Paintbucket on graphs is PSPACE-complete

Ethan J. Saunders and Peter Selinger
Dalhousie University

Abstract

The game of Paintbucket was recently introduced by Amundsen and Erickson. It is played on a rectangular grid of black and white pixels. The players alternately fill in one of their opponent's connected components with their own color, until the entire board is just a single color. The player who makes the last move wins. It is not currently known whether there is a simple winning strategy for Paintbucket. In this paper, we consider a natural generalization of Paintbucket that is played on an arbitrary simple graph, and we show that the problem of determining the winner in a given position of this generalized game is PSPACE-complete.

1 Introduction

The game of Paintbucket was recently introduced by Amundsen and Erickson [1]. It is played on a rectangular grid of black and white pixels. The players alternately fill in one of their opponent's connected components with their own color, until the entire board is just a single color. Here, two pixels are considered to be connected if they share a common edge. The player who makes the final move wins. An example game of Paintbucket is shown in Figure 1(a). It is not currently known whether there is a simple winning strategy for Paintbucket.

In this paper, we consider a natural generalization of Paintbucket that is played on an arbitrary simple graph. Consider a simple undirected graph G, given by a set V of vertices and a set E of two-element subsets of V called edges. We initially assign a color, black or white, to every vertex. Two black vertices are connected if there is a path (of length zero or greater) between them that only passes through black vertices. A *black group* is a non-empty connected component of black vertices, i.e., a maximal connected set of black vertices. White groups are defined analogously. A move by a player consists of picking one of the opponent's groups and flipping the colors of all of its vertices. The winner is again the player who makes the last move. Note that Paintbucket played on a square grid graph, as in Figure 1(b), is exactly the same thing as the original version of Paintbucket played on a rectangular grid of pixels.

In this paper, we prove that the problem of determining the winner in a given position of Paintbucket on graphs is PSPACE-complete.

Related work. Burke and Tennenhouse studied the game Flag Coloring [2], which in the case of two colors can be thought of as an impartial version of Paintbucket. They showed that Flag Coloring on graphs is PSPACE-complete.

2 Avoider-enforcer games

We will show the PSPACE-completeness of Paintbucket on graphs by reduction from a known PSPACE-complete problem, namely the decision problem for *avoider-enforcer games*, which we now define.

Definition 2.1. The avoider-enforcer game is played on a pair $(C, \mathcal{A})$, where C is a finite set of *cells* and $\mathcal{A} = (A_j)_{j \in J}$ is a family of subsets of C, which we call the *avoider sets*. The players, who are called the *avoider* and the *enforcer*, take turns. On each turn, a player claims one cell, which afterwards cannot be

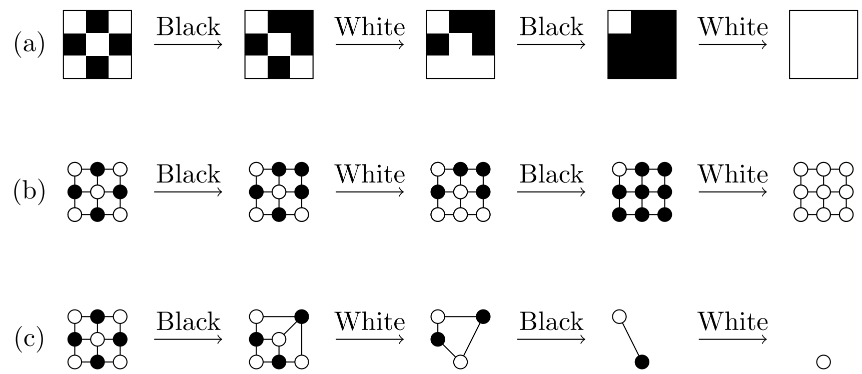

Figure 1: (a) An example game of Paintbucket played by the authors. White makes the last move and therefore wins. (b) The same game, played on a graph. (c) The same game, played on a bipartite graph.

claimed again (we can think of the players as "coloring" the cells). The game finishes when all cells are colored, and the avoider loses if and only if they have colored all the elements of some avoider set.

Remark 2.2. The following is an equivalent description of the avoider-enforcer game. Given a position $(C, \mathcal{A})$, a move by the avoider consists of choosing some $c \in C$, and then the new position is $(C', \mathcal{A}')$, where

$$C' = C \setminus \{c\} \quad \text{and} \quad \mathcal{A}' = (A_j \setminus \{c\})_{j \in J}.$$

A move by the enforcer in position $(C, \mathcal{A})$ consists of choosing some $c \in C$, and then the new position is $(C'', \mathcal{A}'')$, where

$$C'' = C \setminus \{c\} \quad \text{and} \quad \mathcal{A}'' = (A_j)_{j \in J,\, c \notin A_j}.$$

In other words, the avoider removes a cell from C and from all avoider sets, whereas the enforcer removes a cell from C and removes all avoider sets containing it. The game ends when $C = \emptyset$. In this case, either $\mathcal{A} = \emptyset$, in which case the avoider wins, or $\mathcal{A} = \{\emptyset\}$, in which case the enforcer wins.

The decision problem for avoider-enforcer games is the following: given a position $(C, \mathcal{A})$ and a player to move, decide whether the avoider has a winning strategy. This problem is known to be PSPACE-complete, even in the case where each avoider set consists of exactly 6 cells [3].

Remark 2.3. The decision problem for avoider-enforcer games remains PSPACE-complete even if we restrict the possible instances to those where the number of cells $|C|$ is even and the avoider moves first. Indeed, given a position where the enforcer moves first, we can simply add one additional cell c to C and to none of the avoider sets, and let the avoider move first; it is easy to see that it is in the avoider's interest to play their first move at c, after which the game is equivalent to what we started with. We can therefore assume without loss of generality that the avoider moves first. Now, consider the case where $|C|$ is odd. We modify the game by adding one more cell c' and exactly one additional avoider set $\{c'\}$. Clearly, the avoider loses if they ever play at c', and it is also easy to see that the enforcer prefers every other move to c'. Therefore, we can assume without loss of generality that c' will be the last cell played; since it is the enforcer who plays last, this game is equivalent to the one that we started with.

3 Paintbucket on bipartite graphs

The game of Paintbucket on graphs was defined in the introduction. We now give an equivalent description where every connected component of a color is contracted to a single vertex.

The game is played on a connected bipartite graph (V_b, V_w, E), where V_b and V_w are finite sets of vertices that we call *black* and *white* vertices, respectively, and $E \subseteq V_b \times V_w$ is a set of edges, each of which connects a black vertex to a white one.

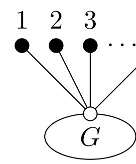

Figure 2: A bipartite graph with k black leaves attached to a white vertex.

A move by Black consists of choosing one white vertex v, coloring it black, and then immediately merging it with all of its (necessarily black) neighbors. More precisely, by "merging", we mean that all of the neighbors $w_1, \ldots, w_k$ of v are deleted from the graph, and new edges are added between v and all the (necessarily white) neighbors of $w_1, \ldots, w_k$. The resulting graph is again bipartite and connected. A move by White is defined dually. The game ends when the graph consists of a single vertex. Black wins if and only if the final vertex is black. Equivalently, the winner is the player who made the last move.

It is easy to see that this description of Paintbucket using bipartite graphs is equivalent to the game of Paintbucket on graphs described in the introduction. For example, Figure 1(c) shows the same game as Figure 1(a) and (b), played in this setting. From now on, when we refer to Paintbucket, we mean Paintbucket on bipartite graphs, unless otherwise stated.

The following lemmas state some properties of Paintbucket that will be useful to us later.

Lemma 3.1. *Consider Paintbucket played on a bipartite graph G with a distinguished white vertex v to which k black leaves are attached, as in Figure 2. Let m be the number of white vertices of G. Then if $m \leqslant k$, Black has a first-player winning strategy. Moreover, if $m < k$, Black has a winning strategy regardless of who goes first.*

Proof. Black's strategy is to play anywhere except v, unless v is the only remaining white vertex. Therefore, each time Black plays, m decreases by exactly 1. Each time White plays, k decreases by at most 1. Therefore, when $m = 1$ and it is Black's turn, there is still at least one black leaf and Black wins by playing v. □

Lemma 3.2. *In Paintbucket on a bipartite graph, if n vertices have the same set of neighbors and a player plays at one of the n vertices, the remaining ones become leaves that share the same neighbor.*

Proof. Suppose that $v_1, \ldots, v_n$ are black vertices that have the same neighbors $u_1, \ldots, u_k$, and suppose White plays at v_1. Then v_1 and $u_1, \ldots, u_k$ are all contracted into a single vertex u, which becomes the unique neighbor of each of $v_2, \ldots, v_n$ in the resulting graph. So they all become leaves sharing the same neighbor u. □

Lemma 3.3. *Consider Paintbucket played on the complete bipartite graph $K_{m,n}$, i.e., the bipartite graph with m black vertices, n white vertices, and all possible edges. Then:*

(a) If $m, n > 1$, the position is a second-player win.

(b) If $m > 1$ and $n = 1$, the position is winning for Black, no matter who goes first.

(c) If $m = 1$ and $n > 1$, the position is winning for White, no matter who goes first.

(d) If $m = n = 1$, the position is a first-player win.

Proof. (d) is obvious, and (b) and (c) follow by Lemma 3.1 and its dual. To prove (a), note that if Black plays first, Black collapses all black vertices and leaves $n-1$ white vertices remaining. White wins by playing at the only remaining black vertex. If White plays first, the situation is dual. □

4 Paintbucket is PSPACE-complete

4.1 Statement of the main result

The decision problem for Paintbucket on bipartite graphs is the following: Given a connected bipartite graph and a player to move, determine whether Black has a winning strategy. For concreteness, we consider the problem size to be the number n of vertices of the graph. Note that it would be equally valid to use the number of edges, because the number of edges in a connected graph is between $n-1$ and n^2, which is polynomial in n. We will prove the following theorem:

Theorem 4.1. *The decision problem for Paintbucket on bipartite graphs is PSPACE-complete.*

4.2 Translation from avoider-enforcer games to Paintbucket

We prove the theorem by reducing the avoider-enforcer game to Paintbucket. Given a position $(C, \mathcal{A})$ of the avoider-enforcer game and a natural number K, we will define a connected bipartite graph $G_K(C, \mathcal{A})$ as follows.

Let $I = |C|$ and enumerate the cells as $C = \{c_1, \ldots, c_I\}$. Also let $\mathcal{A} = (A_j)_{j\in\{1,\ldots,J\}}$. From now on, unless otherwise specified, the indices i, j, and k will always range over the sets $\{1, \ldots, I\}$, $\{1, \ldots, J\}$, $\{1, \ldots, K\}$, respectively. The graph $G_K(C, \mathcal{A})$ will have vertices v_i, u_i, w_k, $t_{j,k}$, r, and s. We assume all of these vertices to be distinct. We refer to these vertices as being of *type* v, u, w, t, r, and s, respectively. We define $G_K(C, \mathcal{A}) = (V_b, V_w, E)$, where

$$\begin{array}{rcl} V_b &=& \{v_i\} \cup \{w_k\} \cup \{r\}, \\ V_w &=& \{u_i\} \cup \{t_{j,k}\} \cup \{s\}, \\ E &=& \{(r, u_i)\} \\ && \cup\, \{(v_i, u_i)\} \\ && \cup\, \{(v_i, s)\} \\ && \cup\, \{(t_{j,k}, r)\} \\ && \cup\, \{(s, r)\} \\ && \cup\, \{(t_{j,k}, w_k)\} \\ && \cup\, \{(s, w_k)\} \\ && \cup\, \{(v_i, t_{j,k}) \mid c_i \in A_j\}. \end{array}$$

The graph is shown schematically in Figure 3. In words, it can be described as follows: for each cell c_i of the avoider-enforcer game, we have a pair of a black vertex v_i and a white vertex u_i, connected to each other by an edge. We also have a cluster of K black vertices $w_1, \ldots, w_K$. For each avoider set A_j, we have a cluster of K white vertices $t_{j,1} \ldots, t_{j,K}$, which are connected to those v_i such that $c_i \in A_j$. All of the vertices w_k and $t_{j,k}$ form a complete bipartite graph. Finally, there is a *universal* black vertex r which is connected to all white vertices, and a *universal* white vertex s which is connected to all black vertices (including r).

4.3 Intended play, shenanigans, and simulation

Our eventual goal is to show that whoever wins the game $(C, \mathcal{A})$ also wins the corresponding game $G_K(C, \mathcal{A})$. To do so, we consider the following notion of "intended play" for the Paintbucket game. Note that there are I pairs of vertices $\{u_i, v_i\}$. The intended play is that the first I moves of the game occur on these vertices, with each pair being played in exactly once. The players are not obliged to play in the intended way, but we will show that any player who deviates from the intended play will lose the game immediately. This is proved in the next two lemmas. When a player deviates from intended play, we call it a *shenanigan* by that player.

Lemma 4.2 (White shenanigans)**.** *Assume $K \geqslant |C| + 2$, and suppose it is White's turn in the Paintbucket game on the graph $G_K(C, \mathcal{A})$. If White's next move is not at a vertex of type v, White loses.*

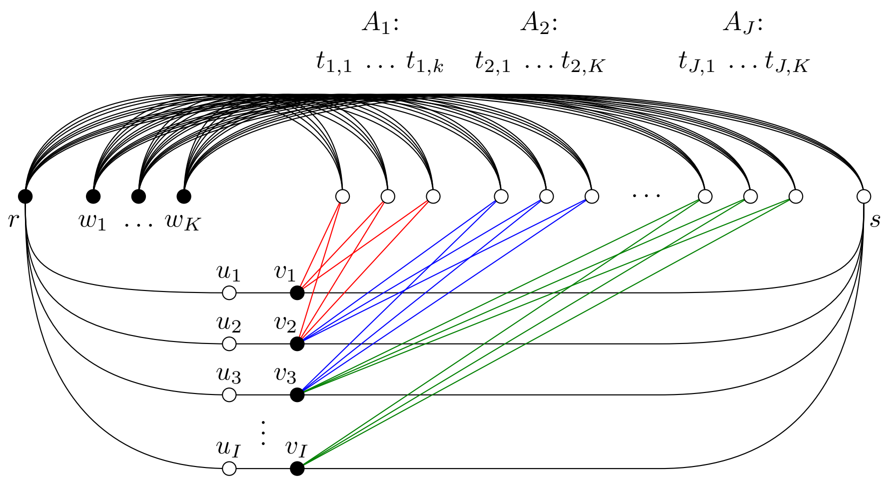

Figure 3: The graph $G_K(C, \mathcal{A})$. Here, we have assumed that $A_1 = \{c_1, c_2\}$, $A_2 = \{c_2, c_3\}$, and $A_J = \{c_3, c_I\}$.

Proof. There are two cases. Case 1: White's move is at the universal vertex r, then all white vertices will be contracted into a single vertex, while at least $w_1, \dots, w_K$ remain as black vertices. Then Black wins on the next move by playing the remaining white vertex. Case 2: White's move is at a vertex of type w, say at w_k. Since all of the vertices of types t and s are neighbors of w_k, they all get contracted into a single white vertex, which we may still call s. This turns all of the $w_{k'}$ where $k' \neq k$ into leaves by Lemma 3.2. Hence, the resulting graph has at least $K-1$ black leaves attached to the single vertex s. Also note that there are at most $I+1$ white vertices remaining in the graph, namely $u_1, \dots, u_I$ and s. By assumption, $K \geqslant |C|+2$, so $I+1 \leqslant K-1$. Then Black wins by Lemma 3.1. ☐

Lemma 4.3 (Black shenanigans)**.** *Assume $K \geqslant |C|+2$ and $\mathcal{A} \neq \emptyset$. Suppose it is Black's turn in the Paintbucket game on the graph $G_K(C, \mathcal{A})$. If Black's next move is not at a vertex of type u, Black loses.*

Proof. By assumption, $\mathcal{A}$ is non-empty, so let j be an index of some avoider set $A_j \in \mathcal{A}$. There are two cases. Case 1: Black's move is at the universal vertex s, then all black vertices will be contracted into a single vertex, while at least K white vertices remain, namely $t_{j,1}, \dots, t_{j,K}$. Then White wins on the next move by playing at the remaining black vertex. This leaves the case where Black's move is at a vertex of type t, say at $t_{j,k}$. Since all of $w_1, \dots, w_K$ and r are neighbors of $t_{j,k}$, they will all be contracted into a single vertex, which we may still call r. Because $t_{j,1}, \dots, t_{j,K}$ all had the same neighbors before Black's move, by Lemma 3.2, the remaining $K-1$ of them turn into leaves which share r as a common neighbor. Moreover, the resulting graph has at most $I+1$ black vertices remaining, namely $v_1, \dots, v_I$ and r. By assumption, $K \geqslant |C|+2$, so $I+1 \leqslant K-1$. Therefore, White wins by the dual of Lemma 3.1. ☐

Lemma 4.4 (Simulation)**.** *Consider a position $(C, \mathcal{A})$ of the avoider-enforcer game and let K be a natural number. In the game of Paintbucket on the graph $G_K(C, \mathcal{A})$, if Black moves at u_i, the resulting graph is isomorphic to $G_K(C', \mathcal{A}')$, where*

$$C' = C \setminus \{c\} \quad \textit{and} \quad \mathcal{A}' = (A_j \setminus \{c\})_{j \in J}$$

Similarly, if White moves at v_i, the resulting graph is isomorphic to $G_K(C', \mathcal{A}')$, where

$$C'' = C \setminus \{c\} \quad \textit{and} \quad \mathcal{A}'' = (A_j)_{j \in J,\, c \notin A_j}.$$

In other words, following Remark 2.2, these moves of Black and White exactly mirror the corresponding moves at c_i in $(C, \mathcal{A})$ by the avoider and enforcer, respectively.

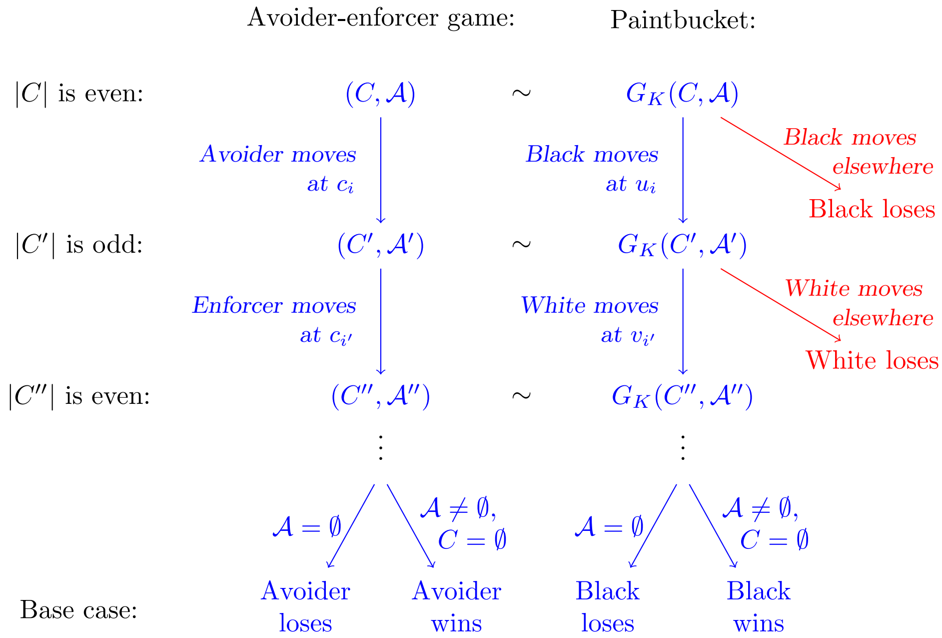

Figure 4: Schematic representation of the proof of Proposition 4.5

Proof. First, consider the Black move at u_i. The vertex u_i has exactly two neighbors, namely r and v_i. By the definition of Paintbucket on bipartite graphs, the result of the move is to delete u_i and merge v_i into r. Note that, since r is a universal vertex, all of the neighbors of v_i are already neighbors of r, so r gains no new neighbors. The move is effectively equivalent to deleting the pair of vertices u_i, v_i from the graph, along with all of their incident edges. This is exactly equivalent to removing the cell c from C and all avoider sets.

Second, consider the White move at v_i. The vertex v_i has several neighbors, namely u_i, s, and for each avoider set A_j that contains c_i, all of the $t_{j,k}$. The effect of moving at v_i is to delete v_i and merge all of its neighbors into a single vertex, which we may still call s. Since s is universal, it already has all black vertices as neighbors, and therefore gains no new neighbors. Thus, White's move is equivalent to simply deleting v_i, u_i, and all the clusters $\{t_{j,k}\}$ corresponding to avoider sets A_j that contain c_i. This is exactly equivalent to removing the cell c from C and removing all avoider sets containing c from $\mathcal{A}$. □

4.4 Proof of the main result

The PSPACE-completeness of Paintbucket is a consequence of the following proposition, which relates the winner of $G_K(C, \mathcal{A})$ to that of $(C, \mathcal{A})$.

Proposition 4.5. *Consider a position $(C, \mathcal{A})$ of the avoider-enforcer game, and assume that $K \geqslant |C| + 2$.*

(a) If $|C|$ is even, Black has a first-player winning strategy in the Paintbucket game $G_K(C, \mathcal{A})$ if and only if the avoider has a first-player winning strategy in $(C, \mathcal{A})$.

(b) If $|C|$ is odd, Black has a second-player winning strategy in the Paintbucket game $G_K(C, \mathcal{A})$ if and only if the avoider has a second-player winning strategy in $(C, \mathcal{A})$.

Proof. We prove properties (a) and (b) by simultaneous induction on $|C|$. We consider three cases. See also Figure 4.

Case 1. $\mathcal{A} = \emptyset$. Since there are no avoider sets, the avoider wins the avoider-enforcer game no matter how the players play. We must show that Black has a winning strategy in the Paintbucket game on $G_K(C, \mathcal{A})$.

Since $\mathcal{A} = \emptyset$, there are no vertices of type t. Therefore, $w_1, \ldots, w_K$ have s as their only neighbor, so there are at least K black leaves attached to a single vertex. Also, the graph has at most $I+1$ white vertices, namely $u_1, \ldots, u_I$ and s. We assumed $K \geqslant |C| + 2 = I + 2$, therefore $I + 1 < K$. Then Black wins by Lemma 3.1, no matter whose turn it is.

Case 2. $\mathcal{A} \neq \emptyset$ and $C = \emptyset$. Then (b) is vacuously true. To prove (a), first note that the avoider-enforcer game is already finished and has been won by the enforcer (since there is an avoider set, which is necessarily empty, and of which the avoider has therefore claimed all members). So we must show that Black loses the game $G_K(C, \mathcal{A})$, moving first. But since $C = \emptyset$, there are no vertices of types u. Black loses by Lemma 4.3.

Case 3. $|C| > 0$ and $|\mathcal{A}| > 0$. To prove (a), assume $|C|$ is even. By Lemma 4.3, Black has no possible winning moves in the Paintbucket game except at vertices of type u. We claim that u_i is a winning move for Black in the Paintbucket game $G_K(C, \mathcal{A})$ if and only if c_i is a winning move for the avoider in the avoider-enforcer game $(C, \mathcal{A})$. Indeed, this follows from the simulation lemma and the induction hypothesis. Namely, if the avoider's move at c_i results in the position $(C', \mathcal{A}')$, then Black's move at u_i results in a position that is isomorphic to $G_K(C', \mathcal{A}')$. The move at u_i is winning if and only if Black has a second-player winning strategy in $G_K(C', \mathcal{A}')$, which, by the induction hypothesis and the fact that $|C'|$ is odd, is the case if and only if the avoider has a second-player winning strategy in $(C', \mathcal{A}')$, which is the case if and only if the move at c_i was winning. The proof of (b) is analogous but dual. □

We can now prove the main theorem, i.e., the PSPACE-completeness of Paintbucket on graphs.

Proof of Theorem 4.1. It is easy to see that Paintbucket is in PSPACE. To show PSPACE-hardness, consider any position $(C, \mathcal{A})$ of the avoider-enforcer game. As noted in Remark 2.3, we can assume without loss of generality that $|C|$ is even and that the player to move is the avoider. We construct a position of Paintbucket as follows. Let $K = |C| + 2$ and consider Paintbucket on the graph $G_K(C, \mathcal{A})$. Note that the size of this graph is polynomial in the size of the avoider-enforcer instance. Then by Proposition 4.5, any algorithm that can determine the winner of the game $G_K(C, \mathcal{A})$ can also determine the winner of $(C, \mathcal{A})$; it follows that the decision problem for Paintbucket is at least as hard as the decision problem for the avoider-enforcer game. Therefore, it is PSPACE-hard. □

5 Conclusion

We showed that the game of Paintbucket on graphs is PSPACE-complete. The obvious open question is whether the original version of Paintbucket, played on a square grid, is also PSPACE-complete. Our current method does not shed any light on this question, but it would be fun to figure it out in future work.

References

[1] M. Amundsen and A. Erickson. Paintbucket. `https://boardgamegeek.com/boardgame/362682/paintbucket`, 2022. Accessed: 2024-11-20.

[2] K. Burke and C. Tennenhouse. Vexing vexillological logic. *International Journal of Game Theory*, 2024. Also available from arXiv:2212.10631.

[3] V. Gledel and N. Oijid. Avoidance games are PSPACE-complete. In *Proceedings of the 40th International Symposium on Theoretical Aspects of Computer Science (STACS 2023)*, volume 254 of *Leibniz International Proceedings in Informatics*, pages 34:1–19, 2023.