A generalization of Arc-Kayles

Antoine Dailly, Valentin Gledel, Marc Heinrich

Noname manuscript No.
(will be inserted by the editor)

A generalization of Arc-Kayles

Antoine Dailly · Valentin Gledel · Marc Heinrich

the date of receipt and acceptance should be inserted later

Abstract The game Arc-Kayles is played on an undirected graph with two players taking turns deleting an edge and its endpoints from the graph. We study a generalization of this game, Weighted Arc Kayles (WAK for short), played on graphs with counters on the vertices. The two players alternate choosing an edge and removing one counter on both endpoints. An edge can no longer be selected if any of its endpoints has no counter left. The last player to play a move wins. We give a winning strategy for WAK on trees of depth 2. Moreover, we show that the Grundy values of WAK and Arc-Kayles are unbounded. We also prove a periodicity result on the outcome of WAK when the number of counters is fixed for all the vertices but one. Finally, we show links between this game and a variation of the non-attacking queens game on a chessboard.

Keywords Combinatorial Games · Arc-Kayles · Graphs

Acknowledgements

This work has been supported by the ANR-14-CE25-0006 project of the French National Research Agency.
The authors would like to thank Nicolas Bousquet for his help.

1 Introduction

Combinatorial games are finite two-player games without chance nor hidden information. Combinatorial Game Theory (see [12] for a survey) was developed to analyze games when the winner is determined by the last move. For these games, one of the player has a winning strategy, *i.e.*, one player is guaranteed to win the game, whatever the other player does. It raises three natural questions: which player has a winning strategy? What is this strategy? Can we compute it efficiently?

In 1978, Schaefer [11] introduced several combinatorial games on graphs. Among them is the game Arc-Kayles. In this game, players take turns deleting an edge and its endpoints from the graph, until no edge remain. The winner is the player making the last move. Another way to describe this game is the following: the two players select edges in order to build a maximal matching. The first player's goal is to create a matching of odd size, while the second player tries to make it of even size.

Schaefer introduced Arc-Kayles as a variant of Node-Kayles, which is a game where the players alternate selecting a vertex and deleting it and all its neighbours from a graph. He proved that Node-Kayles is PSPACE-complete. This game has then been studied on specific graph classes: it has been proven that deciding its outcome is polynomial when playing on graphs with bounded asteroidal number, cocomparability graphs, cographs [4] and bounded degree stars [5]. A derived game called Grim has also been studied in [1]. In this variant, the neighbours of the selected vertex are deleted from the graph if and only if they become isolated.

Antoine Dailly · Valentin Gledel · Marc Heinrich
Univ Lyon, Université Lyon 1, LIRIS UMR CNRS 5205, F-69621, Lyon, France.

It is still an open question whether the problem of deciding which player has a winning strategy for Arc-Kayles is PSPACE-complete or not, and very few results, either general or on specific graph classes, are known. Played on a path, it is equivalent to the octal game 0.07, also called Dawson's Kayles, solved in [6]. More recently, some results have been found for specific classes of graphs: cycles, wheels and subdivided stars with three paths [14]. It was also shown in [8] that the problem is FPT when parameterized by the number of rounds, meaning it can be solved in time $O(f(k) + \text{poly}(n))$ where k is the number of rounds of the game, n is the number of vertices and f is some computable function.

We study a generalization of Arc-Kayles, called Weighted-Arc-Kayles (WAK for short). The game WAK is played on an undirected graph with counters on the vertices. The players take turns selecting an edge, and removing one counter from each of its endpoints. If an edge has an endpoint with no counter left, then it cannot be selected anymore. The game ends when no edge can be selected anymore. When there is only one counter on each vertex, this game is exactly Arc-Kayles. This game was first proposed by Huggan in [7], along with several others, as a possible extension of Arc-Kayles. Our study of WAK is also motivated by a variation of the non-attacking queens game [9]. Consider the game where players alternately place non-attacking rooks on a not necessarily square chessboard. We show that this game can be represented as an instance of WAK.

In order to simplify the study of some graphs, we allow the vertices to have loops. In this paper, we prove that one can decide in polynomial time which player has a winning strategy on loopless trees of depth at most 2 (we consider that a tree reduced to a single vertex has depth 0). This directly solves a particular case of the non-attacking rooks game.

Theorem 1. *There is a polynomial time algorithm computing the outcome of* WAK *on any loopless tree of depth at most* 2*.*

The outcome of WAK on a loopless tree of depth at most 2 is determined by the parity of the sum of the numbers of counters of some sets of vertices, and by inequalities between them. This is not the case with C_3, the cycle on three vertices, which suggests that it might be harder to characterize outcomes for graphs with induced cycles, or at least non-bipartite graphs.

Grundy values are a tool used in Combinatorial Game Theory to refine the question of which of the two players wins (a more formal definition is given in Section 2). The Grundy values of Arc-Kayles (and by extension of Node-Kayles) were conjectured to be unbounded in [14]. We give a positive answer to this conjecture as a corollary of the following:

Theorem 2. *The Grundy values for the game* WAK *are unbounded.*

The paper is organized as follows: in Section 2, we give basic definitions and formally define WAK. Section 3 shows the links between WAK and the game of placing non-attacking rooks on a chessboard. In Section 4, we define the core concept of *canonical graphs*. This notion is used to prove a relation between WAK and Arc-Kayles. It also simplifies the study of graphs in Section 5, where we characterize which player has a winning strategy for WAK on loopless trees of depth at most 2. This characterization only depends on the parity of the weights, and some inequalities between them. Next, we present in Section 6 a periodicity result on the outcomes of WAK positions when the number of counters is fixed for all but one vertex. Finally, we prove in Section 7 that the Grundy values of WAK and Arc-Kayles are unbounded.

2 Definitions and notations

2.1 Combinatorial Game Theory

We will give basic definitions of Combinatorial Game Theory that will be used in the paper. For more details, the interested reader can refer to [2, 3, 12]. *Combinatorial games* [3] are two-player games where:

- the players play alternately;
- there is no chance, nor hidden information;
- the game is finite;
- the winner is determined by the last move alone.

In this paper, the games are *impartial*, i.e., both players have exactly the same set of available moves on any position. The only difference between the two players is who plays the first move. Every position G

of a combinatorial game can be viewed as a combinatorial game with G as the initial position. By abuse of notation, we will often consider positions as games.

From a given position G, the positions that can be reached from G by playing a move are called the *options* of G. The set of the options of G is denoted opt(G). A position of an impartial game can have exactly two *outcomes*: either the first player has a winning strategy and it is called an $\mathcal{N}$*-position* (for "$\mathcal{N}$ext player win"), or the second player has a winning strategy and it is called a $\mathcal{P}$*-position* (for "$\mathcal{P}$revious player win"). The outcome of a position G can be computed recursively from the outcome of its options using the following characterization:

Proposition 3 ([2]). *Let G be a position of an impartial game in normal play.*
- *If $opt(G) = \emptyset$, then G is a $\mathcal{P}$-position;*
- *If there exists a position $G' \in opt(G)$ such that G' is a $\mathcal{P}$-position, then G is an $\mathcal{N}$-position and a winning move is to play from G to G';*
- *If every option of G is an $\mathcal{N}$-position, then G is a $\mathcal{P}$-position.*

Although not classical, we define a relation between games based on their outcomes: if two games G_1 and G_2 have the same outcome, they are *outcome-equivalent*, and we write $G_1 \sim G_2$.

Given two games G_1 and G_2, we define their *disjoint sum*, denoted $G_1 + G_2$, as the game where, at their turn, the players play a legal move on either G_1 or G_2 until both games are finished. The player making the last move wins. If G_1 is a $\mathcal{P}$-position, then $G_1 + G_2$ has the same outcome as G_2. Indeed, the player with a winning strategy on G_2 can apply this strategy and reply to any move made by his opponent on G_1 using the second player's winning strategy on G_1. In order to study the outcome of $G_1 + G_2$ in the case where both G_1 and G_2 are $\mathcal{N}$-positions, we refine the outcome-equivalence by a relation called the *Grundy-equivalence*. Two games G_1 and G_2 are *Grundy-equivalent*, denoted by $G_1 \equiv G_2$, if and only if for any game G, $G_1 + G$ and $G_2 + G$ have the same outcome, i.e., $G_1 + G \sim G_2 + G$. In particular, if $G_1 \equiv G_2$, then $G_1 + G_2$ is a $\mathcal{P}$-position.

We can attribute a value to a game according to its Grundy equivalence class, called the *Grundy value*. The Grundy value of a game position G, denoted $\mathcal{G}(G)$ can be computed using the Grundy values of its options thanks to the following formula:

$$\mathcal{G}(G) = mex(\mathcal{G}(G')|G' \in opt(G))$$

where, given a finite set of nonnegative integers S, $mex(S)$ is the smallest nonnegative integer not in S. In particular, a position G is a $\mathcal{P}$-position if and only if $\mathcal{G}(G) = 0$, which is consistent with Proposition 3.

One of the most fundamental results in Combinatorial Game Theory is the Sprague-Grundy Theorem, which gives the Grundy value of the disjoint sum of two impartial games:

Theorem 4 (Sprague-Grundy Theorem [13]). *Given two impartial games G_1 and G_2, we have*

$$\mathcal{G}(G_1 + G_2) = \mathcal{G}(G_1) \oplus \mathcal{G}(G_2),$$

where $\oplus$, called the Nim-sum, is the bitwise XOR.

2.2 Definition of WAK and notations

In the whole paper we consider weighted graphs $G = (V, E, \omega)$, where V is the set of vertices, E is the set of edges, and $\omega : V \to \mathbb{N}^*$ gives the number of counters on each vertex. Graphs are undirected. There may be loops, in which case an edge looping on a vertex u is denoted by (u, u). For every vertex u, loop(u) is a boolean with value `True` if and only if the edge $(u, u) \in E$, in which case we say that a loop is *attached to* u.

At each turn, the current player selects an edge and removes one counter from both of its endpoints (or its unique endpoint, if the edge is a loop). For any vertex u such that $\omega(u) = 0$, the edges (u, v) cannot be selected anymore. The game then continues until no edge can be selected anymore. Figure 1 shows an example of the moves available from a given position. For simplicity and if confusion is not possible, vertices on caption of figures will be named after their number of counters.

We consider that, when selecting a loop, one counter is removed from its unique endpoint. This convention is defined since we introduce loops to simplify the study of several graphs. The other possible convention, which would remove two counters from the endpoint, could also be considered.

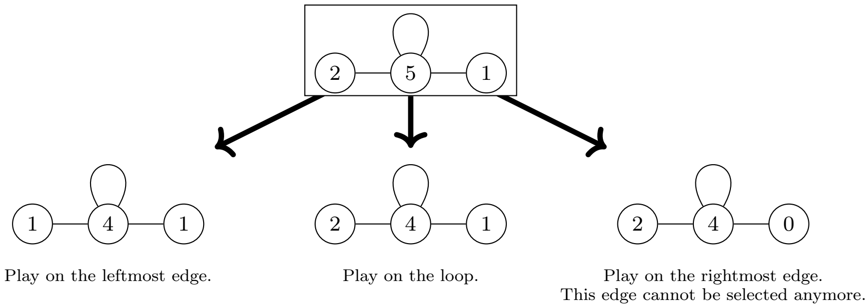


Fig. 1 Example of possible moves for Weighted Arc Kayles on a given position.

If the number of counters of one vertex reaches zero, then the edges adjacent to this vertex cannot be played anymore and the vertex can be removed from the graph. Note that if the graph G is not connected, then WAK played on G is equivalent to the disjoint sum of the connected components of G. Consequently, if we can compute the Grundy values of each of the connected components, we can use Theorem 4 and get the Grundy value of G.

Observation 5. *Let G be a non-connected graph, and $G_1, \ldots, G_k$ its connected components. We have*

$$\mathcal{G}(G) = \mathcal{G}\big(\sum_{i=1}^{k} G_i\big) = \bigoplus_{i=1}^{k} \mathcal{G}(G_i).$$

3 Relation with non-attacking rooks on a chessboard

Inspired by the non-attacking queens game [9], we introduce the non-attacking rooks game. It is played on an $n \times m$ chessboard $\mathcal{C}$. There is a subset $\mathcal{H}$ of the squares of the chessboard whose elements are named *holes*. At each turn, the current player places a rook on a square of the chessboard that is not a hole in such a way that it does not attack any of the already played rooks. The rooks cannot 'jump over' holes. In other words, there can be two rooks on the same row provided there is a hole between them. The first player unable to play loses.

When the chessboard has no holes, the game is fully characterized by the parity of the minimum dimension of the grid. Indeed, at each turn a row and a column are deleted from the chessboard, and the game ends when there is no more row or column to play on. An example of such a game can be seen on Figure 2.

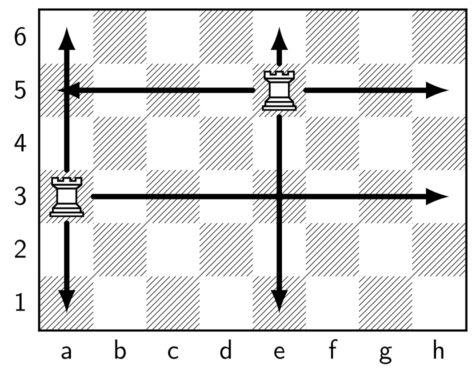


Fig. 2 On a 6×8 chessboard without holes the game always end after six moves and the second player wins.

We prove that the non-attacking rooks game can be viewed a special case of WAK, played on a certain graph.

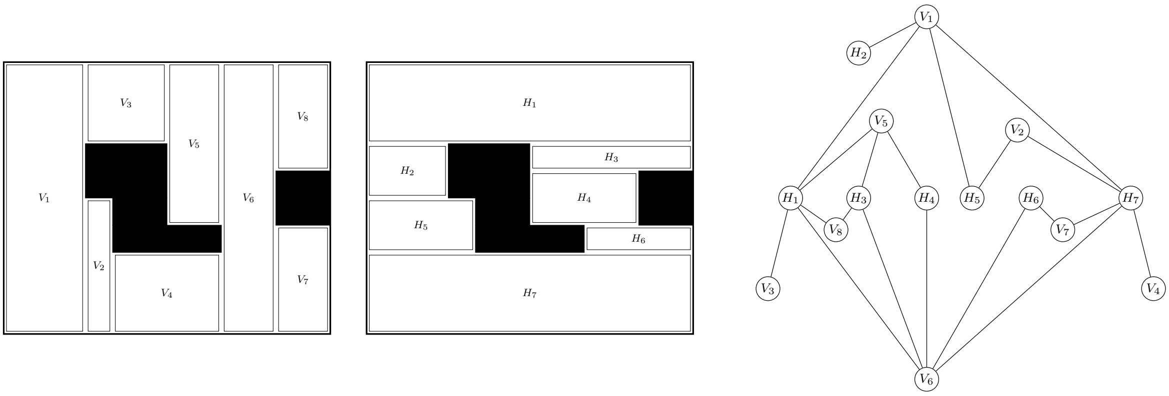


Fig. 3 Vertical and horizontal rectangles cover and the graph associated.

Proposition 6. *Every position of the non-attacking rooks game has an equivalent position in* WAK.

Proof. We define a *vertical rectangle cover* of the chessboard as a set $\mathcal{R}_V$ of rectangles on the chessboard, such that no two rectangles intersect, no rectangle contains holes, the union of the rectangles contains all of the squares of the chessboard that are not holes, and all the squares directly above or below each rectangle are either holes or outside the board.

We can similarly define a *horizontal rectangle cover* where the squares at the left and right border are either holes or outside the chessboard. An example of such covers can be seen on Figure 3. We can always find such rectangle covers, for example by taking pieces of rows or columns between the holes.

Consider a position of the non-attacking rooks game, and let $\mathcal{R}_V$ and $\mathcal{R}_H$ be respectively a vertical and horizontal rectangle cover of the board. Let $G = (V, E, \omega)$ be the weighted graph built as follows:

- for each vertical rectangle V_i, add a vertex v_i with weight the number of columns of V_i minus the number of rooks already present in V_i,
- for each horizontal rectangle H_j, add a vertex h_j with weight the number of rows of H_j minus the number of rooks in H_j,
- if two rectangles V_i and H_j intersect, then add an edge between v_i and h_j.

Each time a player places a rook on the chessboard, it is inside exactly one vertical rectangle V_i and one horizontal rectangle H_j. Since it is forbidden to attack this rook, it is equivalent to remove a column from V_i and a line from H_j. So the graph obtained from this new position is the one in which the weights of v_i and h_j are decreased by one. The position of the rook in the intersection of V_i and H_j does not matter.

Moreover, in a graph G associated with a position of the non-attacking rooks game, for each move on an edge (v_i, h_j), the weights of the vertices are positives so the corresponding rectangles have a positive number of line or columns. Moreover, since the edge (v_i, h_j) exists, the rectangles intersect. By construction, the intersection of V_i and H_j is a rectangle with the same number of columns as V_i, and the same number of rows as H_j. Additionally, there are exactly $\omega(v_i)$ free columns, i.e., not occupied by previously played rooks, and $\omega(h_j)$ free rows. Consequently, it is possible to place a rook in the intersection of V_i and H_j.

Denote by f the construction above that transforms a position G for the rook placement game into a position $f(G)$ for WAK. For each move from G to G' for the non-attacking rooks game, there is an equivalent move from $f(G)$ to $f(G')$. Conversely, for every move from $f(G)$ to some position G_1 of WAK, there is an equivalent move from G to some position $G' \in f^{-1}(G_1)$. Consequently, the two games are equivalent.

□

From the proof of the proposition above, we can see that the exact square on which a rook is placed is not important. Indeed, what really matters is in which area it is placed. For example, in the case of a rectangular hole by the edge of the chessboard, as in Figure 4, there are 5 different rectangular areas: top left, top right, bottom left, bottom right, and bottom center. Consequently, there are only 5 types of moves in this case, and the only thing that a player has to choose is in which area he will play. In this

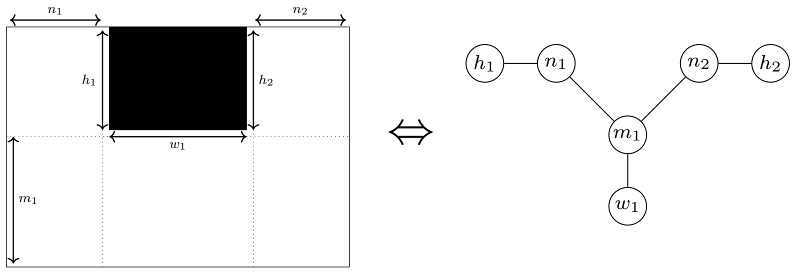


Fig. 4 Reduction when there is a hole by the edge of the chessboard.

case, the equivalent WAK position is a tree of depth 2, which is solved by Theorem 1. The more general case with a hole in the middle of the board seems more complicated.

The rest of paper will now be dedicated to study properties of WAK.

4 Canonical Graphs

The notion of canonical graphs comes from the observation that some vertices have little influence on the outcome of the game. The following definition identifies these vertices. Given a graph $G(V, E, \omega)$ and a vertex $u \in V$, we define the *neighbourhood* of u, denoted by $N(u)$, as the set $\{v \in V | (u, v) \in E\}$.

Definition 1. *Let $G = (V, E, \omega)$ be a weighted graph. We define the following:*
- *A vertex $u \in V$ is* useless *if u has no loop attached to it, and all the neighbors of u have a loop attached to them.*
- *A vertex $u \in V$ is* heavy *if u has no loop attached to it, and $\omega(u) \geq \sum_{v \in N(u)} \omega(v)$.*
- *Two non-adjacent vertices u and v are* false twins *if $N(u) = N(v)$, and $\text{loop}(u) = \text{loop}(v)$.*

The idea is that whenever there is a useless vertex, a heavy vertex, or two false twins, the graph can be simplified. This simplification is described in the definition below. It is illustrated on Figure 5.

Definition 2. *Let $G = (V, E, \omega)$ be a weighted graph. A* reduction *of G is a graph G' obtained by applying any arbitrary sequence of the following steps:*
- *Deleting a useless vertex u;*
- *Deleting a heavy vertex u and attaching a loop to each of its neighbors;*
- *Merging two false twins v_1 and v_2 into a single vertex v with weight $\omega(v_1) + \omega(v_2)$.*

A graph is *canonical* if it has no useless vertices, no heavy vertices, and no false twins. The following lemma ensures that the reduction of the graph preserves its Grundy value (and thus its outcome).

Proposition 7. *If G is a graph and G' is a reduction of G, then $G \equiv G'$.*

Proof. We will prove that the none of the three reduction operations changes the Grundy value of the graph. Let G be a weighted graph, and G' be a graph obtained by applying only one reduction operation on G. The proof is by induction on the total sum of all the weights of G. If all the weights of G are zero, then clearly their is no move on either G or G', and both games have Grundy value 0.

Suppose by induction that the property holds for all graphs G with total weight at most k. Let $G = (V, E, \omega)$ be a graph with total weight $k + 1$, and let G' be a graph obtained from G by applying one step of the reduction rules.

In the following, given an edge e of G (resp. G') such that both its endpoints have positive weights, we denote by G_e (resp. G'_e) the graph obtained from G (resp. G') by playing e. We will prove the two following points:

(a) For every option G_e of G, there is an option G'_e of G' such that $G_e \equiv G'_e$.
(b) For every option G'_e of G', there is an option G_e of G such that $G_e \equiv G'_e$.

By definition of the grundy value as the mex of the options' values, these two properties imply that $G \equiv G'$.

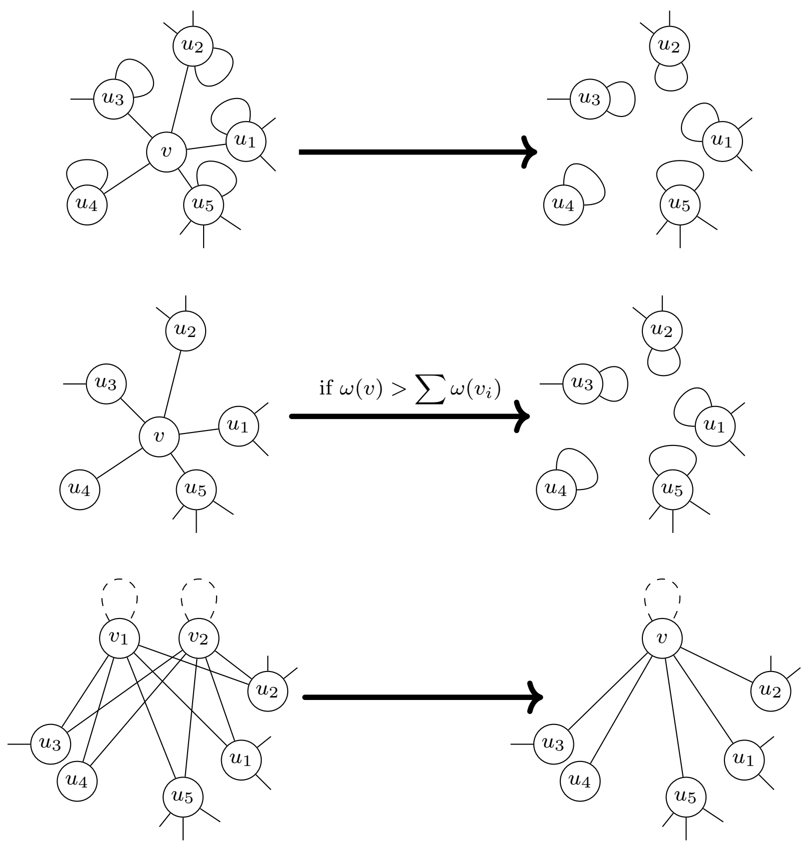


Fig. 5 The three possible reductions for a graph. At the top, v is a useless vertex. In the middle, v is a heavy vertex with $\omega(v) \geq \sum \omega(u_i)$. At the bottom, two false twins are merged, and we have $\omega(v) = \omega(v_1) + \omega(v_2)$, and $\text{loop}(v) = \text{loop}(v_1) = \text{loop}(v_2)$.

First, consider the case where e is an edge in both G and G'. Then we have $G_e \equiv G'_e$. Indeed, suppose that we can apply a certain reduction operation on G, then the same reduction can be applied on G_e since:

- a vertex that is useless in G is also useless in G_e;
- a vertex v that is heavy in G is also heavy in G_e, since any move that decrease the weight of v also decrease the weight of one of its neighbors;
- if two vertices are false twins in G, then they are also false twins in G'.

Hence, we can consider the graph obtained from G_e by applying the same reduction operation as in G. Since e is also an edge of G', we can check easily that this graph is equal to G'_e. The total weight of G_e is at most k, and by applying the induction hypothesis on G_e, we obtain $G_e \equiv G'_e$.

Consequently, we only need to consider the options G_e and $G'_{e'}$ where e is not an edge of G', and e' is not an edge of G. For each of the three possible reduction rules, we will check that the two properties hold for these remaining options:

Case 1: G' is obtained from G by deleting v, a useless vertex.

Let e be an edge of G, and consider the option G_e of G. By the observation above, we can assume that e is not an edge of G', and consequently, we can write $e = (v, u)$ for some vertex $u \in N(v)$. Since v is a useless vertex, we have $u \neq v$, and there is a loop on u. Let e' be the edge (u, u) in G'. Then $G'_{e'}$ is equal to the graph obtained from G_e by removing the (useless) vertex v, and by applying the induction hypothesis on G_e, we have $G_e \equiv G'_{e'}$.

If $G'_{e'}$ is an option of G', then e' is also an edge of G, and this case is already handled by the observation above.

Case 2: G' is obtained from G by deleting a heavy vertex v, and attaching loops to its neighbors.

Let e be an edge of G, and consider the option G_e of G obtained by playing e. We can assume that e is not an edge of G', and consequently e is incident to v and we can write $e = (v, u)$ for some vertex $u \in N(v)$. Let e' be the edge (u, u) of G'. Then $G'_{e'}$ is the graph obtained from G_e by simplifying the heavy vertex v, and using the induction hypothesis on G_e, we have $G_e \equiv G'_{e'}$.

Consider now an option $G'_{e'}$ of G' for some edge e' of G'. We can assume that e' is not an edge of G, and consequently $e' = (u, u)$ for a vertex u adjacent to v in G. Denote by e the edge (u, v) in G. Since playing e' is a valid move in G', we have that $\omega(u) > 0$, and since v is a heavy vertex, $\omega(v) \geq \omega(u) > 0$. Consequently, playing e on G is also a valid move. Additionally, v is still a heavy vertex of G_e, and $G'_{e'}$ is equal to the graph G_e after the reduction of the heavy vertex v. Using the induction hypothesis on G_e, we have that $G'_{e'} \equiv G_e$.

Case 3: G' is obtained from G by merging two false twins v_1 and v_2 into a single vertex v.

First consider an option G_e of G for a certain edge e. We only need to consider the cases where e is not in G'. If $e = (v_1, u)$, with u a neighbor of v_1, consider the edge $e' = (v, u)$ of G'. Since playing e on G is a valid move, we have $\omega(u) > 0$ and $\omega(v_1) > 0$. By definition of the twin vertices reduction, we have $\omega(v) = \omega(v_1) + \omega(v_2) > 0$, hence playing e' in G' is a valid move. Additionally, we can check that $G'_{e'}$ is equal to the graph G_e after merging the two twin vertices v_1 and v_2. As a consequence, using the induction hypothesis we have $G_e \equiv G'_{e'}$. If e is the loop attached to v_1 in G, then by taking e' the loop attached to v in G', and using a similar argument, we can show that $G_e \equiv G'_{e'}$.

Conversely, consider the option $G'_{e'}$ of G' for some edge e'. We can assume that e' is not an edge of G, and consequently $e = (v, u)$ for some vertex u. Since playing e' on G' is a valid move, one of v_1 or v_2 has a positive weight. Without loss of generality, assume $\omega(v_1) > 0$. If $u \neq v$, then take $e = (v_1, u)$, otherwise, if e' is the loop attached to v, take e to be the loop attached to v_1. Then we can easily check that in both cases, playing e on G is a valid move. Additionally, $G'_{e'}$ is equal to the graph G_e after reducing the two false twin vertices v_1 and v_2. By applying the induction hypothesis on G_e, we get that $G_e \equiv G'_{e'}$.

Hence in all three cases, the properties holds, and consequently we have $G \equiv G'$. This ends the induction step and proves the proposition. □

If a graph G is not canonical, we can take a canonical reduction G' of G. By Proposition 7, G' has the same Grundy value as G, and we can study G' instead. This allows to simplify the study of G in many cases. In particular, Proposition 7 gives a straightforward solution when G is a star. Indeed, in this case, all the leaves are false twins and can be merged together without changing the Grundy value. The resulting graph only contains two adjacent vertices without loops.

Another simple consequence of Proposition 7 is the following result:

Corollary 8. *Let $G = (V, E, \omega)$ be a weighted graph. There is a graph G' such that the Grundy value of* WAK *on G is the same as the Grundy value of* Arc-Kayles *on G'.*

Proof. The graph G' is constructed as follows: for every vertex u with weight $\omega(u) > 1$, replace u by $\omega(u)$ vertices, each with weight 1, and each with the same neighbors as u. Then, for every vertex u such that there is a loop attached to u, remove the loop and create a new vertex u' with weight 1 adjacent to u. We can remark that G' is obtained by applying on each vertex the inverse of the simplification procedure for false twins and heavy vertices. Indeed, each vertex u is split into $\omega(u)$ false twins, and each vertex u' created from a loop is heavy. Removing the heavy vertices, and merging back all the false twins by applying the simplification procedure gives back the graph G.

As a consequence, G is a reduction of G' obtained by merging the false twins and removing the heavy vertices that we created. Using Proposition 7, G and G' have the same Grundy values. Since there is no loop in G', and all the vertices of G' have weight 1, WAK played on G' is just an instance of Arc-Kayles. □

The reduction from WAK to Arc-Kayles is not polynomial. Indeed, a vertex v is transformed into a number $\omega(v)$ of vertices, which is exponential in the size of the binary representation of $\omega(v)$.

5 Trees of depth at most 2

In this section, we will give a characterization of the outcome for WAK when the graph is a tree with depth at most 2. We begin by analyzing simple cases before moving on to more complicated ones. Since no confusion is possible, the vertices will be named after their weights.

Given an unweighted graph G, the *set of positions* for G is the set of all possible weight functions ω, denoted $\mathrm{pos}(G)$. Given an order $v_1, \dots, v_n$ on the vertices of G, a specific weight function ω will be denoted by the tuple $(\omega(v_1), \dots, \omega(v_n))$.

Lemma 9. *We have* $\mathcal{G}((a)) = a \bmod 2$.

Proof. The only available move is to play the loop. This decreases the number of counters on the vertex by 1, until it reaches 0. The result then holds by induction. □

Lemma 10. *The Grundy value of* $(a)\!-\!(b)$ *is given by the formula:*

$$\mathcal{G}((a)\!-\!(b)) = ((a+b) \bmod 2) + 2 \times (\min(a,b) \bmod 2)$$

This is summarized in the table below where $m = min(a,b)$ and $M = max(a,b)$:

m \ M	even	odd
even	0	1
odd	3	2

Proof. If $a = 0$ or $b = 0$, the vertex with weight zero can be removed from the graph. The resulting graph is just composed of a single loop, and the result follows from Lemma 9.

Let $a, b > 0$. The Grundy values can be determined by induction as shown in Figure 6.

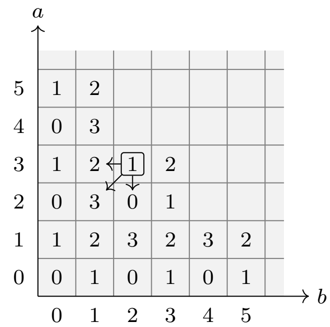


Fig. 6 Table of grundy values for the graph $(a)\!-\!(b)$ with different values for a and b.

□

If a graph is not connected, then by Observation 5, the Grundy value can be computed from the values of the connected components. Since the graph in the following remark occurs several times in later proofs, we give an explicit characterization of its $\mathcal{P}$-positions. It is obtained from the two previous results.

Remark 11. *The graph* $(a) + (b)\!-\!(c)$ *is a* $\mathcal{P}$*-position if and only if one of the two following holds:*

- *$a, b,$ and c are even;*
- *a and $\max(b,c)$ are odd, and $\min(b,c)$ is even.*

The two following proofs use the same argument. The idea is the following: for a fixed graph G, there are some ranges on the weights of the vertices for which the graph is not canonical, and by applying a reduction, the outcome of the graph can be computed by induction on the size of the graph. When the graph is canonical, we prove that a certain set $P \subseteq \mathrm{pos}(G)$ is the set of $\mathcal{P}$-position. This argument is formalized in the following proposition.

Proposition 12. *Let G be a graph, and $S \subseteq \text{pos}(G)$ such that there is no move from a position not in S to a position in S. Let P be a subset of S, and assume that:*

(i) There is no move from a position in P to another position in P;
(ii) From a position in P, any move to some position not in S is a losing move.
(iii) From any position in $S \setminus P$, there is either a move to a position in P, or to a $\mathcal{P}$-position s' not in S;

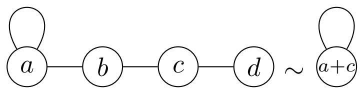
Under these assumptions, a position p in S is a $\mathcal{P}$-position if and only if p is in P.

Proof. The proof is by induction on the sum of the weights. Let p be a position for G such that $p \in S$. If there is no possible move from p, then p is a $\mathcal{P}$-position and $p \in P$, by condition (iii). Thus we can suppose that p has a nonempty set of options.

If $p \notin P$, then by condition (iii), there is a move from p to a position p' such that either $p' \notin S$, and p' is a $\mathcal{P}$-position, or $p' \in P$, and p' is a $\mathcal{P}$-position by applying the induction hypothesis on p'. In both cases, p' is a $\mathcal{P}$-position, and consequently p is an $\mathcal{N}$-position.

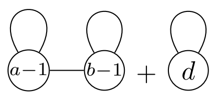
Suppose now that $p \in P$, and let $p' \in opt(p)$. By condition (i), we know that $p' \notin P$. If $p' \in S$, then p' is an $\mathcal{N}$-position by induction hypothesis. If $p' \notin S$, then using condition (ii), p' is also an $\mathcal{N}$-position. Consequently, p is a $\mathcal{P}$-position. □

Lemma 13. *Suppose that the graphs on the left is canonical, then the following outcome-equivalences holds:*



Proof. Let G be the graph on the left in the proposition. Let $S \subset \text{pos}(G)$ be the set of positions satisfying $c < b + d$, and $P \subset S$ be the subset of positions for which $a + c$ is even. Note that if a position s is canonical , then $s \in S$. We want to show that S and P satisfy the three conditions of Proposition 12. Since any move decreases $a + c$ by exactly 1, there is no move from a position in P to another position in P, and point (i) holds.

If s is a position in P such that there is a move from s to a position $s' \notin S$, then necessarily s' is obtained by playing the edge (a, b). After the move, the vertex c becomes a heavy vertex, and consequently s' can be simplified to:



Since $s \in P$ and $s' \notin S$, we know that $c = b+d-1$, and $a+c$ is even. Consequently, $a-1+b-1+d = a + c - 1$ is odd, which implies by Remark 11 that s' is an $\mathcal{N}$-position, and the point (ii) also holds.

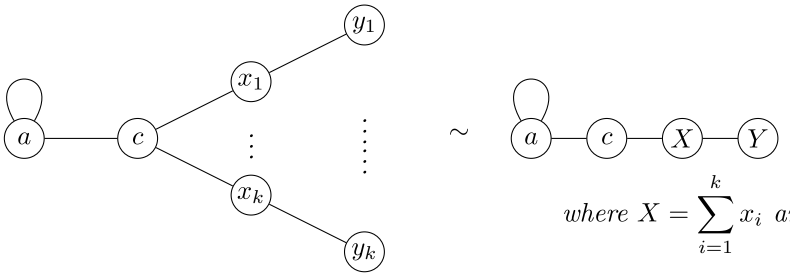
Finally, let $s = (a, b, c, d)$ be a position such that $s \in S \setminus P$. Then one of a or c is odd, and thus non-zero. Consequently, there is a move from s to a position s' by playing either the loop attached to a, or one of the edges (b, c) or (c, d). Since none of these move decrease the quantity $c - b - d$, we have $s' \in P$. So point (iii) holds.

The three conditions of Proposition 12 are satisfied, and the result follows. □

The following Lemma is the key technical result that allows us to prove Theorem 1. Using this result, the proof of the theorem will follow the following ideas. Given a tree T of depth at most 2, we can compute a more simple reduced graph, which has the same outcome as T. The reduced graph is a path on four vertices with a loop on one end. The outcome of the reduced graph can be computed either with the characterization of Lemma 13 if it is canonical, or by reducing the graph further to smaller components.

Note that in the statement of the Lemma, the graph on the left needs not be canonical. In particular, we may have $c \geq a+\sum_{i=1}^{k} x_i$. The proof of the result is a bit technical but presents no theoretical difficulties. We simply check that the three conditions of Proposition 12 are satisfied, and proceed by case analysis.

Lemma 14. *Let $k > 0$, and $x_1, \dots, x_k$ and $y_1, \dots, y_k$ be nonnegative integers such that $x_i > y_i$. Then the following holds*



$$\text{where } X = \sum_{i=1}^{k} x_i \text{ and } Y = \sum_{i=1}^{k} y_i.$$

Proof. Denote by G_1, G_2, G_3, G_4 the graphs shown on Figure 7. On Figure 7, the equivalence $G_2 \sim G_4$ if $X \geq Y + c$ is obtained by simplifying the heavy vertex X in G_2. If $X < Y + c$, then either $c < a + X$, and in this case G_2 is canonical, and the outcome equivalence $G_2 \sim G_3$ is obtained by Lemma 13, or $c \geq a + X$, and then the vertex c is heavy. In this case, the outcome equivalence $G_2 \sim G_3$ is obtained by simplifying the vertex c, and then removing the now useless vertex Y.

On the figure, the edges of these four graphs are marked with labels. Given an edge e of G_1, we will denote by $f(e)$ the edge of G_2 with the same label as e. Additionally, if $s = (a, c, x_1, \ldots x_k, y_1, \ldots y_k)$ is a position of G_1, we will also denote by $f(s)$ the position (a, c, X, Y) of G_2 where $X = \sum_{i=1}^{k} x_i$ and $Y = \sum_{i=1}^{k} y_i$. With these notations, we can easily check that for any two positions s and s' of G_1 such that there is a move from s to s', there is a move in G_2 from $f(s)$ to $f(s')$ by playing $f(e)$.

If either X or Y is equal to zero, then the outcome equivalence holds easily. Consequently, we will assume in the following $X > 0$ and $Y > 0$. We will prove the outcome-equivalence by induction on k. Clearly, the result holds when $k = 1$ since in this case both sides are the same. Consequently, we suppose $k > 1$, and assume that the results holds when there are less than k branches.

Let S be the set of positions $(a, c, x_1, \ldots x_k, y_1, \ldots y_k)$ of G_1 such that for all i, we have $x_i > y_i$. Let P be the subset of positions $s \in S$ such that $f(s)$ is a $\mathcal{P}$-position of G_2. As previously, we will show that S and P satisfy the three conditions of Proposition 12.

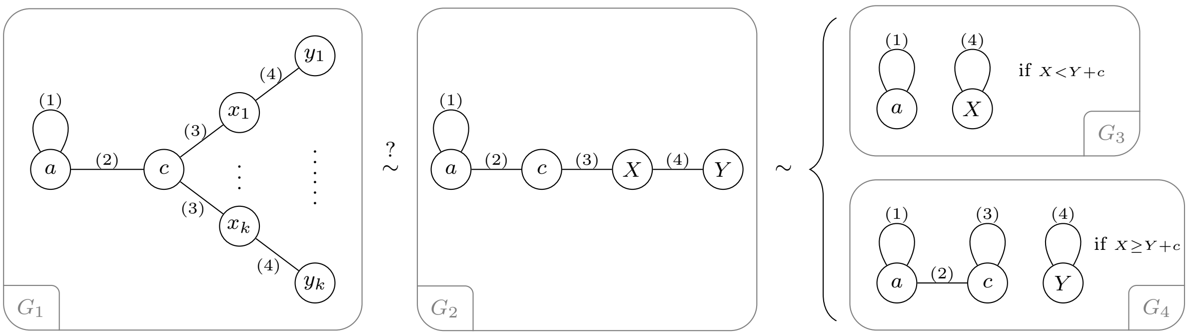
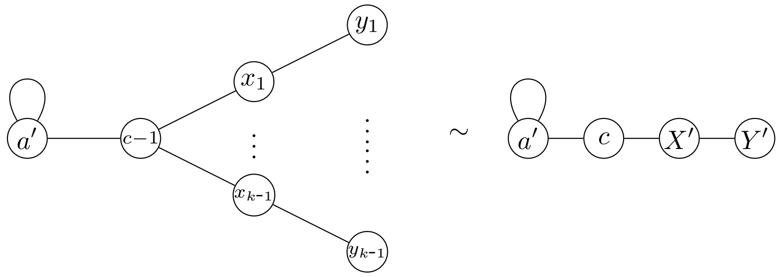


Fig. 7 With $X = \sum_{i=1}^{k} x_i$ and $Y = \sum_{i=1}^{k} y_i$. The corresponding edges between the graphs are marked with the same labels.

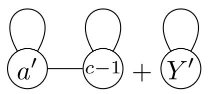
(i) If s and s' are two positions in P, then there is no move from s to s'. Indeed, suppose by contradiction that there is a move from s to s' by playing an edge e. Then $f(s)$ and $f(s')$ are both $\mathcal{P}$-positions of G_2 by definition of P, and there is a move from $f(s)$ to $f(s')$ by playing the edge $f(e)$, a contradiction.

(ii) Let $s = (a, c, x_1, \ldots x_k, y_1, \ldots y_k)$ be a position in P. Let us prove that there is no move from s to a position $s' \notin S$ such that s' is a $\mathcal{P}$-position. Suppose by contradiction that this is not the case, and let s' be a $\mathcal{P}$-position with $s' \notin S$, such that there is a move from s to s'. Since $s' \notin S$, this move necessarily corresponds to playing an edge e marked (3) in Figure 7. Without loss of generality, we can suppose that $e = (c, x_k)$. Since before the move, we had $s \in S$, and after the move we have $s' \notin S$, we know that $x_k = y_k + 1$. The position s' is not canonical. Indeed, after playing the edge e, the vertex with weight y_k becomes a heavy vertex. We can apply the simplification procedure which will first remove this vertex and put a loop on x_k, and then merge the vertices with weights $x_k - 1$ and a. Consequently, s' has the same outcome as:



(1)

where $a' = a + x_k - 1$, $X' = X - x_k$ and $Y' = Y - Y_k$. The equivalence above is obtained by applying induction hypothesis (there are only $k-1$ branches). Denote by G_2' the simplified graph above.
If $X < c + Y$, then we have $G_2 \sim G_3$ (see Figure 7). Since $s \in P$, we know that G_2 is a $\mathcal{P}$-position, and consequently $a + X$ is even. However, since $x_k = y_k + 1$, we also have $X - x_k \geq c - 1 + Y - y_k$, and by the same argument, the outcome G_2' is $\mathcal{P}$ if and only if $a' + X'$ is even. Since $a' + X' = a + x_k - 1 + X - x_k = a + X - 1$, this is not possible.
Thus, we can suppose that $X \geq c + Y$, and then $G_2 \sim G_4$ (see Figure 7). Since $s \in P$, G_2 is a $\mathcal{P}$-position, and by Remark 11 we know that $a + c + Y$ is even. By the equality $x_k = y_k + 1$, we also have $X - x_k \geq c - 1 + Y - y_k$. Consequently, G_2' is not canonical. Indeed, the vertex with weight X' is a heavy vertex, and by applying the simplification procedure, G_2' has the same outcome as:



(2)

Now, since $a+c+Y$ is even, we know that the quantity $a' + c - 1 + Y' = a + x_k - 1 + c - 1 + Y - y_k = a + c - 1 + Y$ is odd, and by Remark 11, this implies that the position above is an $\mathcal{N}$-position. Consequently, G_2' is also an $\mathcal{N}$-position, a contradiction.

(iii) Finally, let $s = (a, c, x_1, \ldots x_k, y_1, \ldots y_k)$ be a position of G_1 such that $s \in S \setminus P$. We want to prove that either s has a move to some position s' in P, or there is a move to a $\mathcal{P}$-position not in S. By assumption, $f(s)$ is a $\mathcal{N}$-position of G_2. We distinguish two possible cases:

- There is an edge e_2 of G_2 such that playing e_2 from the position $f(s)$ is a winning move, and e_2 has a label different from (3). Let $e \in f^{-1}(e_2)$ be an edge of G_1 with the same labels as e_2 such that playing e from s is a legal move, and let s' be the position obtained from s after playing e. Since e has a label different from (3), we have $s' \in S$, and consequently $s' \in P$ since $f(s')$ is a $\mathcal{P}$-position. This proves that there is a move from s to a position $s' \in P$.
- The only winning move from $f(s)$ on G_2 is playing the edge with label (3). As above take e an edge of G_1 with label (3) which is a possible move on s, and denote by s' the position obtained from s after playing e. If $s' \in S$, then $s' \in P$ since $f(s')$ is a $\mathcal{P}$-position. Thus, we can suppose $s' \notin S$. We can also assume that playing an edge in G_2 with a label different than (3) is a losing move. We want to prove that s' is a $\mathcal{P}$-position.

  If $X < Y + c$, then we can see from Figure 7 that there is a winning move in G_2 from the position $f(s)$ by playing an edge with label either (1) or (4). Thus, we can suppose that $X \geq Y + c$, which implies that the outcome of G_2 is the same as the outcome of G_4. We can also assume that the edge labeled (3) is the only winning move for G_4. Indeed, if there was a winning move in G_4 with a label diffèrent than (3), there would be a winning move in G_2 with the same label.

  Using the characterization of Grundy values from Lemma 10, we know that the grundy value of G_4 is $(a + c + Y \mod 2) + 2(\min(a, c) \mod 2)$. Since G_4 is an $\mathcal{N}$-position, this quantity is not zero. We can assume that $\min(a, c)$ is odd. Indeed, if it were even, then decreasing Y by one by playing the edge (4) would be a winning move on G_4. We also have $c < a$, since otherwise one of the two edges marked (1) and (2) would be a winning move. This implies that c is odd. And finally, since playing the edge (3) is a winning move, the position obtained by decreasing c by one must be a $\mathcal{P}$-position, which implies that $a + c - 1 + Y$ is even, and consequently, $a + Y$ is even.

  Without loss of generality, we can assume that $e = (c, x_k)$, and $x_k = y_k + 1$. Using the same argument as in point (ii), we know that s' has the same outcome as the reduced graph in relation (1). Since we know that $X \geq Y + c$, and consequently $X - x_k \geq Y - y_k + c - 1$, this graph can be further simplified to the graph in (2). We know that $c < a$, which implies that $a' = a + x_k - 1 > c - 1$. Additionally, $c - 1$ is even, and $Y' + a' = Y - y_k + a + x_k + 1 = Y + a$ is even. Hence a and Y have the same parity, and by Remark 11 this implies that this graph is a $\mathcal{P}$-position, and consequently, s' is a $\mathcal{P}$-position.

The three conditions of Proposition 12 hold, and the result follows.

□

Theorem 1 is a corollary of this result. We recall that the depth of a vertex u in a rooted tree is the number of edges in the path from the root to u (in particular, the root has depth 0).

Theorem 1. *There is a polynomial time algorithm computing the outcome of* WAK *on any loopless tree of depth at most* 2.

Proof. Let T be a rooted tree of depth at most 2. In T, two leaves attached to the same vertex are false twins. Thus by Lemma 7, they can be merged without changing the outcome of the game. In the resulting tree, each vertex is adjacent to at most one leaf. Now, if a leaf at depth 2 is heavy, then it can be removed, and a loop attached to its neighbor. The vertices with a loop are all false twins (they are all adjacent only to the root of the tree), and can then be merged into a single vertex. If a leaf is adjacent to the root, then we can attach a vertex with weight 0 to it. Let T' be the resulting tree.

By Proposition 7, we have $T \equiv T'$. By applying Lemma 14, the tree T' can be further reduced to P, a path on four vertices with a loop on one end. If P is not canonical, then it can again be reduced to one or several connected components, each with one or two vertices. The Grundy values of these components can be computed thanks to Lemma 9 and Lemma 10. Otherwise, if P is canonical, and Lemma 13 gives the outcome value for P. Since all the reductions, and the characterization of Lemma 13 can be computed in polynomial time, this gives a polynomial time algorithm computing the outcome of any tree of depth at most 2. □

We can see from Lemma 14 and Lemma 13 that the outcome of WAK for a tree of depth at most 2 depends only on the parities of some of the weights and inequalities between them. This is not the case for C_3 (the cycle on three vertices), for which the periodicity of $\mathcal{P}$-positions does not follow this pattern but rather seems to depend on the values modulo 4. A question that arises is whether more complex behaviours can emerge for larger or denser graphs.

However, if the weight of one vertex is large compared to the other weights, we will see in the next section that the behaviour remains simple.

6 Periodicity

In this section, we show a periodicity result on the outcome of WAK positions. More precisely, if we fix the number of counters for all vertices but one, say vertex v_1, the outcomes of this sequence of position is ultimately periodic, with period at most 2. If there is no loop on v_1, when the weight of v_1 is large enough, it becomes a heavy vertex. Thanks to Proposition 7 we already know that the sequence of outcomes is ultimately constant. The following result handles the case where there is a loop on vertex v_1.

Theorem 15. *Let G an unweighted graph with vertices $v_1, \dots v_n$ such that there is a loop attached to v_1. Fix the integers $\omega_i \geq 0$ for $i \geq 2$, and let $\{S_x\}_{x \geq 0}$ be the sequence such that for every x, S_x is the outcome of $(x, \omega_2, \dots \omega_n) \in \mathrm{pos}(G)$. Then $\{S_x\}_{x \geq 0}$ is ultimately 2-periodic with preperiod at most $2 \sum_{i \geq 2} \omega_i$.*

Proof. We show this result by induction on $\Omega = \sum_{i \geq 2} \omega_i$. If the ω_i are all zeros, then $G(x, 0, \dots 0)$ is equivalent to a graph with a single vertex and a loop attached to it. Its outcome is $\mathcal{N}$ if x is odd, and $\mathcal{P}$ if x is even. Suppose that $\Omega > 0$. From the position $(x, \omega_2, \dots \omega_n)$, there are three types of possible moves:

1. $(x - 1, \omega_2, \dots \omega_n)$ by playing on the loop attached to v_1.
2. $(x - 1, \omega'_2, \dots \omega'_n)$ with $\sum_{i \geq 2} \omega'_i = \Omega - 1$ by choosing an edge adjacent to v_1.

3. $(x, \omega'_2, \ldots \omega'_n)$ with $\sum_{i\geq 2} \omega'_i = \Omega - 2$ by choosing an edge not adjacent to v_1 (or $\Omega - 1$ if the edge is a loop).

Let g be the function such that $g(x) = 1$ if, from position $(x, \omega_2, \ldots)$, there is a winning move of type 2. or 3., and $g(x) = 0$ otherwise. Using the induction hypothesis, $g(x)$ is ultimately periodic with period at most 2, and preperiod at most $2\Omega - 1$.

Since the function g takes values in $\{0, 1\}$ and is 2-periodic, there are only 4 possibilities for the values of $g(2\Omega - 1), g(2\Omega), \ldots$. The possible values for $g(2\Omega - 1 + i)$ and the sequence of outcomes of the positions $(2\Omega - 1 + i, \omega_2, \ldots))$, for $i \geq 0$ are summarized in Table 1. We can see that in all four cases in the table, the outcome is periodic starting at $i \geq 2\Omega$.

$g(2\Omega - 1 + i)$, $i \geq 0$	Sequence of outcomes of $(2\Omega - 1 + i, \omega_2, \ldots, \omega_n)$, $i \geq 0$
$1, 1, 1, 1 \ldots$	$\mathcal{N}, \mathcal{N}, \mathcal{N} \ldots$ since there is always a winning move of the form 2 or 3.
$0, 0, 0, 0 \ldots$	$\mathcal{P}, \mathcal{N}, \mathcal{P}, \mathcal{N}, \ldots$ or $\mathcal{N}, \mathcal{P}, \mathcal{N}, \mathcal{P} \ldots$, depending on the outcome of $(2\Omega - 2, \omega_2, \ldots)$. Indeed, playing anything else than the loop attached to v_1 is a losing move and the outcome alternates between $\mathcal{N}$ and $\mathcal{P}$.
$1, 0, 1, 0 \ldots$	$\mathcal{N}, \mathcal{P}, \mathcal{N}, \mathcal{P} \ldots$. Indeed, if i is even, then $g(2\Omega - 1 + i) = 1$, consequently there is a winning move of type 2 or 3. If i is odd, then moves of type 2 and 3 are losing move, and so is the move on the loop attached to v_1.
$0, 1, 0, 1 \ldots$	$X, \mathcal{N}, \mathcal{P}, \mathcal{N}, \mathcal{P}, \ldots$ where X can be either $\mathcal{N}$ or $\mathcal{P}$. Indeed, if i is odd, then $g(2\Omega - 1 + i) = 1$, consequently there is a winning move of type 2 or 3. If i is even and different from 0, then moves of type 2 and 3 are losing move, and so is the move on the loop attached to v_1. When $i = 0$, the outcome can be either $\mathcal{P}$ or $\mathcal{N}$ depending on the outcome of $(2\Omega - 2, \omega_2, \ldots)$.

Table 1 Table of periodicity of the sequence of outcomes of $(2\Omega - 1 + i, \omega_2, \ldots, \omega_n)$, $i \geq 0$ depending on the periodic values of g.

□

Corollary 16. *Given an unweighted graph G, the sequence* $\{\mathcal{G}((x, \omega_2 \ldots, \omega_n))\}_{x\geq 0}$ *of Grundy values for positions of G is ultimately 2-periodic. If the Grundy values in the periodic part are bounded by k, then there is a constant c_k only depending on k such that the preperiod is at most* $2\sum_{i\geq 2} \omega_i + c_k$.

Proof. Simply observe that we can replace G by $G + U$ (the disjoint union of G and U) in the previous statement, for any other graph U. Now if the Grundy value of $(\omega'_1, \ldots \omega'_p) \in \text{pos}(U)$ is k, then $(x, \omega_2, \ldots, \omega'_1 \ldots) \in \text{pos}(G + U)$ is a $\mathcal{P}$-position if and only if $(x, \omega_2, \ldots) \in \text{pos}(G)$ has Grundy value k. Consequently, by applying the result of Theorem 15 on $G + U$, we know that the Grundy values of $(x, \omega_2, \ldots) \in \text{pos}(G)$ are ultimately 2-periodic. The preperiod is at most $2\sum \omega_i + 2\sum \omega'_i$. Taking U and $(\omega'_i)_{i\geq 1}$ such that $\sum \omega'_i$ is the smallest possible with $\mathcal{G}((\omega'_1, \ldots)) = k$, and noting $c_k = 2\sum \omega'_i$ gives the desired result. □

7 Unboundedness of Grundy values

The problem of finding a graph family with unbounded Grundy values is open for a large number of vertex and edge deletion games. Recently, some results have been achieved for the game Graph Chomp [10]. However, for most of these games, the graph families that are studied tend to have ultimately periodic Grundy sequences. For example, in their study of Node-Kayles in [5], the authors found increasingly many irregularities in the non-periodic parts of the Grundy sequences for subdivided stars with three paths, but gave no indication as to whether the irregular values were bounded or not.

In this section, we prove that the Grundy values of WAK are unbounded. This, coupled with the fact that any position of WAK has an equivalent position of Arc-Kayles, as was shown in Section 4, also proves that the Grundy values of Arc-Kayles are unbounded. This result holds even if we restrict ourselves to only play on forests. This answers a problem posed in [14], where the unboundedness of the Grundy values for Arc-Kayles was conjectured. Since Arc-Kayles played on a graph G is Node-Kayles played on the line graph of G, this also implies that the Grundy values of Node-Kayles are unbounded.

We remind the reader that the grundy value of a non-connected graph can be obtained from the values of its connected components:

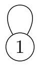
Observation 5. *Let G be a non-connected graph, and $G_1, \ldots, G_k$ its connected components. We have:*

$$\mathcal{G}(G) = \mathcal{G}(\sum_{i=1}^{k} G_i) = \bigoplus_{i=1}^{k} \mathcal{G}(G_i).$$

We prove the following result:

Theorem 2. *The Grundy values for the game* WAK *are unbounded.*

Proof. We inductively build a sequence of graphs, $G_1, G_2, G_3 \ldots$ such that for any $i \neq j$, G_i and G_j have different Grundy values. We construct G_i in such a way that there is a vertex u_i of G_i with a loop attached on u_i such that playing the loop is a winning move.

We take $G_1 =$ 

. There is only one move on G_1 and it is a winning move.

Given a positive integer n, suppose that we have built the graphs $G_1, \ldots, G_n$ with this property. For $1 \leq i \leq n$, let us denote by u_i the vertex of G_i such that there is a winning move on G_i by playing the loop attached to u_i. We construct the graph G_{n+1} in the following way, pictured on Figure 8:

1. For all $i \leq n$, we create two copies G_i' and G_i'' of G_i;
2. We create a vertex u_{n+1} of weight 1 with a loop, which is connected to the vertex u_i' of every G_i'.

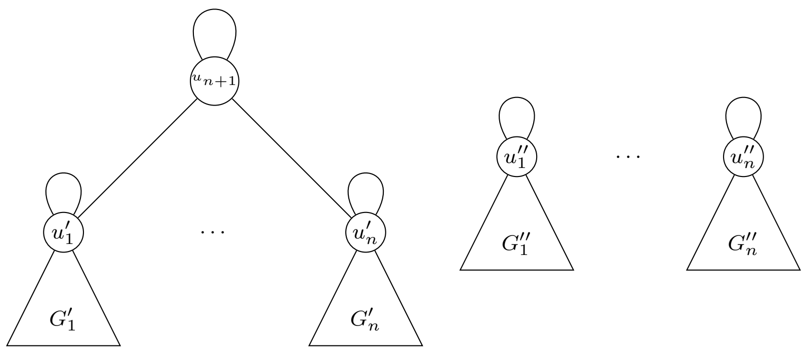


Fig. 8 The inductive construction of the graph G_{n+1}. Note that every vertex has a weight of 1.

Let us now prove that (i) for all $i \leq n$, we have $\mathcal{G}(G_{n+1}) \neq \mathcal{G}(G_i)$ and (ii) playing on the loop attached to u_{n+1} leads to a $\mathcal{P}$-position:

(i) Let $i \leq n$, we will show that there is an option G' of G_{n+1} such that $\mathcal{G}(G') = \mathcal{G}(G_i)$. Using the definition of the Grundy value with the *mex* operator, this implies that $\mathcal{G}(G_{n+1}) \neq \mathcal{G}(G_i)$. Let G' be the option of G_{n+1} obtained by playing on the edge (u_i', u_{n+1}). Since a vertex with no counters left can be removed without changing the game, we now delete both vertices and their incident edges. The graph G' is composed of several components, and can be written as the disjoint sum: $\sum_{\substack{j=1 \\ j\neq i}}^{n}(G_j + G_j) + G_i' + G_i$ where G_i' is the graph obtained from G_i after playing on the loop attached to u_i. Using the induction hypothesis, we know that G_i' is a $\mathcal{P}$-position, hence $\mathcal{G}(G_i') = 0$. Proposition 5 ensures that the Grundy value of G' is:

$$\mathcal{G}(G') = \bigoplus_{\substack{j=1 \\ j\neq i}}^{n} (\mathcal{G}(G_j) \oplus \mathcal{G}(G_j)) \oplus 0 \oplus \mathcal{G}(G_i) = \mathcal{G}(G_i)$$

(ii) Let G' be the the graph obtained from G_{n+1} by playing on the loop attached to u_{n+1}. Then G' can be written as the disjoint sum $G' = \sum_{j=1}^{n} (G_j + G_j)$, which has Grundy value 0 and thus is a $\mathcal{P}$-position.

Thus, the Grundy value of G_{n+1} is different from the Grundy values of the G_i for any $i \leq n$, and a winning move is to play on the loop attached to u_{n+1}. This completes the induction step. □

As shown with Corollary 8, from any position of WAK, one can compute a position of Arc-Kayles with the same Grundy value. Moreover, any position of Arc-Kayles can be changed into an equivalent position of Node-Kayles. Thus Theorem 2 implies the following:

Corollary 17. *The Grundy values for the games* Arc-Kayles *and* Node-Kayles *are unbounded.*

The construction in the proof of Theorem 2 gives a family of graphs of exponential size (by induction, G_n has 3^{n-1} vertices). Since all the vertices have weight 1, the Arc-Kayles positions that we obtain by applying the construction described in the proof of Corollary 8 are of similar size. It may be of interest to find a family of graphs with unbounded Grundy values and of polynomial size, both for WAK and for Arc-Kayles.

References

1. Adams, R., Dixon, J., Elder, J., Peabody, J., Vega, O. and Willis, K. (2016). *Combinatorial Analysis of a Subtraction Game on Graphs.* International Journal of Combinatorics, 2016.
2. Albert, M., Nowakowski, R. and Wolfe D. (2007). *Lessons in play: an introduction to combinatorial game theory.* CRC Press.
3. Berkelamp, E. R., Conway, J. H. and Guy, R. K. (2001-2004). *Winning ways for your mathematical plays* (2nd ed., 4 volumes). Wellesley, MA: A K Peters.
4. Bodlaender, H. L. and Kratsch, D. (2002). *Kayles and nimbers.* Journal of Algorithms, 43(1), 106-119.
5. Fleischer, R. and Trippen, G. (2006). Kayles on the way to the stars. In H. Jaap van den Herik et al. (Eds.), *Computers and Games. 4th International Conference, CG, 2004.* LNCS Book Series (vol. 3846). Berlin Heidelberg: Springer-Verlag. doi:10.1007/11674399_16
6. Guy, R. K. and Smith, C. A. (1956). The G-values of various games. *Mathematical Proceedings of the Cambridge Philosophical Society* (Vol. 52, No. 03, pp. 514-526). Cambridge University Press.
7. Huggan, M. (2015). *Impartial Intersection Restriction Games* (master's thesis). Retrieved from Carleton University Research Virtual Environment (CURVE) (Record b3819898).
8. Lampis, M. and Mitsou, V. (2014). The computational complexity of the game of set and its theoretical applications. *Latin American Symposium on Theoretical Informatics*, (pp. 24-34). Springer Berlin Heidelberg.
9. Noon, H. and Van Brummelen, G. (2006). The non-attacking queens game. *The College Mathematics Journal*, 37(3), 223-227.
10. O'Sullivan, C. (2017). *A vertex and edge deletion game on graphs.* Retrieved from arXiv:1709.01354.
11. Schaeffer, T. J. (1978). On the complexity of some two-person perfect-information games. *J. Comput. System Sci.*, 16(2), 185–225. doi:10.1016/0022-0000(78)90045-4
12. Siegel, A. N. (2013). *Combinatorial game theory* (Vol. 146). American Mathematical Soc..
13. Sprague, R. (1935). Über mathematische Kampfspiele. *Tohoku Mathematical Journal, First Series*, 41, 438–444.
14. Stevens, B. and Huggan M. (2016). Polynomial Time Graph Families For Arc-Kayles. *INTEGERS*, vol. 16, p. 2.