Kayles on the Way to the Stars

Rudolf Fleischer[1] and Gerhard Trippen[2,⋆]

[1] Shanghai Key Laboratory of Intelligent Information Processing,
Department of Computer Science and Engineering,
Fudan University, Shanghai, China
fleischer@acm.org

[2] Department of Computer Science,
The Hong Kong University of Science and Technology, Hong Kong
trippen@cs.ust.hk

Abstract. We present several new results on the impartial two-person game Kayles. The original version is played on a row of pins ("kayles"). We investigate variants of the game played on graphs. We solve a previously stated open problem in proving that determining the value of a game position needs only polynomial time in a star of bounded degree, and therefore finding the winning move - if one exists - can be done in linear time based on the data calculated before.

1 Introduction

Studying combinatorial games may show new interesting algorithmic challenges and may lead to precise problems in complexity theory. Games as models for a diverse set of practical problems attract researchers in mathematics and computer science. Combinatorial games are intrinsically beautiful with interesting features and a theory whose study also provides entertainment on its own.

Many relations between graph theory and combinatorial game theory exist. Like many other games that are being played on graphs, the versions of Kayles we are going to investigate are played on graphs too.

Kayles is a combinatorial game played by two persons who move alternately. Both players can make any move for all game positions. Therefore the game is impartial. On the contrary for example chess is partizan: White can only move white pieces, and Black can only move black pieces. All information about a position is known to both players at any time. There are no chance moves, i.e., no randomization generated by a dice for example is involved.

When playing Kayles on a graph choosing one vertex will remove this vertex and all its neighbors from the graph. Thus the game played on a finite graph is obviously of bounded play. In every move at least one vertex will be taken (this might disconnect the graph in several components), and therefore the game must terminate in time linear in the size of the initial position.

⋆ The work described in this paper was partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. HKUST6010/01E).

H.J. van den Herik et al. (Eds.): CG 2004, LNCS 3846, pp. 232–245, 2006.

We only consider games of normal type, which means a player loses if he[1] is unable to make a move, opposed to the misere version where the last player to move loses.

All these features of the game make it possible to analyze it with the Sprague-Grundy theory [5,1]. Grundy has shown that there exists a function $\mathcal{G} : \mathcal{P} \rightarrow \mathcal{Z}$ from the set of possible game positions $\mathcal{P}$ into the set of integers $\mathcal{Z}$ with the following properties.

For any terminal position P we have $\mathcal{G}(P) = 0$. For any other position P, $\mathcal{G}(P)$ can always be calculated (inductively, starting from the terminal positions). Hereby a position P will take as its value the smallest non-negative integer, called the minimal excluded value (mex), different from all values of $\mathcal{G}(Q_i)$, where position Q_i can be reached from position P by one single move.

The value of a disjunctive combination of games is the *Nim-sum* of the values of the components. The Nim-sum is obtained as the sum of the binary representations of the values added together using the XOR (eXclusive OR) operation, denoted by the commonly used symbol $\oplus$.

If $\mathcal{G}(P)$ is a positive integer, then the game position P can be won by the first player; otherwise the second player can win.

This paper is organized as follows. In Section 2, we give background information to the game we are analyzing. Also different variants will be mentioned. In Section 3, we describe an algorithm how to find a winning move, and show that it always can be found in polynomial time. Determining that there is no winning move in a particular position can be done in polynomial time, too. In Section 4 and Section 5, we present how differences in the underlying sequences affect the sequences of the star game. We close with conclusions and some open problems in Section 6.

2 Background

Below we provide some history on Kayels (2.1) and discuss Kayles on Graphs (2.2).

2.1 History of Kayles

The game Kayles was introduced by Dudeney and independently also by Sam Loyd, who originally called it 'Rip Van Winkle's Game'. It was supposed to be played by skillful players who could either knock down exactly one or two adjacent pins ("kayles") out of a row of pins [3].

We could also think of starting with one heap (or several heaps) of beans, and give the following description of the rules. Each player, when it is his turn to move, may take 1 or 2 beans from a heap, and, if he likes, split what is left of that heap into two smaller heaps [1]. This game is well studied and the value of every game position can be determined in constant time [1].

[1] In this contribution 'he' is used when both 'he' and 'she' are possible.

2.2 Kayles on Graphs

Node-Kayles, a variant of the original Kayles, is played on a graph $G = (V, E)$ with n nodes in the node set V and m edges in the edge set E.

In this paper we will mainly consider the variant in which a move consists of selecting a node v and thereafter removing it and all its neighbors $N(v)$ from the graph. In case of a directed graph, only those vertices $w \in V$ will be called neighbors that can be reached using a directed edge $e = (v, w)$ from node v to node w. In addition, we will look at a variant that is closer to the original Kayles: either one single node or a node together with an arbitrary neighbor can be taken away.

For all variants of Kayles considered we will only investigate normal play, i.e., the first player unable to move loses. Schaefer proved that the problem to determine which player has a winning strategy for Node-Kayles played on arbitrary graphs is PSPACE-complete [7]. Bodlaender and Kratsch investigated the game on special classes of graphs, which include graphs with a bounded asteroidal number, cocomparability graphs, circular arc graphs, and cographs, and showed that the problem is polynomial time solvable in these cases [2].

In this paper we will describe how to find the value of a game position in polynomial time for stars. This solves an open problem in [2]. A *star* is an acyclic connected graph with one distinguished node, the *center* of the star. The center may have any degree Δ, all other nodes have degree at most 2, i.e., they lie on paths emanating from the center, so-called rays.

We will denote a star of degree Δ by $S_{l_1,l_2,\ldots,l_\Delta}$, where the l_i are natural numbers and stand for the length of ray i. Naturally rays of zero-length can be neglected. A degenerated star consisting of only one vertex has degree 0. Degenerated stars of degree one or two form a simple path only. In case of degree one the path length is $l_1 + 1$; for the degree-two case we have a path of length $l_1 + l_2 + 1$.

Kano considered stars where rays always have length 1 [6]. Furthermore in his game not vertices will be removed but any number of edges whereby these edges must belong to the same star. Kano gave some results on double-stars and on forks. A double-star is a graph obtained from two stars by joining their two centers by a new edge, and a fork is defined to be a graph which is obtained from a star and a path by joining the center of the star to one of the end vertices of the path by a new edge.

Node-Kayles has the same characteristics as Dawson's Chess where two phalanxes of Pawns are facing each other just one row apart. The Pawns step forward and capture diagonally as usual in chess. But capture is obligatory so queening is not possible. The winner is the last player to move. (In the original version by Dawson the last player to move loses [1].)

Both games have the same octal encoding .137. Octal encoding is used to describe different variants of take-and-break games, see Table 1 [5,1]. In that sense, .137 means: one single vertex can only be removed if it has no outgoing edges (anymore). Two vertices can only be removed if they form one connected component or if after their removal the remaining part of that connected com-

Table 1. Interpretation of Code Digits

Value of digit d_k	Condition for removal of k beans from a single heap.
0	Not permitted.
1	If the beans removed are the whole heap.
2	Only if some beans remain and are left as a single heap.
3	As 1 or 2.
4	Only if some beans remain and are left as exactly two non-empty heaps.
5	As 1 or 4 (i.e., not 2).
6	As 2 or 4 (i.e., not 1).
7	Always permitted (i.e., 1 or 2 or 4).

ponent is still connected. For three vertices we have more possibilities. We are allowed to remove them if the remaining vertices are left in at most two connected components.

The Grundy values for Node-Kayles have the periodicity 34.
The 34 values for $n \equiv 0, 1, 2, \ldots, 33 \pmod{34}$ are
$(8, 1, 1, 2, 0, 3, 1, 1, 0, 3, 3, 2, 2, 4, 4, 5, 5, 9, 3, 3, 0, 1, 1, 3, 0, 2, 1, 1, 0, 4, 5, 3, 7, 4)$,
with the following seven exceptions.

- $\mathcal{G}(n) = 0$ when $n = 0, 14, 34$;
- $\mathcal{G}(n) = 2$ when $n = 16, 17, 31, 51$.

3 A Step into the No Man's Land

Fraenkel mentioned that there lies a huge no man's land between the polynomial time 0.137 and the PSPACE-hard Node-Kayles [4]. To explore a part of that no man's land we take a look at the stars first. Node-Kayles played on stars will be called Star-Kayles for short.

We can characterize the possible moves in Star-Kayles as follows.

1. Choosing the center vertex of the star will remove $\Delta + 1$ vertices, and split the star in up to Δ simple paths. It seems difficult to denote this game in the octal notation, or any other analogous polynomial notation.
2. Removing a vertex adjacent to the star center will split the star into either $\Delta - 1$ or Δ simple paths. Thereby either three or only two vertices will be removed (if the corresponding ray had only length 1).
3. Choosing the second next vertex to the center of a ray will leave a star of degree $\Delta - 1$ and (a possibly empty) simple path.
4. For any other vertex we will get a star of the same degree with one ray shortened accordingly to the position of the chosen vertex as well as a simple path (possibly empty again).

As the smallest representative for non-degenerated stars we choose stars of degree 3. This saves time and space for computations, but the following arguments can easily be extended to higher degrees. S_{l_1,l_2,l_3} denotes a star with rays of length l_1, l_2, and l_3, respectively.

3.1 A Star Is Born

Let us begin with the simplest star. The center appears first and then three rays will be sent out from the center. First we want to find the Grundy values for the class of stars $S_{l_1,1,1}$.

For $l_1 = 1$ the whole star can be removed in one move by choosing the center vertex (case (1) above). Therefore this situation is a winning position for the first player. Choosing any other of the vertices (2) will always have the same outcome, namely two single vertices. Adding up their values will of course result in a zero-game. No other configuration can be achieved and therefore $\mathcal{G}(S_{1,1,1}) = 1$.

For $l_1 = 2$ we have more possibilities:
Removing the center vertex (1) will leave one single vertex, Grundy value 1.
Choosing the vertex of ray r_2 or r_3 (2) will leave a simple path of length 2 and a single vertex. Adding up these two components will result in a zero-game.
We will also get the value 0 when choosing the first vertex of r_1 (2), because only two single vertices remain.
The last choice (3) will remove ray r_1 completely, and therefore leave a degenerated star behind with one center vertex and two rays of length 1 each. This is actually a simple path of length 3. Its Grundy value is 2.
Now we can determine the mex of these values, which is 3.

We will continue to determine the Grundy values of the stars $S_{l_1,1,1}$ in this way. Since the first move on a larger star may split it into a smaller one, it is best to compute the Grundy values iteratively for increasing values of l_1.

We analyze all four cases of star-decompositions mentioned above occurring for the stars $S_{l_1,1,1}$.

1. Removing the center will just leave a simple path of length $l_1 - 1$, whose value is $\mathcal{G}(l_1 - 1)$.
2. Choosing the first vertex of one of the rays r_2 or r_3 leaves a simple path of length l_1 and a single vertex. The value of this position is $\mathcal{G}(l_1) \oplus \mathcal{G}(1) = \mathcal{G}(l_1) \oplus 1$.
   Choosing the first vertex of r_1 leaves two single vertices and a simple path of length $l_1 - 2$, i.e., a position of value $\mathcal{G}(l_1 - 2) \oplus \mathcal{G}(1) \oplus \mathcal{G}(1) = \mathcal{G}(l_1 - 2) \oplus 0 = \mathcal{G}(l_1 - 2)$.
   To include into this more general description the case $S_{1,1,1}$ we assume that $\mathcal{G}(k) = 0$ for $k \leq 0$.
3. Choosing the second vertex of ray r_1 leaves two simple paths with length $l_1 - 3$ and 3 with value $\mathcal{G}(l_1 - 3) \oplus \mathcal{G}(3) = \mathcal{G}(l_1 - 3) \oplus 2$.
4. When choosing the very last vertex of ray r_1 we get a position of value $\mathcal{G}(S_{l_1-2,1,1})$. For all other vertices (along r_1) the set of all possible values is

$$\bigcup_{i=2}^{l_1-1} (\mathcal{G}(S_{l_1-i,1,1}) \oplus \mathcal{G}(i-3)).$$

To write all possible cases in a more compact way let us define the Grundy values for stars with non-positive length of ray r_1:

- $\mathcal{G}(S_{0,1,1}) = \mathcal{G}(3) = 2$;
- $\mathcal{G}(S_{-1,1,1}) = \mathcal{G}(1) \oplus \mathcal{G}(1) = 0$;
- $\mathcal{G}(S_{-2,1,1}) = \mathcal{G}(0) \oplus \mathcal{G}(0) = 0$.

Removing the vertex of one of the rays r_2 or r_3 (case 2) leads to a game value of $\mathcal{G}(l_1) \oplus 1$ (actually it could also be denoted by $\mathcal{G}(S_{l_1,-1,1})$). In all other cases, we get game value

$$\bigcup_{i=1}^{l_1+1} (\mathcal{G}(l_1 - i) \oplus \mathcal{G}(S_{i-3,1,1})).$$

Next we prove that the class of stars $S_{l_1,1,1}$ is ultimately periodic. The proof is very similar to the proof of the ultimate periodicity of the game 0.137 itself given in [1]. The fundamental idea is that the calculations of Nim-sequences is made easier by using a Grundy scale. Figure 1 shows such a scale being used for the computations of our star classes (a more detailed description follows below.) In general, successive values are written on squared paper and the arrowed entry is computed as the mex of all (or sometimes some accordingly to the game chosen) preceding entries. Then the scale is moved on one place. A very nice tool to do calculations of such kind (and many more) is the Combinatorial Game Suite by Aaron Siegel [8].

In our case we can also use Grundy scales. However, on the first scale we only write the Grundy values for 0.137. We can do this for a few numbers at the beginning first, and then just append more if needed. We fix this scale on the table. Our second scale will carry the Grundy values for our stars in reverse order. It will be shifted step by step to the right. The Grundy value for the star we are calculating will always be written into the first empty cell to the left of the already computed values. For an illustration see Figure 1. The arrow indicates the next value to be determined.

With this picture in mind we can easily transfer the proof given for 0.137 to our class of stars. We keep on calculating game values until two complete periods p lie between the last irregularity of 0.137, which is already known to be $i_1 = 51$,

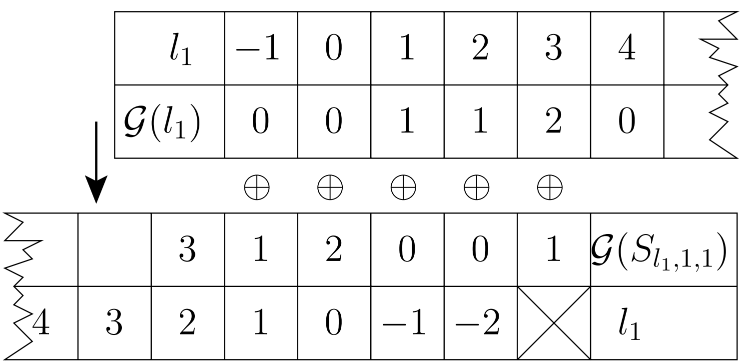

Fig. 1. Calculating $\mathcal{G}(S_{l_1,1,1})$ in Star-Kayles

and the last observed irregularity of our star class, which we call i_2. As at most three nodes can be cut out of a ray we calculate $t = 3$ additional values. So the last value that needs to be computed to verify that the period persists is

$$\mathcal{G}(i_1 + i_2 + 2p + t).$$

This gives the following ultimately periodic sequence for $l_1 \equiv 0, 1, 2, \ldots, 33 \pmod{34}$: (2, 9, 3, 15, 14, 1, 9, 4, 4, 14, 5, 13, 4, 0, 8, 1, 2, 4, 8, 5, 13, 2, 4, 8, 5, 9, 4, 12, 8, 6, 9, 9, 0, 8); Table 2 shows the first 374 game values of that class; $S_{310,1,1}$ is the last irregular value.

Table 2. Game values for the star class $S_{l_1,1,1}$. (A table entry gives the game value for l_1 = row-header + column-header.).

l_1	1	2	3	4	5	6	7	8	9		11		13		15		17		19		21		23		25		27		29		31		33	
0	1	3	0	0	1	1	4	0	5	1	1	0	0	3	1	2	0	0	1	2	2	3	3	5	2	4	3	3	2	2	1	0	0	2
34	1	3	0	0	1	7	4	4	6	5	7	0	0	3	1	2	0	0	1	2	2	4	3	5	6	4	7	3	6	2	1	0	0	2
68	1	3	0	8	1	9	4	2	8	5	9	4	0	3	1	2	0	0	1	2	2	4	8	5	6	4	7	8	6	9	1	0	0	2
102	1	3	0	8	1	9	4	4	8	5	9	4	0	8	1	2	0	0	1	2	2	4	8	5	9	4	12	8	6	9	9	0	0	2
136	1	3	0	8	1	9	4	4	14	5	13	4	0	8	1	2	0	0	1	2	2	4	8	5	9	4	12	8	6	9	9	0	0	2
170	1	3	0	8	1	9	4	4	14	5	13	4	0	8	1	2	4	0	1	2	2	4	8	5	9	4	12	8	6	9	9	0	8	2
204	9	3	0	8	1	9	4	4	14	5	13	4	0	8	1	2	4	0	1	2	2	4	8	5	9	4	12	8	6	9	9	0	8	2
238	9	3	0	8	1	9	4	4	14	5	13	4	0	8	1	2	4	0	1	2	2	4	8	5	9	4	12	8	6	9	9	0	8	2
272	9	3	0	8	1	9	4	4	14	5	13	4	0	8	1	2	4	0	5	2	2	4	8	5	9	4	12	8	6	9	9	0	8	2
306	9	3	15	8	1	9	4	4	14	5	13	4	0	8	1	2	4	8	5	13	2	4	8	5	9	4	12	8	6	9	9	0	8	2
340	9	3	15	14	1	9	4	4	14	5	13	4	0	8	1	2	4	8	5	13	2	4	8	5	9	4	12	8	6	9	9	0	8	2

3.2 The Magic Number of the Game .137 Is 34

When we just gave the proof of the ultimate periodicity of the class $S_{l_1,1,1}$ we did this with the implicit understanding that Star-Kayles has the same period as the basic game. This is indeed the fact and not too surprising as the only difference in this variant to the original game occurs in splitting the star by either removing the center or one of its neighbors. All other moves only break rays which is similar to breaking a row in the original game.

3.3 Dynamic Programming and Memoization

Having calculated all values of the class $S_{l_1,1,1}$ as far as necessary we take a look at the class $S_{l_1,2,1}$. The only difference is that we need to include two further values before we can determine the mex of the set, $\mathcal{G}(l_1+1+1)$ and $\mathcal{G}(l_1) \oplus \mathcal{G}(2)$. These positions arise from either removing the second vertex of r_2 or the first and only vertex of r_3. Furthermore, choosing the center vertex will now leave a game position consisting of two simple paths with value $\mathcal{G}(l_1 - 1) \oplus \mathcal{G}(1)$.

The situation becomes different when we start to examine $S_{l_1,3,1}$. For the first time we need to consider values of stars that do not lie in the same class. Choosing the last vertex of r_2 will leave us with a star of the class $S_{l_1,1,1}$. Here memoization comes into play. Having stored all the values of our former computations we simply look up the desired value.

To show which former values are needed to calculate the value of any star S_{l_1,l_2,l_3} we shall generalize the star decomposition.

1. Choose the center vertex:
   $\mathcal{G}(l_1 - 1) \oplus \mathcal{G}(l_2 - 1) \oplus \mathcal{G}(l_3 - 1)$.
2. Choose the first vertex of any ray:
   For each $i \in \{1, 2, 3\}$, and $i \neq j \neq k \neq i \in \{1, 2, 3\}$
   $\mathcal{G}(l_i - 2) \oplus \mathcal{G}(l_j) \oplus \mathcal{G}(l_k)$.
3. Choose the second vertex of any ray:
   For each $i \in \{1, 2, 3\}$, and $i \neq j \neq k \neq i \in \{1, 2, 3\}$
   $\mathcal{G}(l_i - 3) \oplus \mathcal{G}(l_j + l_k + 1)$.
4. Choose any other vertex:
   For each $i \in \{1, 2, 3\}$, and $i \neq j \neq k \neq i \in \{1, 2, 3\}$
   $\bigcup_{l=4}^{l_i+1} (\mathcal{G}(l_i - l) \oplus \mathcal{G}(S_{l-3,l_j,l_k}))$.

After defining

- $\mathcal{G}(S_{0,l_2,l_3}) = \mathcal{G}(l_2 + l_3 + 1) = 2$,
- $\mathcal{G}(S_{-1,l_2,l_3}) = \mathcal{G}(l_2) \oplus \mathcal{G}(l_3)$, and
- $\mathcal{G}(S_{-2,l_2,l_3}) = \mathcal{G}(l_2 - 1) \oplus \mathcal{G}(l_3 - 1)$,

we can again state a more compact formula for (1) – (4):

For each $i \in \{1, 2, 3\}$, and $i \neq j \neq k \neq i \in \{1, 2, 3\}$ $\bigcup_{l=2}^{l_i+1} (\mathcal{G}(l_i - l) \oplus \mathcal{G}(S_{l-3,l_j,l_k}))$.

We remark that the term for case (1) will appear three times in the above union of all values. Taking the mex of the union of all those sets will give us the game value $\mathcal{G}(S_{l_1,l_2,l_3})$ for any star S_{l_1,l_2,l_3}. Based on this approach we can use dynamic programming to find the Grundy value of any star. Given that the values of all stars smaller than the considered star have already been calculated it takes only $O(n)$ time to find its value, whereby n is the size of the star.

Observation
While using memoization we only need to store the values of stars $S_{l_1,l_2,\ldots,l_\Delta}$, where $l_1 \geq l_2 \geq \ldots \geq l_\Delta$. This comes from the fact that we are only interested in the combinatorial structure of the graph, and therefore all permutations of rays are isomorphic in our viewpoint.

3.4 Leaving the Orbit

While looking at all the data and comparing ultimately periodic sequences of different star classes we first thought that there might be one last exceptional star, after which all other stars can be evaluated by looking at ultimately periodic sequences. However, we had to realize that we could not find that last irregularity. Instead, we observed a constant growth of the last irregular value with the growth of the ray lengths. Therefore we will now investigate the relation between the last irregular value and the ray lengths of the stars.

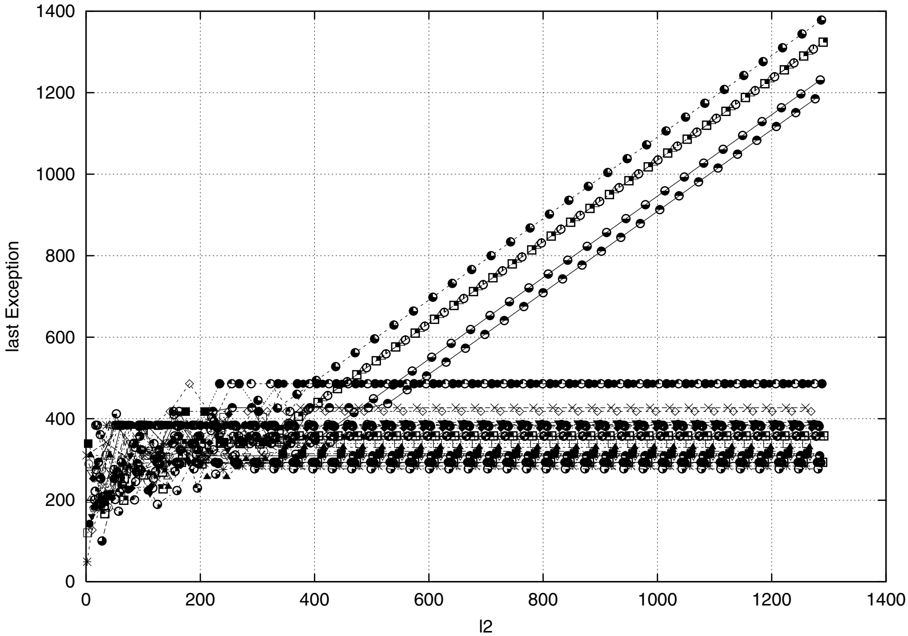

Fig. 2. Last irregular values of the star class $S_{l_1,l_2,1}$

As an example we look at all classes with a fixed ray length $l_3 = 1$ and arbitrary ray length for r_1 and r_2 (see Figure 2). The diagram shows all last irregular values (vertical axis) of the sequences as a function of the length l_2 of the second ray (horizontal axis). A detailed look at all values is not necessary. The aim is to point out that nearly all of these classes become ultimately periodic, while for a few classes the last irregularity will not stop to grow.

All star classes can be grouped in 34 equivalence classes by taking the modulo of the ray length l_2. Table 3 shows all ultimately periodic sequences of star classes with the third ray fixed to length 1, and the second ray has length 1 (mod 34). For greater values of l_2 there is no change anymore in the ultimately periodic sequence.

Most of these equivalences actually become ultimately periodic. However, the star classes $S_{l_1,l_2,1}$ with $l_2 \equiv 7, 15, 18, 27, 29, 32 \pmod{34}$ do not show this behavior. Instead, the last irregularity will grow with the length l_2 of the second ray (as depicted in Figure 2). To give an explanation we prove an easy lemma first.

Table 3. Ultimately periodic sequences of star classes for $l_3 = 1$ and $l_2 \equiv 1 \pmod{34}$

l_2	ultimately periodic sequence																																	
1	9	3	15	14	1	9	4	4	14	5	13	4	0	8	1	2	4	8	5	13	2	4	8	5	9	4	12	8	6	9	9	0	8	2
35	1	14	0	11	18	1	9	0	17	14	14	19	4	18	6	2	15	0	8	18	2	16	3	21	1	24	23	3	13	2	10	19	0	18
69	1	14	0	8	1	1	10	0	24	14	26	23	0	3	1	2	20	0	22	2	2	3	3	16	25	17	27	3	28	2	1	0	0	23
103	1	14	0	11	1	1	31	0	25	10	28	23	0	25	1	2	24	0	18	2	2	3	3	32	1	23	27	3	21	6	14	0	0	18
137	1	33	0	27	32	1	33	0	24	40	33	33	0	18	1	2	21	0	32	2	2	3	3	32	1	23	33	3	15	6	30	0	0	23
171	1	40	0	36	39	1	40	0	36	37	33	34	0	35	1	2	38	0	32	2	2	3	3	36	1	34	38	3	15	6	33	0	0	29
205	1	40	0	34	18	1	35	0	25	10	28	40	0	34	1	2	24	0	18	2	2	3	3	40	1	24	23	3	26	6	14	0	0	25
239	1	37	0	36	41	1	42	0	25	10	28	44	0	35	1	2	20	16	47	2	2	3	3	21	1	28	23	3	17	6	14	0	0	43
273	1	36	0	47	41	1	42	0	25	10	28	47	0	34	1	2	20	16	46	2	2	3	3	21	1	28	23	3	17	6	14	0	0	34
307	1	35	0	34	41	1	35	0	25	10	28	44	0	34	1	2	20	16	48	2	2	3	3	21	1	28	23	3	17	6	14	0	0	34
341	1	35	0	34	47	1	35	0	25	10	28	48	0	34	1	2	20	16	49	2	2	3	3	21	1	28	23	3	17	6	14	0	0	34
375	1	35	0	34	47	1	35	0	25	10	28	28	0	34	1	2	20	16	46	2	2	3	3	21	1	28	23	3	17	6	14	0	0	34
409	1	35	0	34	47	1	35	0	25	10	28	28	0	34	1	2	20	16	46	2	2	3	3	21	1	28	23	3	17	6	14	0	0	34
443	1	35	0	34	47	1	35	0	25	10	28	28	0	34	1	2	20	16	46	2	2	3	3	21	1	28	23	3	17	6	14	0	0	34
477	1	35	0	34	47	1	35	0	25	10	28	28	0	34	1	2	20	16	46	2	2	3	3	21	1	28	23	3	17	6	14	0	0	34
511	1	35	0	34	47	1	35	0	25	10	28	28	0	34	1	2	20	16	46	2	2	3	3	21	1	28	23	3	17	6	14	0	0	34
545	1	35	0	34	47	1	35	0	25	10	28	28	0	34	1	2	20	16	46	2	2	3	3	21	1	28	23	3	17	6	14	0	0	34
579	1	35	0	34	47	1	35	0	25	10	28	28	0	34	1	2	20	16	46	2	2	3	3	21	1	28	23	3	17	6	14	0	0	34

Lemma 1. *Let l_2 and l_3 be fixed lengths of rays r_2 and r_3, and w. l. o. g. $l_2 \geq l_3$. For all l_2 greater than all last irregularities of all classes involved in the calculation of the Grundy-values of stars with ray r_1 of length i from* 1 *up to $l_2 - 1$ we get $\mathcal{G}(S_{i,l_2,l_3}) = \mathcal{G}(S_{i,l_2+34,l_3})$.*

Proof. The values $\mathcal{G}(S_{i,l_2,l_3})$ and $\mathcal{G}(S_{i,l_2+34,l_3})$ for all $i = 1, \ldots, l_2 - 1$ will be looked up as $\mathcal{G}(S_{l_2,i,l_3})$ and $\mathcal{G}(S_{l_2+34,i,l_3})$. As l_2 is greater than the last irregularity of this class and this class also has period 34 these two values must be equal. □

We observe that after passing this last irregularity the sequence will also become ultimately periodic. But for special classes the last irregularity keeps on growing with the lengths of the rays. Therefore we can never find an ultimate, huge star whose sequence builds the ultimately periodic sequence for the sequences of earlier stars. Furthermore, our calculations show that the classes that reveal the feature of a growing last exception are not the same for different values of l_3; not even values for values of l_3 that lie in the same equivalence class modulo 34.

3.5 The Milky Way

As we have seen so far, we can calculate the value of a single star using memoization in polynomial time. As already mentioned in the introduction, a game position composed by a disjunctive combination of several components can be evaluated as the Nim-sum of them. Therefore a game position consisting of many stars is simply the Nim-sum of the Grundy values of all these stars. Hence, computing the value of a game position of several stars needs time polynomial in their sizes. If the value of the position is zero, then there is no wining move. For any positive value we know that there exists a winning move. Every star exhibits only a linear number of decompositions. Therefore, the actual winning move can be found in linear time after the values of the decompositions have been calculated.

4 If the World Were Regular

In this section, we try to find out why Node-Kayles exhibits irregularities in its periodicity of 34, whether this is just an artifact from the underlying Kayles or whether there are also other reasons. To this purpose, let us assume in this section that Kayles was periodic without any irregularities. Then, how would Star-Kayles behave?

If we repeat the same investigations as before, the sequences of the star classes are looking quite regular. As an example we give Table 4, which shows all different ultimately periodic sequences of star classes with the third ray fixed to length 1, and the second ray has length 1 (mod 34).

Table 4. Ultimately periodic sequences of star classes for $l_3 = 1$ and $l_2 \equiv 1 \pmod{34}$ based on a regular sequence

l_2	ultimately periodic sequence
1	9 3 8 8 1 9 4 2 8 5 9 4 0 3 1 2 4 8 5 10 2 3 3 5 5 4 4 8 2 2 9 0 8 2
35	1 16 0 0 1 1 16 0 8 9 1 10 0 3 12 2 0 0 11 2 10 14 3 12 2 9 10 3 2 17 1 0 0 2
69	1 17 0 0 1 1 10 0 16 9 1 18 0 3 16 2 0 0 13 2 22 3 3 19 2 9 24 3 2 2 1 0 0 2
103	1 17 0 0 1 1 32 0 21 9 1 20 0 3 16 2 0 0 13 2 10 3 3 31 2 32 28 3 2 2 1 0 0 2
137	1 17 0 0 1 1 32 0 34 9 1 20 0 3 16 2 0 0 13 2 10 3 3 26 2 17 28 3 2 2 1 0 0 2

Already for $l_2 = 137$ the ultimately periodic sequence has reached the point where it will not change anymore; for all greater values of l_2 with $l_2 \equiv 1 \pmod{34}$ we get the same ultimately periodic sequence. The same holds for all other equivalence classes of $l_2 \pmod{34}$. However, the considerations above were only for a fixed length 1 of the third ray.

To build a basis for all games on degree-3 stars we need to look at different lengths of r_3 as well. From 2 to 20 we get the same regular behavior. But then the unexpected happens: if $l_3 = 21$, then there is an irregular value in the class $l_2 \equiv 22 \pmod{34}$ which is growing with increasing length l_2. There is one very remarkable point about this: those stars have their last irregular value when l_1 is equal to l_2.

5 Back to the Roots

After observing the stars in the sky for a while we start to feel the infinity of the universe: the last irregular value will move farther and farther away as the

Table 5. Changes in the length of the period of sequences of star classes for $l_3 = 1$. (A table entry gives the period multiplier for l_2 = row-header + column-header.).

l_2	1	2	3	4	5	6	7	8	9	10	11	12
0	1	1	1	1	1	1	1	1	1	1	1	1
12	1	1	1	1	1	1	1	1	1	1	1	14
24	1	1	1	1	1	1	1	1	1	1	1	1
36	14	1	14	1	1	1	14	14	1	14	1	1
48	1	5	1	1	1	1	5	5	1	10	14	1
60	1	10	1	1	14	1	10	1	1	10	10	1
72	5	1	1	1	10	1	1	1	10	1	1	10
84	14	1	1	1	1	1	1	5	1	14	1	1
96	1	1	1	70	10	1	1	5	1	5	1	10
108	10	1	10	1	70	1	10	1	1	1	14	1
120	1	1	14	1	1	5	1	1	70	70	70	5
132	1	1	1	1	14	10	1	1	5	70	1	1
144	1	1	1	1	70	10	10	1	1	70	70	10
156	10	1	1	70	1	14	10	1	70	70	70	1
168	10	1	1	70	70	10	10	1	70	70	70	10
180	10	1	1	14	14	10	10	1	70	70	70	1
192	1	1	1	14	1	10	1	1	14	70	14	1
204	10	70	1	14	1	10	1	1	70	70	14	1
216	1	70	1	14	70	10	1	1	70	70	70	1
228	1	70	1	70	1	1	1	1	14	70	14	1
240	1	70	1	14	1	1	1	1	70	70	14	1
252	1	1	1	14	1	1	1	1	14	70	14	1
264	1	1	1	14	1	1	1	1	14	70	14	1
276	1	1	1	14	1	1	1	1	14	70	70	1
288	1	1	1	14	1	1	1	1	14	70	70	1
300	1	1	1	14	1	1	1	1	14	70	14	1
312	1	1	1	14	1	1	1	1	14	70	14	1
324	1	1	1	14	1	1	1	1	14	70	14	1

rays of the stars get longer and longer. This makes us think whether we should come back to earth to take a look at the roots: Pin-Kayles.

In Pin-Kayles a player can knock out one or two adjacent pins only. Thereby the remaining pins might also be split into several groups. In the original game we have only one row of pins, and therefore at most two groups emerge, but in our star-shaped setting we can get up to d groups if we remove the center pin.

The Grundy values for Pin-Kayles have the periodicity 12, and show the following sequence for $n \equiv 0, 1, 2, \ldots, 11 \pmod{12}$: $(4, 1, 2, 8, 1, 4, 7, 2, 1, 8, 2, 7)$, with the following 14 exceptions.

- $\mathcal{G}(n) = 0$ when $n = 0$;
- $\mathcal{G}(n) = 3$ when $n = 3, 6, 18, 39$;
- $\mathcal{G}(n) = 4$ when $n = 9, 21, 57$;
- $\mathcal{G}(n) = 5$ when $n = 28$;
- $\mathcal{G}(n) = 6$ when $n = 11, 22, 34, 70$;
- $\mathcal{G}(n) = 7$ when $n = 15$.

5.1 An Alteration Spell

The magic number of Pin-Kayles is 12, but as we start to play the game on stars we quickly encounter a class of stars that shows a different period. Stars with $l_2 = 24$ and $l_3 = 1$ have a 14-times lengthened period. Every 168 numbers we will get exactly the same numbers again. Only one out of the original 12 positions causes this new period. The next candidate that shows the same behavior and same period has a ray length of $l_2 = 37$. Here, two different positions display irregularities. Three further multipliers of 12 can be found which are 5, 10, and 70.

Table 5 lists the changes in periods as multipliers of 12 for an increasing ray length l_2 and a fixed ray length $l_3 = 1$. The last three rows show the same pattern, and also later rows do not present any changes anymore.

In contrast to the other two Node-Kayles versions we could not find yet a class that does not stop shifting the position of its last irregular value.

Another fact seems notable. The last irregular value of the underlying Pin-Kayles sequence appears at position 70. This coincides with the greatest multiplier of the period we have found so far.

6 The Forest Lies Ahead

We have shown that the problem of determining game values for stars can be done in polynomial time using dynamic programming and memoization. We have seen that Node-Kayles played on stars cannot lead to an ultimate periodic sequence that can be used to describe all stars bigger than a certain size. Instead we presented star classes whose last irregular value will continue to move farther and farther away with growing length of the rays.

Pin-Kayles played on stars seems more promising but further studies are necessary to prove or disprove the existence of such an ultimate periodic sequence.

Our investigations have given a solution for stars, a special kind of trees. It still remains an open problem to show how difficult it is to calculate a game position for arbitrary trees. If one can compute this, solving the problem for a forest is as simple as adding the Grundy values for the single trees together in the usual manner.

References

1. Elwyn R. Berlekamp, John H. Conway, and Richard K. Guy. *Winning Ways for Your Mathematical Plays.* A K Peters, Ltd., 2nd edition, 2001.
2. Hans L. Bodlaender and Dieter Kratsch. Kayles and Nimbers. In *J. Algorithms*, volume 43 (1), pages 106–119, 2002.
3. John H. Conway. *On Numbers and Games.* A K Peters, Ltd., 2nd edition, 2001.
4. Aviezri. S. Fraenkel. Recent results and questions in combinatorial game complexities, Expanded version of invited lecture at Ninth Australasian Workshop on Combinatorial Algorithms, Perth, Western Australia, 27—30 July 1998. In *Proc. AWOCA98 (C. Iliopoulos, ed.)*, pages 124–146, 1998.

5. Richard K. Guy and Cedric A. B. Smith. The G-values of various games. In *Proc. Cambridge Philos. Soc.*, volume 52, pages 514–526, 1956.
6. Mikio Kano. Edge-removing games of star type. In *Discrete Mathematics*, volume 151, pages 113–119, 1996.
7. Thomas J. Schaefer. On the complexity of some two-person perfect-information games. In *Journal of Computer and System Sciences*, volume 16, pages 185–225, 1978.
8. Aaron Siegel. *Combinatorial Game Suite*. http://cgsuite.sourcefourge.net/, 2003.