



## THE GAME OF BLOCKING PEBBLES

**Kyle Burke**

*Dept. of Computer Science, Plymouth State University, New Hampshire*  
kyleburk@usc.edu

**Matthew Ferland**

*Dept. of Computer Science, University of Southern California, California*  
mferland@usc.edu

**Michael Fisher**

*Dept. of Mathematics, West Chester University, Pennsylvania*  
mfisher@wcupa.edu

**Valentin Gledel**

*Dept. of Computer Science, University Grenoble Alpes, Grenoble, France*  
valentin.gledel@univ-grenoble-alpes.fr

**Craig Tennenhouse**

*Dept. of Mathematical Sciences, University of New England, Maine*  
ctennenhouse@une.edu

*Received: 1/25/21, Revised: 7/13/21, Accepted: 10/5/21, Published: 12/21/21*

### Abstract

Graph pebbling is a well-studied single-player game on graphs. We introduce the game of BLOCKING PEBBLES which adapts Graph Pebbling into a two-player strategy game in order to examine it within the context of combinatorial game theory. Positions with game values matching all integers, all numbers, and many infinitesimals and switches are found. This game joins the ranks of other combinatorial games on graphs, games with discovered moves, and partisan games with impartial movement options. The computational complexity of the general case is shown to be PSPACE-hard.

### 1. Introduction

Graph pebbling is an area of current interest in graph theory. In an undirected graph  $G$ , a root vertex  $r$  is designated. Heaps of pebbles are placed on the vertices of  $G$ , with a legal move consisting of choosing a vertex  $v$  with at least two pebbles,

removing two pebbles, and placing a single pebble on a neighbor of  $v$ . The goal is to *pebble*, or place a pebble on, the vertex  $r$ . The *pebbling number* of  $G$ , denoted  $\pi(G)$ , is the fewest number of pebbles necessary so that any initial distribution of  $\pi(G)$  pebbles among the vertices of  $G$ , and any vertex of  $G$  chosen as the root, has a sequence of moves resulting in the root being pebbled.

Introduced by Chung in 1989 [5], a number of results on pebbling of different families of graphs have been found. Of note are pebbling numbers of paths, cycles [13], and continuing work on a conjecture of Graham's on the Cartesian products of graphs [5]. Time complexity is also known, both for determination of  $\pi(G)$  and for the minimum number of moves in a successful pebbling solution, for general graphs. See [9] for a survey of results in graph pebbling.

The results and language here are in reference to combinatorial game theory (CGT). The *nim sum*, also called the *digital sum*, of non-negative integers is the result of their sum in binary without carry. This is denoted  $x_1 \oplus x_2$  if there are only two numbers, and in the case of more we use the notation  $\sum \oplus x_i$ . For more notation and background on the computation of CGT game values, we refer the reader to [1,3].

In Sect. 2 we introduce a two-player combinatorial ruleset based on graph pebbling, with subsequent sections addressing results on both impartial and partisan positions. This game involves strategic play that results in blocking the moves of one's opponent. AMAZONS is another well-known game which also involves a notion of blocking. However, in AMAZONS, the blocking is always permanent (burnt square) or temporary (queen occupies square). Due to the standard pebbling toll in BLOCKING PEBBLES, each pebble only has mobility for a finite time.

There are several pebbling games which appear in the literature [10,11,14]. The one which is most similar to the game introduced here was originated by Lagarias and Saks in 1989 to solve a problem of Erdős. These games do not include the non-toll moves across an edge in the "wrong direction." This type of move is unique to BLOCKING PEBBLES (as far as we are aware). There are also other pebbling games older than the one introduced by Lagarias and Saks [10]. These games bear no resemblance to BLOCKING PEBBLES and are used to study graph algorithm complexity.

## 2. Ruleset and Play

A game of BLOCKING PEBBLES consists of a directed acyclic graph  $G$  and a 3-tuple  $(b, r, g)$  at each vertex of  $G$ , representing the number of blue, red, and green pebbles. Left may move blue and green pebbles, while Right may move red and green. This follows one convention of BRG-HACKENBUSH (see Fig. 1) wherein players may remove an edge of their own color or the neutral color green. In

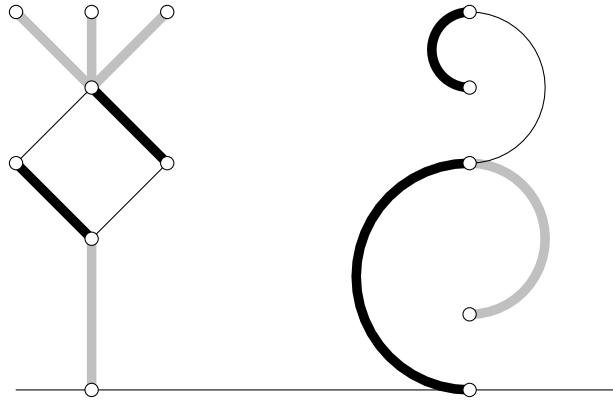


Figure 1: A BRG-HACKENBUSH position with blue, red, and green represented by thin, thick black, and grey lines, respectively.

BRG-HACKENBUSH all dyadic rationals and numbers are achievable game values. In addition, when allowing for infinite positions, all real numbers and ordinals are achievable values, but switches are not. By contrast, in BLOCKING PEBBLES players may move any number of pebbles at a single vertex within certain constraints on the graph and pebble distribution. In this way, BLOCKING PEBBLES is similar to GRAPH NIM [8] [4].

**Ruleset 1.** Given a tuple of the form  $(b, r, g)$  at each vertex of a directed acyclic graph  $G$ , Left can make one of the following two moves from the vertex  $v$ .

1. Move a positive number of blue and/or green pebbles from  $v$  to an in-neighbor of  $v$
2. Remove two blue and/or green pebbles from  $v$  and place one on an out-neighbor of  $v$  and discard the other

No blue pebbles can be moved to a vertex with a non-zero number of red pebbles. Right has the obvious symmetric moves.

Play proceeds following the normal play convention, where the last player to make a legal move wins.

Note that if Left removes one blue and one green pebble from  $v$  she may add the green to  $v$ 's out-neighbor. However, it is always preferable to instead add the blue as this results in a position with more blue pebbles and increases the number of vertices blocked by Left.

As an example, consider the position in Figure 2. At the top is a position in BLOCKING PEBBLES. Note that Left cannot move any blue pebbles from vertex  $A$

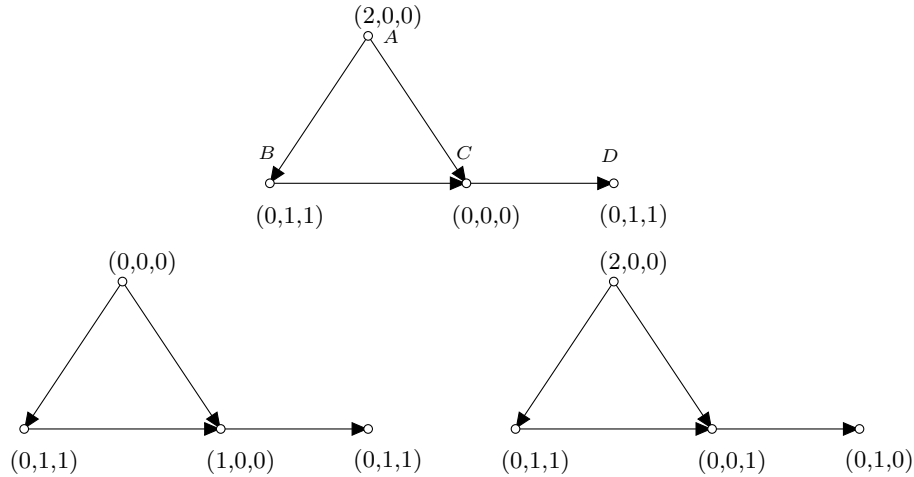


Figure 2: A position in BLOCKING PEBBLES and two of Left's options

to  $B$  since  $B$  already contains a red pebble. However, Left can move a single blue pebble from  $A$  to  $C$  at a cost of one blue pebble. She can also move the one green pebble from  $D$  to  $C$ .

An interesting property of this ruleset is the existence of *discovered moves*, similar to discovered attacks in chess. A player may be unable to move at one point in the game, but after their opponent moves then the game is once again playable by the first player. As an example consider a simple out-star with two red pebbles on the source and a single blue pebble on a sink node. Left has no moves, but once Right moves Left can move their pebble to the source. The presence of discovered moves precludes this being a strong placement game. For more on these types of games, see [7] and [12].

### 3. Blue-Red-Green Blocking Pebbles

In this section we will address some families of game values that are achievable in BLOCKING PEBBLES. We will only address finite graphs, hence we will not encounter non-dyadic rationals. This is similar to BRG-HACKENBUSH, described in Sect. 1. Due to the complexity of analysis, we will also restrict our graphs to orientations of stars, paths, and small graphs.

We begin with a simple result.

**Theorem 3.1.** *For every  $k \in \mathbb{Z}$  there is a position in BLOCKING PEBBLES with value  $k$ .*

*Proof.* Let  $G$  be a single arc directed from  $u$  to  $v$ . If  $k > 0$  then place  $2k$  blue pebbles and a single red pebble on  $u$ , and no pebbles on  $v$ . Switch red and blue pebbles if  $k < 0$ . This allows for  $k$ -many moves for Left by moving blue pebbles from  $u$  to  $v$ , but the presence of a red pebble on  $u$  prevents moving any blue pebbles in the reverse direction. Zero is trivially achieved by a graph with no pebbles, or any number of other pebble distributions.  $\square$

Regarding infinitesimals,  $\downarrow$  is realized by an out-star with two leaves; that is, a vertex  $u$  with out-neighbors  $v_1, v_2$ . Vertex  $v_1$  has a blue pebble, and  $v_2$  has one red and one green pebble. Left can move the blue or green pebble to  $u$ , which is simple to identify as  $*$ . Right, however, can move the green to the source vertex  $u$  resulting in  $*$ , the red to  $u$  resulting in zero, or both red and green pebbles to  $u$  which is also a zero position. The initial position is  $\{*[0, *]\} = \downarrow$ .

Due to the blocking rule, BLOCKING PEBBLES is relatively unique among partisan combinatorial games. In BRG-HACKENBUSH the presence of a move for one player does not inhibit moves for the other. In CLOBBER, another two-player partisan combinatorial game, (see [2]), the presence of a red piece actually encourages movement for Left, and vice versa. This is a property common to all dicot games. However, in BLOCKING PEBBLES a single well-placed blue pebble, for example, can cut off many of Right’s moves. The only other well-known ruleset with this property appears to be AMAZONS, which does not allow for discovered moves. It is natural, then, that many positions result in game values that are switches.

Parts (1) and (4) of the next result show that every integer switch is achievable with a specified pebbling configuration on the out-star  $K_{1,2}$ .

In the following lemma, we use the following notation for a bLue/Red pebbling configuration of the out-star  $K_{1,2}$ :  $[(a, b), [c, d], [e, f]]$  is the configuration with  $a$  blue pebbles and  $b$  red pebbles on the central vertex,  $c$  blue pebbles and  $d$  red pebbles on one of the pendant vertices, and  $e$  blue pebbles and  $f$  red pebbles on the other pendant vertex.

**Lemma 3.2.** *The following results pertain to a given BLOCKING PEBBLES configuration on the out-star  $K_{1,2}$ .*

1. For  $c \geq 1$ , the position  $[(a, b), [0, c], [0, 0]]$  has value  $-\lfloor \frac{b}{2} \rfloor$  if  $a = 1$  and value  $\{\lfloor \frac{a}{2} \rfloor - 1 \mid \lfloor \frac{a-b}{2} \rfloor + 1\}$  if  $a \geq 2$ ,
2. for  $a, b, c, d \geq 1$ , the position  $[(a, b), [c, 0], [0, d]]$  has value  $\lfloor \frac{a-b}{2} \rfloor$ ,
3. for  $a, b, c, d, e \geq 1$ , the position  $[(a, b), [c, 0], [d, e]]$  has value  $\lfloor \frac{a-1}{2} \rfloor$ ,
4. for  $a, b, c, d \geq 1$ , the position  $[(0, 0), [a, b], [c, d]]$  has value  $\{a + c - 1 \mid -(b + d - 1)\}$ ,
5. for  $a, b, c \geq 1$ , the position  $[(0, 0), [a, b], [0, c]]$  has value  $\{a - 1 \mid -(3(b + c) - 5)\}$ ,

6. for  $a, b \geq 1$ , the position  $[(0, 0), [a, b], [0, 0]]$  has value  $\{3a - 5 \mid -(3b - 5)\}$ ,

7. for  $b \geq 1$ ,  $[(1, 0), [0, b], [0, 0]]$  and  $[(2, 0), [0, b], [0, 0]]$  are both zero positions.

*Proof.* For **Case (1)**, the position  $[(1, 1), [c, 0], [0, 0]]$  is the zero position. It is also readily checked that the position  $[(1, 2), [c, 0], [0, 0]]$  has value 0.

If  $b > 2$ , then Left has no move from  $[(1, b), [c, 0], [0, 0]]$ . From  $[(1, b), [c, 0], [0, 0]]$ , Right may move to the position  $[(1, b - 2), [c, 0], [0, 1]]$ , which has value  $\lfloor \frac{-b+1}{2} \rfloor = -\lfloor \frac{b}{2} \rfloor + 1$  by induction. Hence,  $[(1, b), [c, 0], [0, 0]]$  has value  $-\lfloor \frac{b}{2} \rfloor$  as required.

If  $a \geq 2$ , then Left's best move from  $[(a, b), [c, 0], [0, 0]]$  is to  $[(a - 2, b), [c, 0], [1, 0]]$  which has value  $\lfloor \frac{a-2}{2} \rfloor$  (Right has no move from this position and Left has  $\lfloor \frac{a-2}{2} \rfloor$  moves). Right's only move is to  $[(a, b - 2), [c, 0], [0, 1]]$  which has value  $\lfloor \frac{a-b+2}{2} \rfloor$ , also by induction. Hence,  $[(a, b), [c, 0], [0, 0]]$  has value  $\{\lfloor \frac{a}{2} \rfloor - 1 \mid \lfloor \frac{a-b}{2} \rfloor + 1\}$  when  $a \geq 2$ .

For **Case (2)**, it is clear that the position  $[(1, 1), [c, 0], [0, d]]$  is a zero position. If  $a \geq 2$ , then, from  $[(a, 1), [c, 0], [0, d]]$ , Left has a move to  $[(a - 2, 1), [c + 1, 0], [0, d]]$  and Right has no move. Thus,  $[(a, 1), [c, 0], [0, d]]$  has value  $\lfloor \frac{a-1}{2} \rfloor$ , by induction.

A similar argument establishes the claim that  $[(1, b), [c, 0], [0, d]]$  has value  $\lfloor \frac{1-b}{2} \rfloor$ .

Now if  $a, b \geq 2$ , then, from  $[(a, b), [c, 0], [0, d]]$ , Left has the move to  $[(a - 2, b), [c + 1, 0], [0, d]]$  and Right has the move to  $[(a, b - 2), [c, 0], [0, d + 1]]$ . By induction, we see that  $[(a, b), [c, 0], [0, d]]$  has value

$$\{\lfloor \frac{a-2-b}{2} \rfloor \mid \lfloor \frac{a-b+2}{2} \rfloor\} = \{\lfloor \frac{a-b}{2} \rfloor - 1 \mid \lfloor \frac{a-b}{2} \rfloor + 1\} = \lfloor \frac{a-b}{2} \rfloor.$$

For **Case (3)**, note that if  $a = 1$ , then there are no moves for either player; the formula given correctly yields the game value 0. If  $a = 2$ , then from the position  $[(2, b), [c, 0], [d, e]]$  Left has the move to  $[(0, b), [c + 1, 0], [d, e]]$ . From here, Left has no move and Right has  $e$  moves. Thus the position  $[(0, b), [c + 1, 0], [d, e]]$  has value  $-e$ . Hence,  $[(2, b), [c, 0], [d, e]]$  has value 0, as required.

If  $a > 2$ , then, from  $[(a, b), [c, 0], [d, e]]$ , Left can move to  $[(a - 2, b), [c + 1, 0], [d, e]]$  which has value  $\lfloor \frac{a-3}{2} \rfloor = \lfloor \frac{a-1}{2} \rfloor - 1$ , by induction. Right has no moves from  $[(a, b), [c, 0], [d, e]]$ . Hence,  $[(a, b), [c, 0], [d, e]]$  has value

$$\{\lfloor \frac{a-1}{2} \rfloor - 1 \mid \} = \lfloor \frac{a-1}{2} \rfloor$$

as desired.

For **Case (4)**, note that Left's only move from  $[(0, 0), [a, b], [c, d]]$  is to  $[(1, 0), [a - 1, b], [c, d]]$ . This last position has value  $a + c - 1$  by induction. Similarly, Right's only move from  $[(0, 0), [a, b], [c, d]]$  is to  $[(0, 1), [a, b - 1], [c, d]]$ . This position has value  $-(b + d - 1)$  by induction. It now follows that  $[(0, 0), [a, b], [c, d]]$  has value

$$\{a + c - 1 \mid -(b + d - 1)\}.$$

Cases (5) and (6) follow from the previous result and Case (7) is trivial.  $\square$

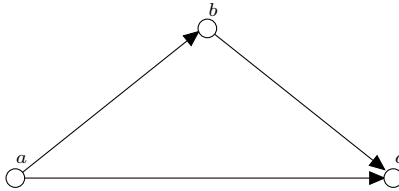


Figure 3: A transitive triple graph

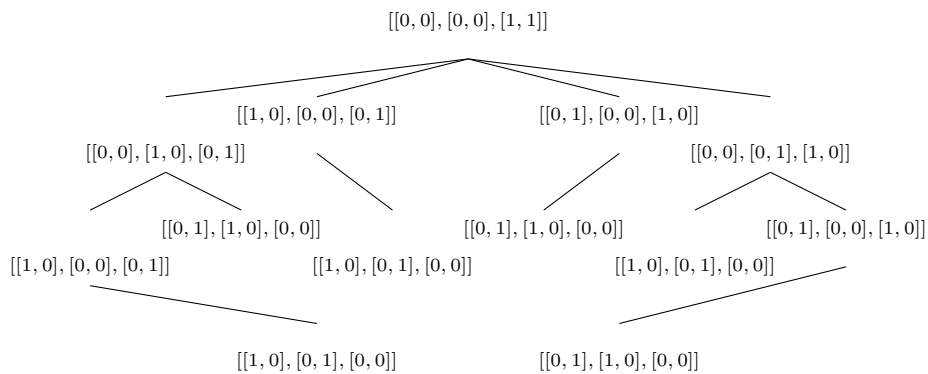


Figure 4: Game tree of the position  $[[0, 0], [0, 0], [1, 1]]$  on the transitive 3-cycle

We now consider a transitive 3-cycle graph (Figure 3) with vertices  $a, b, c$  and arcs  $ab, ac$ , and  $bc$ . The pebbling configurations considered below are written so that the first array entry corresponds to the source vertex,  $a$ , the second corresponds to  $b$ , and the third to the sink vertex,  $c$ .

An interesting result concerning the transitive 3-cycle crops up from the somewhat unnatural starting position where 1 blue pebble and  $k$  red pebbles occupy the same starting vertex. Specifically we prove that

**Theorem 3.3.** *For  $k > 1$  the pebbling configuration on the transitive 3-cycle given by  $[[0, 0], [0, 0], [1, k]]$  has game value  $(3 - 3k) +_{k-4}$ .*

We see the game tree of the base case in Figure 4.

In order to prove this result, we consider several positions which arise as subpositions of the above pebbling configuration.

**Lemma 3.4.** *Consider the following pebbling configurations of the transitive 3-cycle  $T$ . Then, the position*

1.  $[[1, 0], [0, j], [0, k]]$  has value  $-3k - 2j + 2$  if at least one of  $j$  or  $k$  is  $\geq 1$ ;
2.  $[[0, j], [1, 0], [0, k]]$  has value  $-3k - 2j + 3$  if  $j, k \geq 1$ ;
3.  $[[0, 0], [1, 0], [0, k]]$  has value  $-3k + 3$  if  $k \geq 2$  and value  $-\frac{1}{2}$  if  $k = 1$ ;
4.  $[[0, j], [1, 0], [0, 0]]$  has value  $-2j + 3$  if  $j \geq 2$  and value  $0$  if  $j = 1$ ;
5.  $[[0, j], [0, k], [1, 0]]$  has value  $-3k - 2j + 4$  if  $j, k \geq 1$ ;
6.  $[[0, j], [0, 0], [1, 0]]$  has value  $-2j + 4$  if  $j \geq 2$  and value  $1$  if  $j = 1$ ;
7.  $[[0, 0], [0, k], [1, 0]]$  has value  $\{-2k + 2 \mid -3k + 5\}$  if  $k \geq 2$  and value  $\frac{1}{2}$  if  $k = 1$ ;
8.  $[[0, 0], [0, j], [1, k]]$  has value  $\{-3k - 2j + 2 \mid -4k - 3j + 5\}$  if  $j \geq 2$  and  $k \geq 1$  and value  $\{-3k \mid -4k + 3\}$  if  $j = 1$  and  $k \geq 1$ ;
9.  $[[0, j], [0, 0], [1, k]]$  has value  $\{-3k - 2j + 3 \mid -4k - 2j + 5\}$  if  $j, k \geq 1$ ; and
10.  $[[0, \ell], [0, j], [1, k]]$  has value  $-4k - 3j - 2\ell + 4$  if  $j, k, \ell \geq 1$ .

*Proof.* All claims will be proven simultaneously using induction (on the height of the game tree). Base cases are easily checked and left to the interested reader.

**Case (1):** From  $[[1, 0], [0, j], [0, k]]$  Left has no move; Right's best move is to  $[[1, 0], [0, j + 1], [0, k - 1]]$ . By induction, this position has value  $-3k - 2j + 3$  by (1). Hence,  $[[1, 0], [0, j], [0, k]]$  has value

$$\{ \mid -3k - 2j + 3 \} = -3k - 2j + 2$$

as desired.

**Case (2):** Left again has no move from the starting position. Right's best move is to  $[[0, j + 1], [1, 0], [0, k - 1]]$ . If  $k = 1$ , this position has value  $-2j + 1$  by (4); if  $k \geq 2$ , this position has value  $-3k - 2j + 4$  by (2). In either case,  $[[0, j], [1, 0], [0, k]]$  has value  $-3k - 2j + 3$ .

**Case (3):** First suppose that  $k \geq 2$ . Left can move to  $[[1, 0], [0, 0], [0, k]]$  from  $[[0, 0], [1, 0], [0, k]]$ . From (1), this position has value  $-3k + 2$ . Right's best move is to  $[[0, 1], [1, 0], [0, k - 1]]$  with value  $-3k + 4$ . Hence,  $[[0, 0], [1, 0], [0, k]]$  has value

$$\{-3k + 2 \mid -3k + 4\} = -3k + 3.$$

If  $k = 1$ , then Left's only move becomes  $[[1, 0], [0, 0], [0, 1]]$  which has value  $-1$  by (1) and Right's only move is to  $[[0, 1], [1, 0], [0, 0]]$  with value  $0$  by (4). Thus,  $[[0, 0], [1, 0], [0, 1]]$  has value  $-\frac{1}{2}$ .



**Case (4):** For  $j \geq 3$ , Left has no move from  $[[0, j], [1, 0], [0, 0]]$  and Right can move to  $[[0, j - 2], [1, 0], [0, 1]]$  with value  $-2j + 4$ , by (2), giving  $[[0, j], [1, 0], [0, 0]]$  the game value of  $-2j + 3$ .

If  $j = 2$ , then Right's move is to  $[[0, 0], [1, 0], [0, 1]]$  with value  $-\frac{1}{2}$ . Thus,  $[[0, 2], [1, 0], [0, 0]]$  has a value of  $-1 (= -2 \cdot 2 + 3)$ .

Finally, if  $j = 1$ , then neither Left nor Right has a move from  $[[0, 1], [1, 0], [0, 0]]$  and so its value is 0.

**Case (5):** Let  $k \geq 2$ . Left has no move and Right can move to  $[[0, j + 1], [0, k - 1], [1, 0]]$  (Right's best move). By induction, this position has value  $-3k - 2j + 5$  giving  $[[0, j], [0, k], [1, 0]]$  the value  $-3k - 2j + 4$ .

If  $k = 1$ , then, again, Left has no move. However, Right can move to  $[[0, j + 1], [0, 0], [1, 0]]$ . By (6), this position has value  $-2j + 2$ , thus giving  $[[0, j], [0, 1], [1, 0]]$  the value  $-2j + 1$ .

**Case (6):** First we consider the case  $j > 2$ . Left's only move is to  $[[0, j], [1, 0], [0, 0]]$ . By (4) this position has value  $-2j + 3$ . Right's only move is to  $[[0, j - 2], [0, 1], [1, 0]]$ . By (5) this position has value  $-2j + 5$ . Hence, if  $j > 2$ , then  $[[0, j], [0, 0], [1, 0]]$  has value  $-2j + 4$ .

If  $j = 2$ ,  $[[0, 2], [1, 0], [0, 0]]$  has value  $-1$  by (4). Right's move to  $[[0, 0], [0, 1], [1, 0]]$  has value  $\frac{1}{2}$  by (7). Thus,  $[[0, 2], [0, 0], [1, 0]]$  has value 0.

Finally, if  $j = 1$ , then Left's move,  $[[0, 1], [1, 0], [0, 0]]$ , have value 0 and Right has no moves. Thus,  $[[0, 1], [0, 0], [1, 0]]$  has value 1.

**Case (7):** If  $k \geq 2$ , Left's move to  $[[1, 0], [0, k], [0, 0]]$  has value  $-2k + 2$  by (1). In this case, Right's move to  $[[0, 1], [0, k - 1], [1, 0]]$  has value  $-3k + 5$  by (5). Therefore  $[[0, 0], [0, k], [1, 0]]$  has value

$$\{-2k + 2 \mid -3k + 5\}.$$

If  $k = 1$ ,  $[[1, 0], [0, 1], [0, 0]]$  has value 0 and  $[[0, 1], [0, 0], [1, 0]]$  has value 1. Hence  $[[0, 0], [0, 1], [1, 0]]$  has value  $\frac{1}{2}$ .

**Case (8):** First suppose  $j \geq 2$  and  $k \geq 2$ . Then, Left's move to  $[[1, 0], [0, j], [0, k]]$  has value  $-3k - 2j + 2$  by (1). Right has two sensible moves: one to  $[[0, 1], [0, j], [1, k - 1]]$  and one to  $[[0, 1], [0, j - 1], [1, k]]$ . The former has value  $-4k - 3j + 6$  by (10) and the latter has value  $-4k - 3j + 5$ , also by (10). Thus,  $[[0, 0], [0, j], [1, k]]$  has value

$$\{-3k - 2j + 2 \mid -4k - 3j + 5\}.$$

Next we look at the case where  $j \geq 2$  and  $k = 1$ . Left's move to  $[[1, 0], [0, j], [0, 1]]$  has value  $-2j - 1$  by (1). Right's move to  $[[0, 1], [0, j], [1, 0]]$  has value  $-3j + 2$  by (5). Right's move to  $[[0, 1], [0, j - 1], [1, 1]]$  has value  $-3j + 1$  by (10). Hence,  $[[0, 0], [0, j], [1, 1]]$  has value

$$\{-2j - 1 \mid -3j + 1\}.$$

We now consider the case  $j = 1$  and  $k \geq 2$ . Left's only move is to  $[[1, 0], [0, 1], [0, k]]$ . This position has value  $-3k$  by (1). Right's move to  $[[0, 1], [0, 0], [1, k]]$  has value  $\{-3k + 1 \mid -4k + 3\}$  by (9) and his move to  $[[0, 1], [0, 1], [1, k - 1]]$  has value  $-4k + 3$  by (10). Therefore, the position  $[[0, 0], [0, 1], [1, k]]$  has value

$$\{-3k \mid \{-3k + 1 \mid -4k + 3\}, -4k + 3\}.$$

It can be shown that the option  $\{-3k + 1 \mid -4k + 3\}$  is reversible. Hence the canonical form of the position  $[[0, 0], [0, 1], [1, k]]$  has value

$$\{-3k \mid -4k + 3\}.$$

Finally, we consider the case  $j = 1$  and  $k = 1$ . Left's move from  $[[0, 0], [0, 1], [1, 1]]$  to  $[[1, 0], [0, 1], [0, 1]]$  has value  $-3$  by (2). Right's move to  $[[0, 1], [0, 0], [1, 1]]$  has value  $\{-2 \mid -1\} = -\frac{3}{2}$ . The move to  $[[0, 1], [0, 1], [1, 0]]$  has value  $-1$  by (5). Thus, the position  $[[0, 0], [0, 1], [1, 1]]$  has value  $\{-3 \mid -\frac{3}{2}\} = -2$ .

**Case (9):** First suppose that  $k \geq 2$ . Then Left's move from  $[[0, j], [0, 0], [1, k]]$  to  $[[0, j], [1, 0], [0, k]]$  has value  $-3k - 2j + 3$  by (2). Right's best move is to  $[[0, j], [0, 1], [1, k - 1]]$  with value  $-4k - 2j + 5$ , thus giving the position  $[[0, j], [0, 0], [1, k]]$  the game value of

$$\{-3k - 2j + 3 \mid -4k - 2j + 5\}.$$

If  $k = 1$ , Left's only move has value  $-2j$ , again by (2). Right's move to  $[[0, j], [0, 1], [1, 0]]$  has value  $-2j + 1$  by (5). Hence,  $[[0, j], [0, 0], [1, 1]]$  has value

$$\{-2j \mid -2j + 1\} = (-4j + 1)/2.$$

**Case (10):** Let  $j = 1$  and  $k \geq 2$ . Left has no move from this starting position and Right has three sensible moves. Right can move to  $[[0, \ell + 1], [0, 0], [1, k]]$  with value  $\{-3k - 2\ell + 1 \mid -4k - 2\ell + 3\}$  by (9), or to  $[[0, \ell + 1], [0, 1], [1, k - 1]]$  with value  $-4k - 2\ell + 3$  by (10), or to  $[[0, \ell], [0, 2], [1, k]]$  with value  $-4k - 2\ell + 2$  by (10). The last move is optimal for Right, hence the position  $[[0, \ell], [0, 1], [1, k]]$  has game value  $-4k - 2\ell + 1$ , as required.

If  $j = 1$  and  $k = 1$ , then Left has no move from  $[[0, \ell], [0, 1], [1, 1]]$  and Right has again three sensible moves. Right's move to  $[[0, \ell + 1], [0, 0], [1, 1]]$  has value  $(-4\ell - 3)/2$  by (9); Right's move to  $[[0, \ell], [0, 2], [1, 0]]$  has value  $-2\ell - 2$  by (5); and Right's move to  $[[0, \ell + 1], [0, 1], [1, 0]]$  has value  $-2\ell - 1$  by (5). Therefore, the position  $[[0, \ell], [0, 1], [1, 1]]$  has value  $-2\ell - 3$ .

If  $j \geq 2$  and  $k = 1$ , then Left has no move from  $[[0, \ell], [0, j], [1, 1]]$  and Right has three moves, each not costing a pebble to make: Right can move to  $[[0, \ell + 1], [0, j], [1, 0]]$  with value  $-3j - 2\ell + 2$  by (5); Right can move to  $[[0, \ell], [0, j + 1], [1, 0]]$  with value  $-3j - 2\ell + 1$  by (5); and Right can move to  $[[0, \ell + 1], [0, j - 1], [1, 0]]$

with value  $-3j - 2\ell + 1$  by (10). Hence, the position  $[[0, \ell], [0, j], [1, 1]]$  has value  $-3j - 2\ell$ .

Finally, if  $j \geq 2$  and  $k \geq 2$ , then, as in every other sub-case, Left has no move. Right has his usual three moves: Right can move to  $[[0, \ell + 1], [0, j], [1, k - 1]]$  with value  $-4k - 3j - 2\ell + 6$  by (10); Right can move to  $[[0, \ell], [0, j + 1], [1, k - 1]]$  with value  $-4k - 3j - 2\ell + 5$ ; or Right could move to  $[[0, \ell + 1], [0, j - 1], [1, k]]$  with value  $-4k - 3j - 2\ell + 5$ . Thus,  $[[0, \ell], [0, j], [1, k]]$  has game value  $-4k - 3j - 2\ell + 4$ .  $\square$

With Lemma 3.4 in hand, we can now prove Theorem 3.3.

*Proof.* Left has two moves from the starting position  $[[0, 0], [0, 0], [1, k]]$ : Left can move to  $[[1, 0], [0, 0], [0, k]]$  with value  $-3k + 2$  by Lemma 3.4(1) or to  $[[0, 0], [1, 0], [0, k]]$  with value  $-3k + 3$  by 3.4(3). The latter move is clearly the optimal move for her.

There are two types of moves that Right can make: Right can move to  $[[0, \ell], [0, 0], [1, k - \ell]]$ , where  $1 \leq \ell \leq k$ , or to  $[[0, 0], [0, j], [1, k - j]]$ , where  $1 \leq j \leq k$ .

First suppose that Right moves to  $[[0, \ell], [0, 0], [1, k - \ell]]$ , where  $1 \leq \ell < k$ . This position has value

$$\{-3k + \ell + 3 \mid -4k + 2\ell + 5\} = (-3k + \ell + 3) + \{0 \mid -k + \ell + 2\}$$

by Lemma 3.4(9).

Next, suppose that Right moves to  $[[0, k], [0, 0], [1, 0]]$ . This position has value

$$-2k + 4$$

by Lemma 3.4(6).

We now consider the other type of move for Right. Suppose that Right moves to  $[[0, 0], [0, j], [1, k - j]]$ , where  $1 < j < k$ . This position has value

$$\{-3k + j + 2 \mid -4k + j + 5\} = (-3k + j + 2) + \{0 \mid -k + 3\}$$

by Lemma 3.4(8).

Next, suppose that Right moves to  $[[0, 0], [0, 1], [1, k - 1]]$ . This position has value

$$\{-3k + 3 \mid -4k + 7\} = (-3k + 3) + \{0 \mid -k + 4\}$$

by Lemma 3.4(8).

Finally, suppose that Right moves to  $[[0, 0], [0, k], [1, 0]]$ . This position has value

$$\{-2k + 2 \mid -3k + 5\} = (-2k + 2) + \{0 \mid -k + 3\}$$

by Lemma 3.4(7).

We will now show that the move to  $[[0, 0], [0, 1], [1, k-1]]$  is Right's optimal move. First note that since  $\{0 \mid -k+4\} \leq 1$ , it follows that

$$(-3k+3) + \{0 \mid -k+4\} \leq -3k+4 < -2k+4.$$

Next, observe that if  $k = 2$ , then

$$(-3 \cdot 2 + 3) + \{0 \mid -2 + 4\} = -2 < -\frac{3}{2} = (-2 \cdot 2 + 2) + \{0 \mid -2 + 3\}.$$

If  $k \geq 3$ , then  $-k+2 < \{0 \mid -k+3\}$  and so it follows that

$$-3k+4 = (-2k+2) + (-k+2) < (-2k+2) + \{0 \mid -k+3\}.$$

Hence

$$(-3k+3) + \{0 \mid -k+4\} < (-2k+2) + \{0 \mid -k+3\}$$

for  $k \geq 2$ .

To show that

$$(-3k+3) + \{0 \mid -k+4\} < (-3k+\ell+3) + \{0 \mid -k+\ell+2\}, \quad \ell \geq 1,$$

it suffices to show that  $\{\ell \mid -k+2\ell+2\} + \{k-4 \mid 0\} > 0$ . To this end, note that Left's move to  $\ell + \{k-4 \mid 0\}$  is a winning first move. Right's move to  $-k+2\ell+2 + \{k-4 \mid 0\}$ , leads to  $(-k+2\ell+2) + (k-4) = 2\ell-2 \geq 0$ , after Left's response. Right's move to  $\{\ell \mid -k+2\ell+2\} + 0$  is no better, leading to  $\ell+0 = \ell \geq 1$ .

Our last task is to show that

$$(-3k+3) + \{0 \mid -k+4\} < (-3k+j+2) + \{0 \mid -k+3\}, \quad \text{for } j > 1.$$

This can be established by showing that  $\{j-1 \mid -k+j+2\} + \{k-4 \mid 0\} > 0$ . This prove of this fact is virtually identical to the proof of the similar statement in the preceding paragraph and so it will be omitted.

It now follows that the value of the position  $[[0, 0], [0, 0], [1, k]]$  is

$$\{-3k+3 \mid -3k+3 + \{0 \mid -k+4\}\} = (-3k+3) + \{0 \mid \{0 \mid -(k-4)\}\} = (-3k+3) +_{k-4}.$$

□

In the table below we present, without proof, other interesting game values achievable as bLue/Red Blocking Pebbles positions.

Underlying Digraph	Pebbling Configuration	Game Value
Transitive 3-Cycle	$[[1,0],[2,4],[0,0]]$	$1/4$
Transitive 3-Cycle	$[[3,1],[0,0],[0,1]]$	$1/2$
Transitive 3-Cycle	$[[2,3],[0,0],[1,0]]$	$3/4$
Transitive 3-Cycle	$[[4,4],[0,0],[0,0]]$	$\pm 1/2$
Transitive 3-Cycle	$[[3,5],[0,0],[1,0]]$	$\uparrow *$
Transitive 3-Cycle	$[[3,5],[2,0],[0,0]]$	$\uparrow^{[2]} *$
Directed $P_3$	$[[0,0],[2,2],[0,0]]$	$*2$

We end this section with a short discussion of the differences between BLOCKING PEBBLES and BRG-HACKENBUSH.

As noted above, the blocking mechanic of BLOCKING PEBBLES results in a preponderance of switches, while BRG-HACKENBUSH has no such positions. Also, while we would be surprised to find a dyadic that is not the game value for some BLOCKING PEBBLES position, we have found many dyadic rationals difficult to construct, even with the use of computational methods. BRG-HACKENBUSH positions, on the other hand, are easily constructed that have rational non-integer game values.

### 3.1. Green-only Games

The game of BLOCKING PEBBLES restricted to green pebbles is an impartial game, with positions admitting only numbers as game values. The interested reader will seek out [1,3] for more on Sprague-Grundy Theory and numbers. While there is no use for players to employ a blocking strategy, the game remains mathematically interesting for its connections to its roots in graph pebbling.

First we consider in-stars and out-stars, with green pebble distributions denoted by  $\langle g_0, g_1, \dots, g_n \rangle$  and  $\langle g_0, g_1, \dots, g_n \rangle$  respectively. In each case  $g_i \geq 0$  and  $g_0$  is the number of pebbles on the center vertex.

**Theorem 3.5.** *The value of an in-star with distribution  $\langle g_0, g_1, \dots, g_n \rangle$  is  $*g_0$ .*

*Proof.* We will demonstrate this using induction on  $g_0$ . First note that if  $g_0 = 0$  then any move of a green pebble to the center from a leaf, resulting in the loss of a pebble, can be countered by returning it to the same leaf. Next we note that any move from  $\langle g_0, g_1, \dots, g_n \rangle$  results in a change to  $g_0$ , and that there is a move from this position that results in any number of pebbles on the center node strictly less than  $g_0$ . Hence the in-star is equivalent to a NIM heap of size  $g_0$ .  $\square$

The *nim dimension* of a ruleset is the greatest integer  $k$  where a position in the ruleset has value  $*2^{k-1}$  but no position has value  $*2^k$ . A ruleset in which the nim dimension is unbounded is said to have infinite nim dimension, as Santos and Silva showed is true for KONANE in [6]. Thm. 3.5 implies that GREEN BLOCKING PEBBLES also has infinite nim dimension, whereas the nim dimension of BLUE-RED BLOCKING PEBBLES is still unknown.

The fact revealed in Thm. 3.5 that an in-star is equivalent to a single NIM heap can be generalized to multiple heaps with an out-star.

**Theorem 3.6.** *The value of an out-star with distribution  $\langle g_0, g_1, \dots, g_n \rangle$  is  $*(\bigoplus_{i=1}^n g_i)$ . That is, the nim sum of all heaps.*

*Proof.* We note that this game is analogous to NIM, except instead of removing pebbles from a heap they are moved to the center at no cost. The player with the

advantage simply plays the winning NIM strategy. Any move of a pebble from the center vertex to a leaf can immediately be reversed, at a net cost of one pebble from the center. Thus the number of pebbles at the center do not contribute to the game value, which equals the nim sum of the leaf heaps.  $\square$

On a path we get a similar result.

**Theorem 3.7.** *If  $(g_1, \dots, g_n)$  is a distribution of green pebbles along a path directed left to right, then the game value is  $*$  $(\sum \oplus g_{2k})$ .*

*Proof.* An empty path is trivial, so let us assume the claim is false and consider the set  $C$  of all counter-examples with the fewest total number of pebbles. From  $C$  let  $(g_{0_1}, g_{0_2}, \dots, g_{0_n})$  be the last when ordered lexicographically. Any move from this position either decreases the total number of pebbles, or increases its lexicographic position. Therefore all options of  $(g_{0_1}, g_{0_2}, \dots, g_{0_n})$  are outside  $C$  and hence the claim holds for them. Since each has a digital sum of even terms that differs from  $(\sum \oplus g_{2k})$ , and all smaller sums are realized through NIM moves on the even heaps, we see that  $(g_{0_1}, g_{0_2}, \dots, g_{0_n})$  also satisfies the claim. Therefore,  $C$  is empty and the claim is true.  $\square$

Note that in Theorems 3.5, 3.6, and 3.7, the strategy is equivalent to NIM. In fact, in these particular cases BLOCKING PEBBLES is very similar to the game of POKER NIM, wherein players make NIM moves but retain any removed pebbles, and may add them to a heap instead of removing. While POKER NIM is loopy and BLOCKING PEBBLES is not, both games played optimally have the same strategy and the same reciprocal moves for non-NIM moves.

We now introduce a reduction formula for all trees, which can be applied to the three previous results.

**Construction 3.8.** Let  $T$  be any oriented tree with a given distribution of green pebbles, let  $S$  be its set of source vertices, and let  $O$  be the set of vertices of  $T$  reachable by an odd length directed path from some vertex in  $S$ . Additionally, for a given subset  $W$  of vertices, let  $p(W)$  be the combined total number of pebbles on  $W$ .

We construct the following digraph  $D(T)$  from  $T$  as follows (see Figure 5):

1.  $V(D(T)) = \{\sigma\} \cup O$ , where  $O$  has the same pebbling distribution as it does in  $T$  and  $\sigma$  is a vertex with no pebbles.
2.  $E(D(T)) = E(O) \cup \{\sigma \rightarrow \theta \mid \theta \in O\}$

**Proposition 3.9.** *The game value of BLOCKING PEBBLES on  $T$  is equal to the game value on  $D(T)$ .*

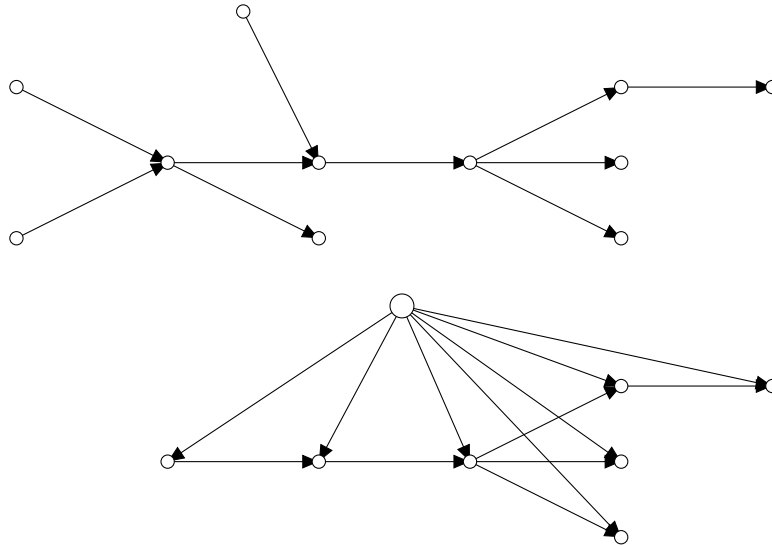


Figure 5: An oriented tree  $T$  and the resulting graph  $D(T)$  from Const. 3.8

*Proof.* The key observation is that the pebbling games on  $T$  and  $D(T)$  are both equivalent to POKER NIM on the set  $O$ . Since the two games have the same set of NIM moves, their game values are equivalent.  $\square$

Applying Const. 3.8 to an in-star results in a single arc, and when  $T$  is a directed path as in Thm. 3.7,  $D(T)$  is simply an out-star. Thus, many tree positions can be reduced to positions on fewer vertices.

It is worth noting, however, that many trees will not reduce to simple positions.

In particular the transitive triple graph, a  $K_3$  oriented without a cycle (Figure 3), has proven very difficult to analyze. However, we present here the set of  $\mathcal{P}$ -positions.

**Theorem 3.10.** *A position in BLOCKING PEBBLES on a transitive triple with  $g_1$  green pebbles on the source vertex,  $g_3$  on the sink, and  $g_2$  on the remaining vertex, is a  $\mathcal{P}$ -position if and only if  $g_2 = g_3$ .*

*Proof.* Note that, as in all other green-only positions, pebbles on the source vertex are superfluous. Since any move that increases the total  $g_2 + g_3$  can be undone, we can consider these heaps as NIM heaps and play accordingly.  $\square$

We close this section with a very simple result, but one that may prove useful in future investigations into the game.

**Theorem 3.11.** *A single green pebble on the sink node of a transitive tournament on  $n$  vertices is equivalent to a NIM heap of size  $n$ .*

*Proof.* We simply consider all options of this position. Since the pebble can only move back, and can move to any previous node, this is equivalent to removing any number of stones from a NIM heap.  $\square$

#### 4. Blue-Red-Green Blocking Pebbles is PSPACE-hard

We next show that it is computationally intractable to determine the outcome class of a general BLUE-RED-GREEN BLOCKING PEBBLES position. More specifically, via a reduction from POSITIVE CNF, we show that BLUE-RED-GREEN BLOCKING PEBBLES is PSPACE-hard.

PSPACE is a class of computational problems that can be solved by a Turing Machine with a polynomial amount of writing space. Problems that are PSPACE-hard are those where the worst-case instances are at least as difficult to compute as the hardest things in PSPACE, modulo a polynomial amount. This hardness can be proven by finding a reduction from an already-known-to-be PSPACE-hard problem to the game in question.

PSPACE-hard problems are widely considered to be intractable because no known algorithm exists to solve them in polynomial-time, despite this being a significant open problem for over 40 years. (Indeed, the Millenium Problem P vs NP is more heavily studied and NP is a subset of PSPACE.) By proving that a ruleset is PSPACE-hard, we are showing that it is not trivial to calculate winning strategies. This means that actual competition is interesting; humans stand a chance against computer players, as no efficient perfect player exists unless  $P = PSPACE$ .

We reduce from POSITIVE CNF to show hardness. POSITIVE CNF is a game played with two players, True and False, with alternating turns, on a list of boolean variables and a CNF (Conjunctive Normal Form) formula using those variables and with no negated literals. In the starting positions, all variables begin unassigned. The True player, on their turn, sets one unassigned variable to true. On the False player's turn, they set one unassigned variable to false. After all variables are set, the True player wins if the CNF evaluates to true, otherwise False wins. POSITIVE CNF is PSPACE-hard<sup>1</sup> [16].

To reduce from POSITIVE CNF to BLOCKING PEBBLES, we need both to have a way for the players to alternate setting variables, and a way for the evaluation of the CNF to determine the winner of the game. We achieve this using 3 different gadgets. We have a gadget for the players to set the variables (the Variable Gadget,

<sup>1</sup>At the original time of submission, POSITIVE CNF was known to be hard for 11-CNF formulas [16]. Since then, an impressive improvement was discovered, showing that it is hard even on 6-CNF formulas [15].



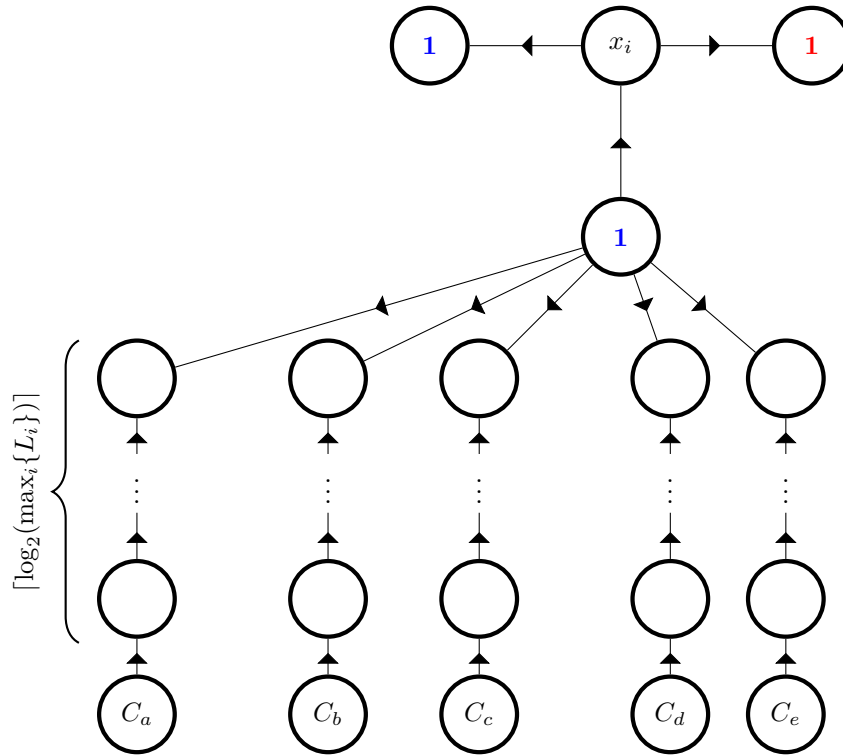


Figure 6: Example of a variable gadget for variable  $x_i$ . Here,  $x_i$  is included in clauses  $C_a, C_b, C_c, C_d,$  and  $C_e$ , a subset of all the clauses.  $L_i$  is the number of unique literals in clause  $C_i$ , so  $\lceil \log_2(\max_i\{L_i\}) \rceil$  is enough that even if a clause gets all of the blue pebbles from the variables, it will not be enough to go back deep into the variable gadget.

Fig. 6), a gadget for the False Player to select a clause they believe they have falsified (the Clause Gadget, Fig. 7), and a gadget that allows the True player to win if no clause is falsified, but allows False to win if they do falsify a clause (the Goal Gadget, Fig. 8).

We consider the following properties of the formula in the POSITIVE CNF position:

- The formula uses  $n$  variables and has  $m$  clauses.
- The  $i$ -th clause contains  $L_i$  unique literals.

The first gadget to describe is the variable gadget. See Fig. 6 for an example. The first pebble moved onto the vertex labelled  $x_i$  corresponds to that player choosing

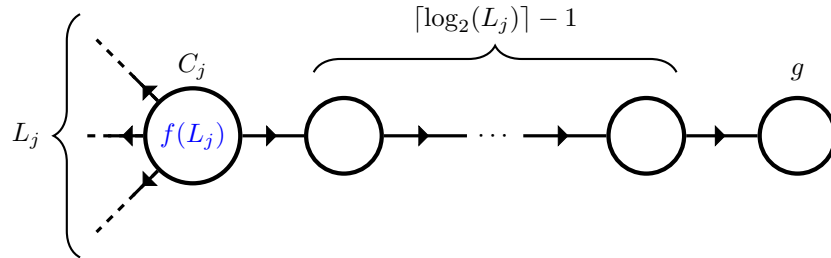


Figure 7: Clause Gadget for a single clause  $C_j$  with  $L_j$  unique literals.  $f(L_j)$  is the number needed to push  $L_j$  to the next power of two, so  $f(L_j) = 2^{\lceil \log_2(L_j) \rceil} - L_j$ .

the variable  $x_i$  in POSITIVE CNF. If Red (True) moves there, then all pebbles in the gadget are unable to move for the remainder of the game. However, if Blue (False) moves there, then they can later move the pebble down with the other pebble, then down one of the paths to a single clause vertex (this will also give 2 moves for Red).

These clause paths are long to prevent players from traversing back "upwards" later with a cache of tokens on the clauses.

Each clause gadget (see Fig. 7) includes a vertex  $C_i$ , which is connected to the  $L_i$  unique literals in that CNF clause via these paths as shown in our variable gadget.  $g$  is part of the goal gadget (Fig. 8). We want to ensure that if and only if Blue moves in each of the variables in  $L_j$ , then they can get exactly one pebble to  $g$ .

We enforce this by requiring Blue to accumulate exactly a power of two ( $2^k$ ) pebbles on the clause vertex  $C_j$  so they can push those pebbles down a path of length  $k$  to reach  $g$ . If there are  $L_j$  literals in clause  $j$ , then the path needs to require  $\lceil \log_2(L_j) \rceil$  moves to reach  $g$ , meaning there needs to be  $\lceil \log_2(L_j) \rceil - 1$  vertices on the path between  $C_j$  and  $g$ . In the many cases where  $L_j$  is not a power of 2, we have to start with extra blue pebbles on the clause vertex to "round it up"; this is just  $f(L_j)$ , where  $f(n) = 2^{\lceil \log_2(n) \rceil} - n$ .

We will give Red (True) a single red pebble on our goal gadget that can traverse a path of length  $n \cdot (5 + \lceil \log_2(\max_i\{L_i\}) \rceil) + 2 \sum_i f(L_i) - n$ . If Blue can not reach  $g$ , then (as we will prove later) Red will win with this path. If Blue can reach  $g$ , however, then they can follow Red down that long path.

If Blue *can* reach  $g$  from one of the clause vertices, they can use the goal gadget to follow Red's single pebble down the path. Red will be forced to traverse down this path before Blue arrives, since Red's only other pebbles are in the variable gadget, where they can move in once or twice, depending on if Blue activates the variable or not. It will take Blue at least two moves for each variable they activate to reach the goal node, so Red's pebble on the goal gadget will never get in the way of Blue's, so long as Blue activates at least as many variables as Red.

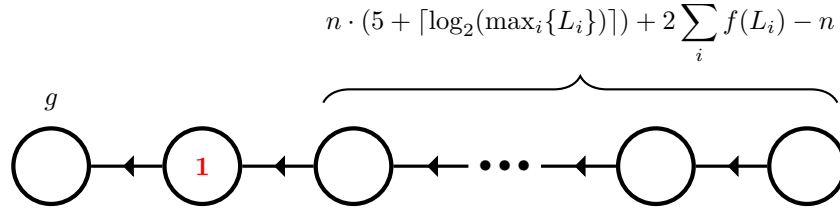


Figure 8: Goal Gadget. This gives Red more moves than Blue if Blue cannot reach vertex  $g$ . If Blue is able to reach  $g$ , then they can follow Red down the path and win.

**Lemma 4.1.** *When Blue makes a move on  $k$  variables gadgets, then no matter which player wins the game, the number of moves Red can make is at least  $n \cdot (5 + \lceil \log_2(\max_i \{L_i\}) \rceil) + 2 \sum_i f(L_i) - n$  and no more than  $n \cdot (5 + \lceil \log_2(\max_i \{L_i\}) \rceil) + 2 \sum_i f(L_i) + k$*

*Proof.* For the first part, clearly Red can always make  $n \cdot (5 + \lceil \log_2(\max_i \{L_i\}) \rceil) + 2 \sum_i f(L_i) - n$  moves by playing on the goal gadget, no matter what else may be happening in the game.

For the upper bound, Red can only move once on variable gadgets they assigned, and twice on variable gadgets that Blue assigned. This will give them up to  $2k + n - k = n + k$  moves total from variable gadgets.

In total, Red has:

$\begin{aligned} \text{Red moves} &= n + k \\ &+ n \cdot \left( 5 + \left\lceil \log_2(\max_i \{L_i\}) \right\rceil \right) + 2 \sum_i f(L_i) - n \\ &= n \cdot \left( 5 + \left\lceil \log_2(\max_i \{L_i\}) \right\rceil \right) + 2 \sum_i f(L_i) + k \end{aligned}$	<p>Variable gadgets</p> <p>Goal gadget.</p>
---	---

□

**Theorem 4.2** (Hardness of BLOCKING PEBBLES). *BLOCKING PEBBLES is PSPACE-hard.*

*Proof.* We complete the proof by showing that the transformation described results in a proper reduction. In other words, that Red/True wins going first in the POSITIVE CNF instance if and only if Red/True wins going first in the resulting BLOCKING PEBBLES position. For notational simplicity, we will let  $X = \lceil \log_2(\max_i \{L_i\}) \rceil$ .

We will refer to *variable assignment* in our reduction: this corresponds to a player moving a pebble of their color onto the corresponding  $x_i$ .

[ $\Rightarrow$ ]: Assume that Red/True wins moving first in the POSITIVE CNF game. Then Red has a strategy to prevent Blue from falsifying all variables in any one clause. Red may follow the strategy of the corresponding game of POSITIVE CNF. Whenever Blue does not make a variable assignment when there are still unassigned variable gadgets remaining, Red may assign a variable arbitrarily. Similarly, when Red's response in the POSITIVE CNF game has already been made, then they may also play arbitrarily. Thus, once all variables are assigned, for every true assignment in the corresponding POSITIVE CNF, there is a red pebble moved on the corresponding variable gadget (and perhaps on other variable gadgets as well). Then, by construction, Blue cannot accumulate enough Blue pebbles on any  $C_i$  vertex of a clause gadget to reach the goal gadget.

On the clause gadgets, without moving any pebble to the goal gadget, Blue may make several moves from a pile of pebbles by repeatedly dropping one pebble to move another pebble down an arc, then moving that pebble back to the pile. This strategy produces  $2(n - 1)$  moves from a pile of  $n \geq 1$  pebbles. Note that we can't get any more moves than this from a clause pile since we are unable to move a pebble to any sufficiently larger sequence of in-neighbor moves. So we obtain an upper bound on the number of moves for Blue on the clause gadgets by assuming all of their blue pebbles are on a single clause vertex:

$$2 \left( n + \sum_i f(L_i) - 1 \right)$$

In total:

Blue moves $\leq n$	claiming variables
$+ n \cdot (X + 2)$	moving claimed variables to clauses
$+ 2 \left( n + \sum_{i=1}^m f(L_i) - 1 \right)$	back-and-forth moves in clauses
$= n \cdot (X + 5) + 2 \sum_{i=1}^m f(L_i) - 2$	
$< n \cdot (X + 5) + 2 \sum_{i=1}^m f(L_i)$	
$\leq$ Red moves	by Lemma 4.1

Thus, Red will win.

[ $\Leftarrow$ ]: We will prove this by contrapositive. Assume that Red/True does not win in the POSITIVE CNF position. Red may make moves corresponding to assigning

variables in the POSITIVE CNF to be True or they may deviate from that. If Red makes a move corresponding to a True assignment, then Blue will respond with the appropriate winning False assignment. If Red makes a move that does not correspond to a True assignment, then Blue can arbitrarily pick a remaining variable, pretend that Red assigned that to True, and respond with the appropriate winning False assignment. If Red ever makes True one of the variables Blue has already pretended they claimed, then Blue will choose yet another remaining variable, pretend Red makes that True, and again choose the appropriate winning response.

Since Blue has a winning strategy in the POSITIVE CNF position, this will result in all the variable gadgets for at least one clause being claimed by Blue pebbles.

Thus, in the BLOCKING PEBBLES board resulting from our transformation, Blue has a strategy to get enough Blue pebbles onto at least one  $C_i$  vertex to have a pebble reach the goal vertex.

Now let us count the number of moves they will have.

$$\begin{aligned}
 \text{Blue moves} &\geq \lfloor n/2 \rfloor && \text{(claiming variables)} \\
 &+ \lfloor n/2 \rfloor \cdot (X + 2) && \text{(moving claimed variables to clauses)} \\
 &+ \left\lceil \log_2(\min_i \{L_i\}) \right\rceil && \text{(move from claimed } C_i \text{ to } g) \\
 &+ n \cdot \left( 5 + \left\lceil \log_2(\max_i \{L_i\}) \right\rceil \right) + 2 \sum_i f(L_i) - n \\
 &&& \text{(moves on the goal gadget)} \\
 &\geq n \cdot \left( 6 + \left\lceil \log_2(\max_i \{L_i\}) \right\rceil \right) + 2 \sum_i f(L_i) \\
 &\quad + \lfloor n/2 \rfloor + \lfloor n/2 \rfloor \cdot X - n \\
 &> n \cdot \left( 6 + \left\lceil \log_2(\max_i \{L_i\}) \right\rceil \right) + 2 \sum_i f(L_i) + n/2 - n \\
 &= n \cdot \left( 5 + \left\lceil \log_2(\max_i \{L_i\}) \right\rceil \right) + 2 \sum_i f(L_i) + n/2 \\
 &\geq \text{Red moves} && \text{(by Lemma 4.1)}
 \end{aligned}$$

Thus, Blue will have more moves than Red and will win in the BLOCKING PEBBLES position.  $\square$

Theorem 4.2 gives us the hardness for the general game on graphs, but, as with many results like this, it is likely not the final word on the matter, for two reasons. First, it is not clear at this point what graph structure(s) for actual play are. So, the range of this reduction may not line up with real-world competition.

Second, it is simultaneously possible that the game cannot be solved in polynomial-space and may be hard for a more difficult complexity class (a superset of PSPACE,

e.g. EXPTIME). One reason for this is that the sizes of the pebble-piles could be exponential in the size of the position description. Additionally, since the graph could contain cycles, the game is loopy, which can often lead to EXPTIME-hardness. Improvements to this result could include:

- Algorithms showing that a player can avoid our constructions from given starting positions (while maintaining the winnability for a player),
- Reductions to more structured graphs than in the range of our reduction, or
- Reductions from supersets of PSPACE.

## 5. Further Directions

There remain many open questions and avenues for further study of BLOCKING PEBBLES. In particular, we would like to resolve the question of game values for all-green games. As we have mentioned, it has proven difficult to determine these values when the underlying graph contains cycles.

Through the use of computational software, in particular CGSuite [17], we have been able to find positions with many dyadic game values. It remains an open question whether or not there is a dyadic rational  $\frac{a}{2^b}$  that is not the value of any position in BLOCKING PEBBLES.

With regards to computational complexity, in addition to the potential improvements listed in that section, the computational hardness of GREEN BLOCKING PEBBLES remains an open problem.

## References

- [1] M. Albert, R. Nowakowski, and D. Wolfe, *Lessons in Play: An Introduction to Combinatorial Game Theory*, CRC Press, 2007.
- [2] M. Albert, J. Grossman, R. Nowakowski, and D. Wolfe, An introduction to clobber, *Integers* **5(2)**, (2005).
- [3] E. Berlekamp, J. Conway, and R. Guy, *Winning Ways for Your Mathematical Plays*, volume 1-4, AK Peters Natick, 2003.
- [4] N. Calkin, K. James, J. Janoski, S. Leggett, B. Richards, N. Sitaraman, and S. Thomas, Computing strategies for graphical nim, *Congr. Numer.* **202** (2010), 171-185.
- [5] F. Chung, Pebbling in hypercubes, *SIAM J. Discrete Math.* **2(4)** (1989), 467-472.
- [6] C. dos Santos and J. Silva, Konane has infinite nim-dimension, *Integers* **8(1)**, #G2, 6pp.
- [7] S. Faridi, S. Huntemann, and R. Nowakowski, Games and complexes i: Transformation via ideals, in *Games of No Chance 5* **70:285** (2019).

- [8] M. Fukuyama, A nim game played on graphs, *Theoret. Comput. Sci.* **1-3(304)** (2003), 387-399.
- [9] G. Hurlbert, A survey of graph pebbling, *Congr. Numer.* **139** (1999), 41-64.
- [10] L. Kirousis and C. Papadimitriou, Searching and pebbling, *Theoret. Comput. Sci.* **47** (1986), 205-218.
- [11] Q. Liu, *Red-Blue and Standard Pebble Games: Complexity and Applications in the Sequential and Parallel Models*, Master's thesis, MIT, Cambridge, MA, 2017.
- [12] R. Milley and G. Renault, Dead ends in misère play: the misère monoid of canonical numbers, *Discrete Math.* **313(20)** (2013), 2223-2231.
- [13] L. Pachter, H. Snevily, and B. Voxman, On pebbling graphs, *Congr. Numer.* **107** (1995), 65-80.
- [14] M. Prudente, *Two-Player Graph Pebbling*, PhD thesis, Lehigh University, Bethlehem, PA, 2015.
- [15] M. Rahman and T. Watson, 6-uniform maker-breaker game is pspace-complete, in *38th International Symposium on Theoretical Aspects of Computer Science (STACS 2021)*, Schloss Dagstuhl-Leibniz-Zentrum für Informatik (2021).
- [16] T. Schaefer, On the complexity of some two-person perfect-information games, *J. Comput. System Sci.* **16(2)** (1978), 185-225.
- [17] A. Siegel, *Combinatorial Game Suite: An open-source program to aid research in combinatorial game theory*, <http://cgsuite.sourceforge.net> (2004).