalized door and crossover objects, is PSPACE-complete.

This was notably used in the Nintendo Paper by Aloupis, Demaine, and Guo to prove NP-hardness of various games [1]. They introduce these structures in games such as The Legend of Zelda: A Link to the Past, all three Donkey Kong Country games, Super Mario Bros, and Lemmings, which proves that these games are PSPACE-complete. The structures all rely on a way to impede the player's progress that can be manipulated via the paths specified.

7 Stochastic Games

A stochastic game is a standard 2-player game in which one of the players is simulated by a fair coin. The question of asking whether a player can force a win with probability larger than $\frac{1}{2}$ in such a game is PSPACE-complete. We describe this in terms of the following problem:

Definition 3 *Consider a Boolean formula* φ *of the form*

$$\varphi = (\exists x_1)(Rx_2)(\exists x_3)\ldots[Pr(Win) > \frac{1}{2}]$$

where the R_i *are variables chosen at random.*

Such a formula is in the set Stochastic SAT *if it is satisfiable.*

A related formulation allows us to analyze both 0 and 2-player games of various types.

7.1 Deterministic Constraint Logic

Deterministic Constraint Logic (DCL) is a 0-player game (a *simulation*) with an unbounded number of moves. We are given a directed graph, and a set of rules determining how edges may reverse in this graph at each timestep. This is deterministic - it requires no input at all from a player. The problem is clearly in PSPACE because each timestep has a polynomial-space representation. Showing that it is PSPACE-hard relies on building a graph that is able to test whether a formula is in Q-CNF-SAT - a given target edge will only reverse at some future timestep if there is a satisfying assignment for the input formula. This can be done via a variety of timing gadgets and switches that ensure that the edges cycle in sync with one another to hit certain thresholds on certain vertices. A crossover gadget exists as well, proving that it is PSPACE-hard even in the case of planar graphs.

8 Two Player Bounded Games

A two-player game is analyzed to determine whether some player (usually the first to move) has a winning strategy. Games with 3+ players can be analyzed as two-player games by collapsing all but one player into an adversarial "player" who takes $n-1$ turns. We can consider the game solved if, for all sequences of moves by Player 2, there exists a sequence of moves by Player 1 that can win the game. Since the players take turns, these quantifiers are interleaved and form what's essentially a QSAT instance.

One such game is the SAT-style game defined by Schaefer [4]. In this game, players take turns assigning truth values to unassigned variables of a Boolean formula. There are two kinds of game and three kinds of win.

- *Game Types*
  - Impartial: Any player can set any unassigned variable on their turn.
  - Partisan: Variables are evenly split among two colors, and player 1 may only assign variables of color 1. Same for player 2 and color 2.
- *Win Types*
  - Default: Player 1 wins if the formula is satisfied at the end, Player 2 wins if it isn't.
  - Seek: The first player to satisfy the formula wins (where all unassigned variables are null).
  - Avoid: The first player to satisfy the formula loses (where all unassigned variables are null).

With this classification in mind, Schaefer found the following games are PSPACE-complete:

- Impartial Default Positive 11-CNF-SAT
- Impartial Default Positive 11-DNF-SAT
- Partisan Default CNF-SAT
- (Impartial/Partisan) Avoid Positive CNF-SAT
- (Impartial/Partisan) Seek Positive CNF-SAT
- (Impartial/Partisan) Avoid Positive 2-DNF SAT
- (Impartial/Partisan) Seek Positive 3-DNF SAT

Another such game is Kayles, which requires players to alternate marking nodes of a graph, and no player may choose a node adjacent to a marked node (i.e. they build an independent set). The loser is the first one who has no more moves. This problem is PSPACE-complete. The version of the problem where the graph is bipartite and each player is given half of the nodes that only they may mark is also PSPACE-complete.

Next, we analyze the game of Geography, which is simply taking turns moving a token along edges of a digraph where nodes (in Node-Geography) or edges (in Edge-Geography) can only be used one time by either player, and the loser is the first who cannot make a move. The problem asks, given a board state, which player will win. This problem is PSPACE-complete for Node-Geography on directed graphs, but in P for undirected. Similarly, this problem is PSPACE-complete for Edge-Geography on both directed and undirected graphs, but in P for bipartite.

Finally, we look at the game of Reversi (aka Othello). The game is in PSPACE because the number of moves is limited by the size of the board. The game is PSPACE-complete via a reduction from directed bipartite Node-Geography with max degree at most 3 [5]. This is achieved by having edges be represented by white lines that are flipped by black to correspond to a choice, and by having certain types of rail gadgets that force play to proceed in an order. This culminates in Black being able to place a piece in the lower right and win, if and only if they can win the Geography game.

9 Open Problems

1. The metatheorems enable easy hardness results for many games, but they use generalized games and require very intricate and specific setups that may never occur in the real game. It remains to be shown for many games how this setup can be constrained to results only achievable within the mechanics of the game (e.g. not with custom levels made out of

the engine). An example of this is Minecraft, which is clearly PSPACE-hard since it can simulate arbitrary circuits and restrict player movement towards a goal (usually the End Portal) based on those circuits, using only a map made in-game and placing the player in Adventure Mode; or Magic the Gathering, which is AH-hard using only two tournament-legal decks [2]. However, other games may not be able to create the same contrived scenarios as in their reductions.

2. Many games have known membership results but not hardness results, including the hardness of Arimaa, the hardness of Nine Men's Morris, and the hardness of Quoridor (a personal favorite). The metatheorems could potentially be applied to many games lacking hardness results to attempt to prove them.

3. It remains to be shown if there are metatheorems for proving EXPTIME-completeness.

4. The P = PSPACE and PH = PSPACE problems are two results questioning the relationship between PSPACE and other classes. The first asks whether P = PSPACE (a stronger result than P = NP, since NP is contained in PSPACE) and the second asks whether PSPACE is exactly the entirety of the polynomial hierarchy or whether there exists a problem in PSPACE outside of PH.

10 Important Problems

- QSAT - PSPACE-complete (original proof in book)

  Given a fully quantified Boolean formula, determine if it is true.

- DCL - PSPACE-complete (original proof in book)

  Given an instance of constraint logic, determine if a given edge will ever be activated.

- SAT Games - PSPACE-complete [4]

  Given a fully-quantified Boolean formula (of a certain type), where players alternate fixing the values of variables, determine if a specific player has a winning strategy. (The rules vary depending on the variant being played.)

- Kayles - PSPACE-complete [4]

  Given a graph and two players alternating to select nodes of an independent set, determine if a player has a winning strategy.

- Geography - PSPACE-complete [3, 4]

  Given a directed graph where players may not revisit a node/edge (depending on the variant), determine if a player has a winning strategy. (Node version is PSPACE-complete on digraphs, Edge version retains PSPACE-completeness on undirected general graphs but is in P for bipartite graphs)

References

[1] Greg Aloupis, Erik D. Demaine, and Alan Guo. Classic Nintendo games are NP-hard. Computing Research Repository, abs/1203.1895, 2012. http://arxiv.org/abs/1203.1895

[2] Stella Biderman. Magic: the Gathering is as Hard as Arithmetic. Artificial Intelligence, abs/2003.05119, 2019. https://arxiv.org/abs/2003.05119

[3] Aviezri S. Fraenkel, Edward R. Scheinerman, and Daniel Ullman. Undirected edge geography. Theoretical Computer Science, 112(2): 371–381, 1993. https://www.sciencedirect.com/science/article/pii/030439759390026P.

[4] Thomas J. Schaefer. On the complexity of some two-person perfect information games. Journal of Computer and Systems Science, 16:185–225, 1978. https://www.sciencedirect.com/science/article/pii/0022000078900454

[5] Shigeki Iwata and Takumi Kasai. The Othello game on an $n \times n$ board is PSPACE-complete. Theoretical Computer Science, 123(2):329–340, January 1994.