FIBONACCI-LIKE SEQUENCES FOR VARIANTS OF THE TOWER OF HANOI, WITH CORRESPONDING GRAPHS AND GRAY CODES

BENOÎT RITTAUD

Abstract. We modify the rules of the classic Tower of Hanoi puzzle in a way to get the Fibonacci sequence involved in the optimal algorithm of resolution and show some nice properties of such a variant. In particular, we deduce from this *Tower of Hanoi-Fibonacci* a Gray-like code on the set of binary words without the factor 11, which has some properties interesting in its own right, and from which an iterative algorithm for the Tower of Hanoi-Fibonacci is obtained. Such an algorithm involves the Fibonacci substitution. Eventually, we briefly extend the study to some natural generalizations.

1. Introduction

The Tower of Hanoi is a puzzle invented by Édouard Lucas [8, 9] in which a set of n disks of different radii from 1 to n are put on a peg A in decreasing order, thus materializing a triangular tower. Two other pegs, B and C, are empty. The aim of the game is to move all the disks on peg C (or, in a roughly equivalent version, on either B or C), following the two rules: (1) the disks are moved one at a time, taking a disk on top of a peg and putting it on top of another peg, and (2) a disk cannot be put over a smaller disk. This is what we will call the classic Tower of Hanoi puzzle. A set-theoretic description of the puzzle is as follows: Write either d_k or k for the disk of radius k, and $\Delta_k = \{1, \dots, k\}$ for the set of the k smallest disks (with $\Delta_k = \varnothing$ for $k < 1$). Any state of the puzzle corresponds to an ordered 3-partition of Δ_n, written as (A, B, C) (it will be of convenience here to write with the same letter a peg and the set of disks on it). Such a partition is referred to as a *state* of the puzzle, also referred to as a *regular state* in the literature, especially when it is necessary to emphasize that disks on each peg have to be set up with decreasing size (an assumption that is weakened in some studies but that will not be considered here). A move from such a state to another one, say (A', B', C'), is allowed if and only if the two ordered partitions are equal up to some $d \in \Delta_n$ such that $d \in \{\min(A), \min(B), \min(C)\} \cap \{\min(A'), \min(B'), \min(C')\}$.

Many variants of the puzzle have been proposed since Lucas's original one. We point the reader to the highly valuable book [5] for a general synthesis on the subject.

Lucas already understood that the Tower of Hanoi was deeply linked to numeration systems. Indeed, he wrote in 1893 [10, p. 58] that

> Increasing the number of pegs and slightly modifying the rule of the game would easily provide representations of all numeration systems. [*En augmentant le nombre de tiges et en modifiant légèrement les règles du jeu, on obtiendrait facilement des représentations de tous les systèmes de numération.*]

The optimal algorithm to solve the puzzle with n disks requires $2^n - 1$ moves (hence passes through 2^n states), and the total number of admissible states is 3^n. (There exists a "worst" algorithm that solves the puzzle passing through all 3^n states exactly once.) Hence, it is not a surprise that there are natural links between the Tower of Hanoi and binary and ternary numeration systems. At Lucas's time, only integral numeration systems were known. Because the "worst" solution (i.e., the solution that passes through all possible states) requires $3^n - 1$ moves, it is sensible to ask for more pegs to represent other numeration systems. But now that noninteger numeration systems are known, we can give to Lucas's sentence a new meaning, keeping the three initial pegs and modifying the rules of the game, to get a Tower of Hanoi version of some nonconventional ways to write integers.

A first possibility consists in restricting the moves allowed between pegs. For example, we can forbid any direct move from A to C and from C to A. It is well-known that this constraint leads to the "worst" algorithm mentioned beforehand, that visits every possible state of the puzzle among the 3^n ones. Each of the possible variants of this kind is linked to some numeration system (as well as to some Gray code) defined by a linear recurring sequence (see [12]).

The initial question that gave rise to the present article was the converse: find natural alternative rules for the Tower of Hanoi such that the minimal number of moves required to solve the puzzle with n disks corresponds to a sequence fixed *a priori*. One of these sequences for which an answer can be found is the Fibonacci sequence, and we will show that the answer described here extends to some other linear recurring sequences as well.

Apart from the sequence of minimal moves, other links between the Tower of Hanoi and the Fibonacci sequence can be made. In particular, it is shown in [4] that, for the classic puzzle with n disks, the number of *key states* (i.e., for which the minimal number of moves to reach $(\varnothing, \varnothing, \Delta_n)$ is exactly twice the minimal number of moves to reach $(\Delta_n, \varnothing, \varnothing)$) is equal to F_{n-1}. The same article mentions the following other result, due to Merryfield and published in [2]: for any n, the set of distinct A_k (resp. B_k, C_k) attained during the optimal algorithm of resolution of the standard puzzle is of cardinality F_{n+2} (resp. F_{n+1}, F_{n+2}).

The present paper is organized as follows. In Section 2, we recall the relevant properties of the classic Tower of Hanoi we wish to generalize. Section 3 is devoted to the main variant we are interested in. In this variant, *Fibonacci moves* are defined, involving the sets Δ_k in a way that shares some similarity with the *switching Tower of Hanoi* rules defined in [7]. We prove that the variant defined by these kind of moves, the *Tower of Hanoi-Fibonacci*, is optimally solved in a number of moves essentially given by the Fibonacci sequence (Section 3.2). Then, we provide a link with the classic Zeckendorf-Fibonacci numeration system (Section 3.2) and deduce from it an iterative algorithm for the Tower of Hanoi-Fibonacci. We then study a Gray-like code associated with this numeration system (Section 3.3), then investigate the general properties of the graph associated with the puzzle (Section 3.4). Eventually, in Section 4, we briefly investigate some generalizations and questions, in two directions. The first one is when the definition of Fibonacci moves is modified to get an optimal algorithm that requires a number of moves given by a linear recurring sequence of the form $m_n = m_{n-p} + m_{n-q} + 1$. The second one considers complementary restrictions on moves between pegs, which gives rise to a Tribonacci-like sequence.

2. The Classic Tower of Hanoi

In the following, a subset $\{d_{k_1}, \ldots, d_{k_i}\}$ of Δ_n with $k_1 < \cdots < k_i$ is simply written $k_1 \cdots k_i$.

It is known since Lucas that the Tower of Hanoi has a solution for any $n \geqslant 0$, and that there is a unique solution with minimal number m_n of moves. Such a solution can be described recursively by the following decomposition, valid for any $n \geqslant 1$, from which we can deduce that $m_n = 2m_{n-1} + 1$, hence $m_n = 2^n - 1$ (because $m_0 = 0$):

$$(\Delta_n, \varnothing, \varnothing) \xrightarrow{m_{n-1}} (n, \Delta_{n-1}, \varnothing) \xrightarrow{1} (\varnothing, \Delta_{n-1}, n) \xrightarrow{m_{n-1}} (\varnothing, \varnothing, \Delta_n).$$

In this decomposition (as well as in all others in the following), the number written above an arrow corresponds to the number of moves involved.

To prove that this is indeed the (only) optimal solution, it is enough to prove that, in an optimal solution, the disk d_n moves exactly once. From now, we write $X \sqcup Y \sqcup Z$ for an ordered partition $\{X, Y, Z\}$ of $\{d_1, \dots, d_n\}$ in which the order of the elements X, Y, and Z is not known. Also, we write $X \sqcup Y \sqcup Z \longrightarrow R \sqcup S \sqcup T$ to define a move (or a sequence of moves) in which the position of the element R (resp. S, T) of the final partition is the same as the position of the element X (resp. Y, Z) in the initial partition. The rules of the puzzle imply that any move of d_n is of the form $d_n \sqcup \Delta_{n-1} \sqcup \varnothing \xrightarrow{1} \varnothing \sqcup \Delta_{n-1} \sqcup d_n$. Let $\mathcal{S}$ be the set of all partitions of the form $d_n \sqcup \Delta_{n-1} \sqcup \varnothing$, and let M be the number of moves of the disk d_n in an optimal solution. Hence, there are $2M$ elements of $\mathcal{S}$, written S_i for $i \in \{1, \dots, 2M\}$, such that our optimal solution can be decomposed in the following way:

$$(\Delta_n, \varnothing, \varnothing) =: \mathcal{S}_0 \longrightarrow \mathcal{S}_1 \xrightarrow{1} \mathcal{S}_2 \longrightarrow \cdots \longrightarrow \mathcal{S}_{2M-1} \xrightarrow{1} \mathcal{S}_{2M} \longrightarrow \mathcal{S}_{2M+1} := (\varnothing, \varnothing, \Delta_n).$$

The optimality assumption and the symmetry of the pegs implies that, for any $0 \leqslant i \leqslant M$, the number of moves corresponding to $\mathcal{S}_{2i} \longrightarrow \mathcal{S}_{2i+1}$ is equal to m_{n-1}. Hence, we have $m_n = (M+1)m_{n-1} + M$, so the optimal choice corresponds to $M = 1$, as expected.

Because the number of states during the optimal resolution of the puzzle is $1 + m_n = 2^n$, it is natural to consider the binary expansion of length exactly n to code the successive states from 0 to $2^n - 1$. It is easily proved by induction that the index k (between 1 to n) of the leftmost changing digit from the binary expansion of i to the binary expansion of $i+1$ corresponds to the disk that is moved when going from the state i to the state $i+1$. As a consequence, we have that, for any $1 \leqslant k \leqslant n$, the number of times the disk d_k is moved is 2^{n-k}.

Also, consider the alternative coding of the states of the puzzle with n disks by elements of $\{0,1\}^n$ given by the following rules: the initial state $(\Delta_n, \varnothing, \varnothing)$ is labelled 0^n, and when we go from partition (A, B, C) to (A', B', C') by moving the disk $d = d_k$, the label of the state (A', B', C') is defined as the label of (A, B, C) in which the kth digit has been switched (that is: this digit becomes a 0 if it was a 1 and a 1 if it was a 0). Then, an induction shows that the sequence of codings of the successive states thus obtained coincides with the *reflected binary Gray code*, that is: the list $\mathcal{G}_n$ of all binary words with exactly n letters defined recursively by $\mathcal{G}_0 = \{0\}$ and $\mathcal{G}_n = 0\mathcal{G}_{n-1} + 1\overline{\mathcal{G}_{n-1}}$ (where, for a sequence $\mathcal{L} = \{x_1, \dots, x_k\}$ of words and d a letter, $d\mathcal{L} = \{dx_1, \dots, dx_n\}$ and, with $\mathcal{M} = \{y_1, \dots, y_\ell\}$, the notation $\mathcal{L} + \mathcal{M}$ stands for $\{x_1, \dots, x_k, y_1, \dots y_\ell\}$). The fundamental property of such a list $\mathcal{G}_n$ is that any two consecutive elements of the list differ by exactly one digit.

It was observed in [6] that the graph $\mathcal{H}_n = (V_n, E_n)$ of the classic Tower of Hanoi has a fractal structure similar to the Sierpiński triangle, $\mathcal{H}_n$ being made of three copies of $\mathcal{H}_{n-1}$ for any $n \geqslant 1$, any two of these copies being linked by a single edge corresponding to a move of the form $n \sqcup \Delta_{n-1} \sqcup \varnothing \longrightarrow \varnothing \sqcup \Delta_{n-1} \sqcup n$ (see Figure 1).

Eventually, the optimal solution of the puzzle can be described by the following algorithm: move d_1 (always in the way $A \to B \to C \to A$ if n is odd, and in the way $A \to C \to B \to A$ if n is even), then, while there is a disk $d_k \neq d_1$ that can be moved, move that disk, then move again d_1.

3. The Tower of Hanoi-Fibonacci

3.1. **Definition and Optimal Algorithm.**

Definition 3.1. *Let X and Y be two different pegs such that, for some $k \in \Delta_n$, we have $X = k\tilde{X}$ and $Y = \Delta_{k-1}\tilde{Y}$. Write Z for the third peg. We define a k-*Fibonacci move *as a*

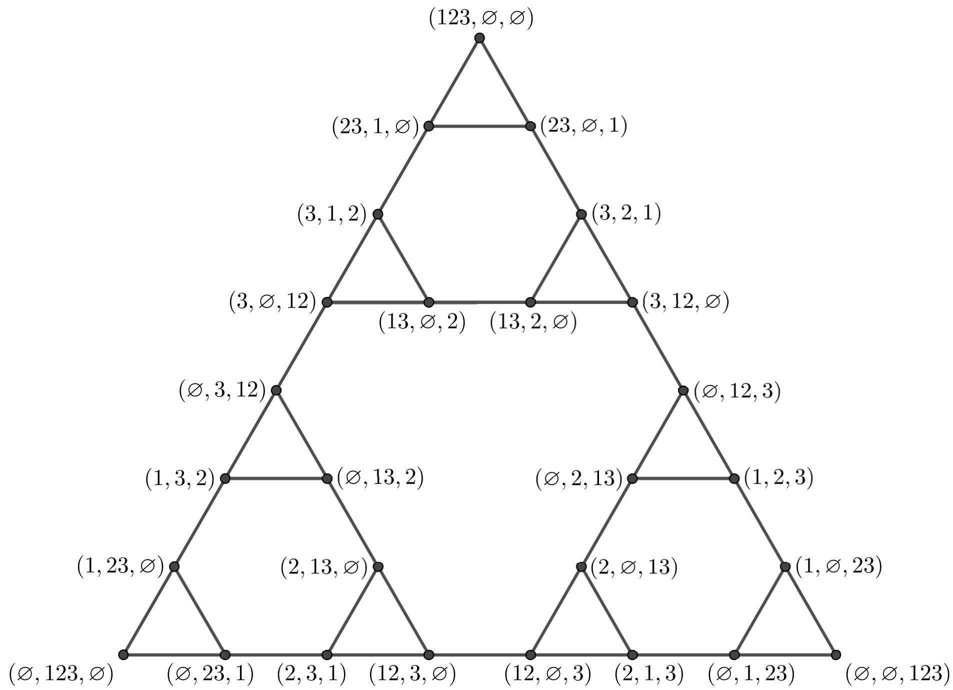

FIGURE 1. The graph $\mathcal{H}_3$ of the classic Tower of Hanoi with 3 disks.

move that consists in putting simultaneously $k-1$ and k onto Z, i.e.:

$$k\tilde{X} \sqcup \Delta_{k-1}\tilde{Y} \sqcup Z \longrightarrow \tilde{X} \sqcup \Delta_{k-2}\tilde{Y} \sqcup (k-1)kZ.$$

A Fibonacci move *is a k-Fibonacci move for some k. The* Tower of Hanoi-Fibonacci *is the Tower of Hanoi puzzle in which only Fibonacci moves are allowed. (Note that this definition will be slightly modified in Section 3.4.)*

Note that the 1-Fibonacci move is the one in which only one disk is moved (the disk d_1). Hence, this move is the only one common to the Tower of Hanoi-Fibonacci and the classic Tower of Hanoi. Figure 2 provides an example of a 3-Fibonacci move.

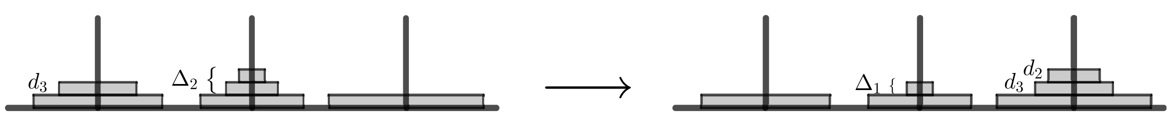

FIGURE 2. The 3-Fibonacci move from $(35, \Delta_2 4, 6)$ to $(5, \Delta_1 4, 236)$.

Theorem 3.2. *The Tower of Hanoi-Fibonacci with n disks admits a solution for any $n \geqslant 0$. There is only one optimal algorithm for it, that needs exactly $F_{n+2}-1$ Fibonacci moves (hence passes through F_{n+2} different states).*

As an example, here is the optimal solution in the case $n = 5$.

$$(\Delta_5, \varnothing, \varnothing) \longrightarrow (2345, \varnothing, 1) \longrightarrow (345, 12, \varnothing) \longrightarrow (45, 1, 23) \longrightarrow$$

$$(45, \varnothing, 123) \longrightarrow (5, 34, 12) \longrightarrow (15, 34, 2) \longrightarrow (5, 1234, \varnothing) \longrightarrow$$

$$(\varnothing, 123, 45) \longrightarrow (\varnothing, 23, 145) \longrightarrow (12, 3, 45) \longrightarrow (1, \varnothing, 2345) \longrightarrow (\varnothing, \varnothing, \Delta_5)$$

Proof. The proof is similar to the classic case. First, the cases $n = 0$ and $n = 1$ admit trivial solutions, with $m_0 = 0 = F_2 - 1$ and $m_1 = 1 = F_3 - 1$, where m_n stands for the minimal number of moves to solve the puzzle with n disks. Now, put $n \geqslant 2$ and assume that the puzzle with $n-1$ and $n-2$ disks are both solvable, with $m_{n-2} = F_n - 1$ and $m_{n-1} = F_{n+1} - 1$. Consider the puzzle with n disks. To be moved, the disk of radius n needs to be alone on its peg, and needs the tower Δ_{n-1} to be on another peg. A similar reasoning as the one made for the classic puzzle (see Section 2) shows that, in an optimal solution, the disk d_n moves exactly once. Therefore, we get a recursive description of the optimal solution of the puzzle with n disks:

$$(\Delta_n, \varnothing, \varnothing) \stackrel{m_{n-1}}{\longrightarrow} (n, \Delta_{n-1}, \varnothing) \stackrel{1}{\longrightarrow} (\varnothing, \Delta_{n-2}, (n-1)n) \stackrel{m_{n-2}}{\longrightarrow} (\varnothing, \varnothing, \Delta_n).$$

Hence, we have that $m_n = m_{n-1} + 1 + m_{n-2}$, so, by the induction hypothesis, $m_n = (F_{n+1} - 1) + 1 + (F_n - 1) = F_{n+2} - 1$, as required. □

3.2. **Link with the Zeckendorf-Fibonacci Numeration System.** The classic link between binary numeration system and the standard Tower of Hanoi extends to the *Zeckendorf* (or *Zeckendorf-Fibonacci*) *numeration system* and the Tower of Hanoi-Fibonacci. This link will provide us with an iterative algorithm for the optimal solution of the latter puzzle.

Recall that, as proved in [13], for any fixed $n \geqslant 2$ and any integer $0 < k < F_{n+1}$, there exists a unique finite sequence $(u_i)_{2 \leqslant i \leqslant n} \in \{0,1\}^{n-1}$ such that $u_i u_{i+1} = 0$ for any i and $k = \sum_{2 \leqslant i \leqslant n} u_i F_i$. Such a sequence is the *Zeckendorf-Fibonacci expansion* of k, $Z(k)$, also written as $[u_n \cdots u_2]_F$. When we need the length of a Zeckendorf-Fibonacci expansion to be of a certain kind (as in the following theorem), we allow ourselves to append leading 0s to it, hence considering $[0u_n \cdots u_2]_F$ as equivalent to $[u_n \cdots u_2]_F$.

In the following, a *ZF-sequence* (or *ZF-word*) will denote any binary sequence (or binary word) satisfying the property that it does not contain 11 anywhere in its terms. Two ZF-words like $u_n \cdots u_2$ and $0u_n \cdots u_2$ will be regarded as equivalent. Under this equivalence relation, the Zeckendorf-Fibonacci expansion of k is unique. Moreover, this expansion defines a bijection from $\mathbb{N}^* := \mathbb{N} \backslash \{0\}$ onto the set of (nonempty) ZF-sequences.

The Zeckendorf-Fibonacci expansion of $k > 0$ can be obtained by the application of the following algorithm:

- $r := k$, $u_i := 0$ for all $i \geqslant 2$, $n := \max(j \geqslant 2 \;:\; F_j \leqslant k)$
- while $r > 0$:
  - $i := \max(j \;:\; F_j \leqslant r)$
  - $u_i := 1$
  - $r := r - F_i$
- return$((u_i)_{2 \leqslant i \leqslant n})$.

Theorem 3.3. *Let $n \geqslant 0$ be some integer, let $0 < k < F_{n+2}$, and let $k - 1 = [u_{n+1} \cdots u_2]_F$ and $k = [v_{n+1} \cdots v_2]_F$. Let j be the largest index such that $u_j \neq v_j$. The kth move of the optimal solution of the Tower of Hanoi-Fibonacci puzzle with n disks is a $(j-1)$-Fibonacci move.*

Proof. This is a simple induction making use of the recursive description of the algorithm

$$(\Delta_n, \varnothing, \varnothing) \longrightarrow (n, \Delta_{n-1}, \varnothing) \longrightarrow (\varnothing, \Delta_{n-2}, (n-1)n) \longrightarrow (\varnothing, \varnothing, \Delta_n).$$

The property is true for $n = 0$ and $n = 1$. Assume it is true for $n-2$ and $n-1$ for some $n \geqslant 2$. The moves from $(\Delta_n, \varnothing, \varnothing)$ to $(n, \Delta_{n-1}, \varnothing)$ are moves from 1 to $m_{n-1} = F_{n+1} - 1$, so their Zeckendorf-Fibonacci expansion of length n are all of the form $[0u_n \cdots u_2]_F$. Hence, by the induction hypothesis on the puzzle with $n-1$ disks, the property is true for all these moves.

Now, the Fibonacci move $(n, \Delta_{n-1}, \varnothing) \longrightarrow (\varnothing, \Delta_{n-2}, (n-1)n)$ is the F_{n+1}th one, of Zeckendorf-Fibonacci expansion of length n equal to $[10\cdots 0]_F$. The largest moving disk in this move is the nth one, and the largest index j as defined in the theorem is equal to $n+1$, so the theorem is also valid for this move.

The remaining F_n moves are the ones with Zeckendorf-Fibonacci expansion of length n of the form $[10u_{n-1}\cdots u_2]_F$, where $[u_{n-1}\cdots u_0]_F$ is the Zeckendorf-Fibonacci expansion of length $n-2$ of $k - F_{n+1}$. Hence, we can apply the induction hypothesis on the puzzle with $n-2$ disks, and we are done. □

Theorem 3.4. *Let $0 < k \leqslant n$ be two integers. For the Hanoi-Fibonacci puzzle with n disks, the number of k-Fibonacci moves in the optimal algorithm is equal to F_{n+1-k}.*

Proof. We proceed by induction on n and (decreasing) induction on k. For $k = n$, the number we are looking for is equal to 1, which corresponds indeed to $F_{n+1-k} = F_1 = 1$. For $k = n-1$, it is easy to check that there is also exactly one k-Fibonacci move, a number equal to $F_{n+1-k} = F_2$. Now, for $k < n-1$, by the induction hypothesis, the part $(\Delta_n, \varnothing, \varnothing) \longrightarrow (n, \Delta_{n-1}, \varnothing)$ of the algorithm involves a number of k-Fibonacci moves equal to $F_{(n-1)+1-k} = F_{n-k}$, and the part $(\varnothing, \Delta_{n-2}, (n-1)n) \longrightarrow (\varnothing, \varnothing, \Delta_n)$ involves $F_{(n-2)+1-k} = F_{n-k-1}$ moves, hence a total equal to $F_{n-k} + F_{n-k-1} = F_{n+1-k}$ moves. □

Theorem 3.3 provides a complete iterative algorithm for the Tower of Hanoi-Fibonacci, with the only issue that, for 1-Fibonacci moves (i.e., a move of the single disk d_1), one has to determine on which peg the disk d_1 has to move. Here is an answer to this question. Let us say that d_1 is moving to the right (resp. to the left) whenever it moves from A to B, from B to C or from C to A (resp. from A to C, from B to A or from C to B). Thus, by Theorem 3.4, we can code the sequence of 1-Fibonacci moves for the puzzle with n disks as a word $\mu_n \in \{l, r\}^{F_n}$, where l denotes a move to the left and r a move to the right. We then have the following result.

Theorem 3.5. *In the optimal algorithm for the Tower of Hanoi-Fibonacci:*

- *if $n \in 2\mathbb{N}^*$, then the kth letter of μ_n is an r if and only if $Z(k)$ has an even number of 1s;*
- *if $n \notin 2\mathbb{N}^*$, then the kth letter of μ_n is an r if and only if $Z(k)$ has an odd number of 1s.*

Proof. We proceed by induction on $n \geqslant 2$. Write the decomposition of the optimal solution of the puzzle, with the corresponding number of 1-Fibonacci moves (given by Theorem 3.4 with $k = 1$) above each arrow.

$$(\Delta_n, \varnothing, \varnothing) \overset{F_{n-1}-1}{\longrightarrow} (n, \Delta_{n-1}, \varnothing) \overset{0}{\longrightarrow} (\varnothing, \Delta_{n-2}, (n-1)n) \overset{F_{n-2}-1}{\longrightarrow} (\varnothing, \varnothing, \Delta_n).$$

Assume, for example, $n \in 2\mathbb{N}^*$ (the other case would be similar). Consider the kth letter of μ_n corresponding to a 1-Fibonacci move among the $F_{n-1} - 1$ first ones. By the induction hypothesis, that $n-1$ is odd, and that the 1-Fibonacci moves corresponding to $(\Delta_n, \varnothing, \varnothing) \longrightarrow (n, \Delta_{n-1}, \varnothing)$ are the same as the one for $(\Delta_n, \varnothing, \varnothing) \longrightarrow (n, \varnothing, \Delta_{n-1})$, but with exchanging the rs and the ls, we have that the considered kth letter is an r if and only if $Z(k)$ has an even number of 1s. For a value of k corresponding to a 1-Fibonacci move among the last $F_{n-2} - 1$ Fibonacci moves, the reasoning is the same, with the additional consideration that the Zeckendorf expansion of k is now of the form $[10u_{n-3}\cdots u_2]_F$. □

Let us also mention the following qualitative results that show in particular that the number of 1-Fibonacci moves to the left and to the right are as balanced as possible.

Corollary 3.6. *We have* $\mu_0 = \varnothing$, $\mu_1 = l$, *and for any* $n \geqslant 2$, $\mu_n = (\mu_{n-1}\mu_{n-2})^*$ *(hence* $\mu_n = \mu_{n-2}\mu_{n-3}\mu_{n-3}\mu_{n-4}$ *for* $n \geqslant 4$*), where* μ^* *stands for the word in which each* l *has been replaced by an* r *and each* r *by an* l*. Moreover, denoting by* $|\mu|_d$ *the number of letters* $d \in \{r, l\}$ *in the word* μ*, we have*

$$|\mu_{3n}|_r = |\mu_{3n}|_l, \qquad |\mu_{3n+1}|_l - |\mu_{3n+1}|_r = 1, \qquad \text{and} \qquad |\mu_{3n+2}|_r - |\mu_{3n+2}|_l = 1.$$

Proof. The first part is proved by an induction on n and the following decomposition of the optimal solution of the puzzle (for $n \geqslant 2$), in which the words on the arrows stand for the sequences of moves of d_1 during 1-Fibonacci moves:

$$(\Delta_n, \varnothing, \varnothing) \xrightarrow{\mu^*_{n-1}} (n, \Delta_{n-1}, \varnothing) \longrightarrow (\varnothing, \Delta_{n-2}, (n-1)n) \xrightarrow{\mu^*_{n-2}} (\varnothing, \varnothing, \Delta_n).$$

The second part is a simple induction on n. □

3.3. **Gray-like Code.** To complete notations set up for lists in Section 2, for any list $\mathcal{L}$ of elements of $\{0,1\}^n$, write $'\mathcal{L}$ (resp. $\mathcal{L}'$) for the list made of all elements of $\mathcal{L}$ in which the leftmost (resp. rightmost) letter of each element is removed. Then, set $\mathcal{N}_0 = \varnothing$, $\mathcal{N}_1 := \{1\}$ and $\mathcal{N}_n := 10\overline{'\mathcal{N}_{n-1}} + 10\overline{\mathcal{N}_{n-2}}$ (here writing each word always with a 1 as leftmost digit). Eventually, let $\mathcal{G} := \sum_{i \geqslant 0} \mathcal{N}_i := \{g_1, g_2, \ldots\}$. Such a construction may be seen as a mirroring process analogous to the classic one for binary Gray codes, as shown in Table 1.

$g_1 =$	0	0	0	0	0	1	$\mathcal{N}_1$
$g_2 =$	0	0	0	0	1	0	$\mathcal{N}_2$
$g_3 =$	0	0	0	1	0	0	$\mathcal{N}_3$
$g_4 =$	0	0	0	1	0	1	
$g_5 =$	0	0	1	0	0	1	$\mathcal{N}_4$
$g_6 =$	0	0	1	0	0	0	
$g_7 =$	0	0	1	0	1	0	
$g_8 =$	0	1	0	0	1	0	$\mathcal{N}_5$
$g_9 =$	0	1	0	0	0	0	
$g_{10} =$	0	1	0	0	0	1	
$g_{11} =$	0	1	0	1	0	1	
$g_{12} =$	0	1	0	1	0	0	
$g_{13} =$	1	0	0	1	0	0	$\mathcal{N}_6$
$g_{14} =$	1	0	0	1	0	1	
$g_{15} =$	1	0	0	0	0	1	
$g_{16} =$	1	0	0	0	0	0	
$g_{17} =$	1	0	0	0	1	0	
$g_{18} =$	1	0	1	0	1	0	
$g_{19} =$	1	0	1	0	0	0	
$g_{20} =$	1	0	1	0	0	1	

(g_1 to g_{20} form $\mathcal{G}_6$.)

TABLE 1. The Gray-like code of the Tower of Fibonacci-Hanoi with $n = 6$ disks.

Recall that the *Hamming distance* between $w, w' \in \{0,1\}^n$, written $h(w,w')$, is the number of their different digits, that is: for $w = w_1 \cdots w_n$ and $w' = w'_1 \cdots w'_n$, we have $h(w,w') = \text{card}(1 \leqslant i \leqslant n \ : \ w_i \neq w'_i)$. When w and w' do not have the same number of letters, we append as many 0s as necessary to the shortest one to make it of the same length as the other.

Theorem 3.7. *The set $\mathcal{G} = \{g_1, g_2, \ldots\}$ is in bijection with the set of all nonempty ZF-words (assuming the equivalence between $u_n \cdots u_2$ and $0u_n \cdots u_2$). Moreover, for any $n \geqslant 1$, we have*

$$h(g_n, g_{n+1}) = \begin{cases} 2, & \text{if } n+1 = F_k \text{ for some } k \geqslant 3; \\ 1, & \text{otherwise.} \end{cases}$$

For any $n \geqslant 1$, the list $\mathcal{N}_n$ is made of all the ZF-words of length n (with a 1 as a leftmost digit), each appearing exactly once.

Proof. By induction, assume that $'\mathcal{N}_{n-1} \cup \mathcal{N}_{n-2}$ contains all ZF-words of $\{0,1\}^{n-2}$ exactly once, $'\mathcal{N}_{n-1}$ (resp. $\mathcal{N}_{n-2}$) containing those with a 0 (resp. a 1) as the leftmost digit. Hence, by the definition of $\mathcal{N}_n$, $\mathcal{N}_n$ contains all ZF-words in $\{0,1\}^n$ with 10 as the leftmost digits, each exactly once. As a consequence, by the induction hypothesis, $'\mathcal{N}_n \cup \mathcal{N}_{n-1}$ contains all ZF-words of $\{0,1\}^{n-1}$ exactly once, with $'\mathcal{N}_n$ (resp. $\mathcal{N}_{n-1}$) containing those with a 0 (resp. a 1) as the leftmost digit. Hence, $\mathcal{G}$ is in bijection with the set of all ZF-words (but the empty one).

Now consider the property of Hamming distances. By induction, assume the property is satisfied for $\mathcal{G}_{n-1}$ and that the Hamming distance among two successive elements of $\mathcal{N}_k$ is always equal to 1. We have $\mathcal{G}_n = 0\mathcal{G}_{n-1} + \mathcal{N}_n = 0\mathcal{G}_{n-1} + 10\overline{'\mathcal{N}_{n-1}} + 10\overline{\mathcal{N}_{n-2}}$. Hence, by the induction hypothesis, the Hamming distance of any pair of successive elements of $\mathcal{G}_n$ is equal to 1, apart, possibly, for the pairs $(0g_{F_{n+1}-1}, g_{F_{n+1}})$ (made by the last element of $\mathcal{G}_{n-1}$ and the following one in $\mathcal{G}_n$) and $(g_{F_{n+1}+F_{n-1}-1}, g_{F_{n+1}+F_{n-1}})$ (made by the last element of $\mathcal{G}_{n-1} + 10\overline{'\mathcal{N}_{n-1}}$ and the following one in $\mathcal{G}_n$).

To prove that the induction hypothesis is valid also for n, it only remains to prove that the Hamming distance of the first of these pairs is equal to 2 and the one of the second pair is equal to 1. For the first one, by the definition of $\mathcal{N}_n$, the two different digits of $0g_{F_{n+1}-1}$ and $g_{F_{n+1}}$ are the two leftmost ones, that is: writing $0g_{F_{n+1}-1}$ in the form $010u_{n-3} \cdots u_1$, we have $g_{F_{n+1}} = 100u_{n-3} \cdots u_1$. Hence, their Hamming distance is indeed equal to 2. For the second pair, by construction we have (writing w' for the word w from which the leftmost digit has been removed) $g_{F_{n+1}+F_{n-1}-1} = 10g'_{F_{n+1}-F_{n-1}} = 10g'_{F_n}$ and $g_{F_{n+1}+F_{n-1}} = 10g_{F_n-1}$. As already noted, $g_{F_n} = 0100u_{n-4} \cdots u_1$ and $g_{F_n-1}0010u_{n-4} \cdots u_1$, so $10g'_{F_n} = 1000u_{n-4} \cdots u_1$ and $10g_{F_n-1} = 1010u_{n-4} \cdots u_1$, hence their Hamming distance is equal to 1. □

Strictly speaking, a Gray code has the property that two consecutive elements are always of Hamming distance equal to 1, hence our list $\mathcal{G}$ is only a Gray-like code. One may wonder if we could recover a real Gray code that lists all the ZF-words. By itself, such a question is too large, and a natural restriction on it is to ask for such a Gray code to be *length-increasing*, that is: the list $(g_n)_{n \geqslant 1}$ should order the words in such a way that the leftmost 1 of g_n is of increasing index with n. It is easy to check that such a natural condition cannot be satisfied, because for w the last ZF-word of length $n-1$ and w' the first one of length n, we necessarily have $h(w,w') \geqslant 2$. This remark leads to the following result.

Theorem 3.8. *Let $(u_n)_{n>0}$ be a length-increasing sequence made of all nonnull ZF-words (each of them appearing exactly once). For any $n > 0$, we have $h(u_n, u_{n+1}) \geqslant h(g_n, g_{n+1})$.*

Basically, this result says that, among all orderings of the set of ZF-words satisfying the length-increasing property, the list $\mathcal{G}$ minimizes the Hamming distance between its consecutive elements.

As it is done in [11, Theorem 5] for another Gray-like code linked to Fibonacci combinatorics given in [3], it is possible to "de-mirror" the construction of $\mathcal{G}_n$, in the following sense.

Theorem 3.9. *Let $n > 1$ and let us define from the list $\mathcal{N}_{n-1}$, a new list by writing each element of $\mathcal{N}_{n-1}$ twice in a row if and only if this element ends with a 0. More precisely, the new list is defined by the following algorithm, where $\mathcal{N}_{n-1} = \{g_{F_n}, \dots, g_{F_{n+1}-1}\}$:*

- $\mathcal{L} := \varnothing$
- *for i from 0 to $F_{n-1} - 1$:*
  if g_{F_n+i} has a 0 as a rightmost digit then $\mathcal{L} := \mathcal{L} + \{g_{F_n+i}, g_{F_n+i}\}$,
  else $\mathcal{L} := \mathcal{L} + \{g_{F_n+i}\}$
- *return($\mathcal{L}$).*

The list thus obtained is equal to $\mathcal{N}'_n$. Moreover, for $n > 3$, there is a unique way to concatenate (on the right) to each element of $\mathcal{N}'_n$, a 0 or a 1 so as to get a list of ZF-words such that the Hamming distance of any consecutive words is equal to 1. The list obtained after this concatenation is equal to $\mathcal{N}_n$.

Before providing the proof, let us mention that the sequence of 0s and 1s to be concatenated on the right of the words of $\mathcal{N}'_n$ will be made mathematically explicit in Theorem 3.10 below. Also, a practical way to carry out these concatenations is to consider a word in the list $\mathcal{N}'_n$ ending with a 1 (we leave it to the reader to check that the first word of this kind appears in the third place of $\mathcal{N}'_n$ at the latest, for any $n \geqslant 3$): because it must remain a ZF-word in the process, we have to concatenate a 0 to it. Then, the other concatenations are made inductively by the following process, given that the Hamming distance between consecutive words must be equal to 1: a word $g \in \mathcal{N}'_n$ with d as the last digit concatenated to it being given, let g' be its successor (or its predecessor) in the list. If $g = g'$, then concatenate the digit $1 - d$ to g', otherwise concatenate d to it.

Proof of Theorem 3.9. The result can be checked for $n \leqslant 4$, so in the sequel we assume $n > 4$. We have that $\mathcal{N}_n = 10\overline{{}'\mathcal{N}_{n-1}} + 10\overline{\mathcal{N}_{n-2}}$, hence ${}'\mathcal{N}_n = 0\overline{{}'\mathcal{N}_{n-1}} + 0\overline{\mathcal{N}_{n-2}}$, and

$$\begin{aligned}
\mathcal{N}_n &= 10\overline{{}'\mathcal{N}_{n-1}} + 10\overline{\mathcal{N}_{n-2}} \\
&= 10\overline{(0\overline{{}'\mathcal{N}_{n-2}} + 0\overline{\mathcal{N}_{n-3}})} + 10\overline{(10\overline{{}'\mathcal{N}_{n-3}} + 10\overline{\mathcal{N}_{n-4}})} \\
&= 100(\mathcal{N}_{n-3} + {}'\mathcal{N}_{n-2}) + 1010(\mathcal{N}_{n-4} + {}'\mathcal{N}_{n-3}),
\end{aligned}$$

so

$${}'\mathcal{N}_n = 00(\mathcal{N}_{n-3} + {}'\mathcal{N}_{n-2}) + 010(\mathcal{N}_{n-4} + {}'\mathcal{N}_{n-3}).$$

Hence, writing $\mathcal{Z}_n := \mathcal{N}_n + {}'\mathcal{N}_{n+1}$, we have

$$\mathcal{Z}_n = 100\mathcal{Z}_{n-3} + 1010\mathcal{Z}_{n-4} + 00\mathcal{Z}_{n-2} + 010\mathcal{Z}_{n-3}.$$

For any list $\mathcal{L}$, write $\underline{\mathcal{L}}$ for the list $\mathcal{L}$ in which any sequence of the same element is replaced by a single occurrence of this element. To be precise, if $\mathcal{L} = \{\ell_1, \dots, \ell_k\}$, then $\underline{\mathcal{L}}$ is given by the following algorithm:

- $\underline{\mathcal{L}} := \{\ell_1\}$; $i := 1$
- for j from 2 to k:
  if $\ell_j \neq \ell_i$ then $\underline{\mathcal{L}} := \underline{\mathcal{L}} + \{\ell_j\}$ and $i := j$
- return($\underline{\mathcal{L}}$)

The first part of the theorem can therefore be rewritten as $\underline{\mathcal{N}'_n} = \mathcal{N}_{n-1}$. We prove this by induction, the induction hypothesis being: for some $n \geqslant 5$ and for any $1 < k < n$, we have $\underline{\mathcal{N}'_k} = \mathcal{N}_{k-1}$ and $\underline{\mathcal{Z}'_k} = \mathcal{Z}_{k-1}$. (The cases where $n < 5$ can be checked separately.) Consider $\mathcal{Z}_n$ first. Note that if two lists $\mathcal{L}_1$ and $\mathcal{L}_2$ have no common element, then $\underline{\mathcal{L}_1 + \mathcal{L}_2} = \underline{\mathcal{L}_1} + \underline{\mathcal{L}_2}$. In the equality $\mathcal{Z}'_n = 100\mathcal{Z}'_{n-3} + 1010\mathcal{Z}'_{n-4} + 00\mathcal{Z}'_{n-2} + 010\mathcal{Z}'_{n-3}$, the four lists on the right side are pairwise disjoint (as sets), because of the different prefixes of their elements. Hence, by the induction hypothesis, we have

$$\begin{aligned}
\underline{\mathcal{Z}'_n} &= \underline{100\mathcal{Z}'_{n-3}} + \underline{1010\mathcal{Z}'_{n-4}} + \underline{00\mathcal{Z}'_{n-2}} + \underline{010\mathcal{Z}'_{n-3}} \\
&= 100\underline{\mathcal{Z}'_{n-3}} + 1010\underline{\mathcal{Z}'_{n-4}} + 00\underline{\mathcal{Z}'_{n-2}} + 010\underline{\mathcal{Z}'_{n-3}} \\
&= 100\mathcal{Z}_{n-4} + 1010\mathcal{Z}_{n-5} + 00\mathcal{Z}_{n-3} + 010\mathcal{Z}_{n-4} \\
&= \mathcal{Z}_{n-1}.
\end{aligned}$$

Now, we have $\mathcal{N}_n = 100\mathcal{Z}_{n-3} + 1010\mathcal{Z}_{n-4}$. The lists $100\mathcal{Z}'_{n-3}$ and $1010\mathcal{Z}'_{n-4}$ are disjoint (because of their prefixes), so

$$\begin{aligned}
\underline{\mathcal{N}'_n} &= \underline{100\mathcal{Z}'_{n-3} + 1010\mathcal{Z}'_{n-4}} \\
&= 100\underline{\mathcal{Z}'_{n-3}} + 1010\underline{\mathcal{Z}'_{n-4}} \\
&= 100\mathcal{Z}_{n-4} + 1010\mathcal{Z}_{n-5} \\
&= \mathcal{N}_{n-1}.
\end{aligned}$$

Now, let us prove the following statement: *let $g \in \mathcal{N}_{n-1}$ with 0 as its rightmost digit (so $g0$ and $g1$ belong to $\mathcal{N}_n$). There is no word between $g0$ and $g1$ in $\mathcal{N}_n$. The same is true when replacing $\mathcal{N}$ by $\mathcal{Z}$ in this statement.* This is done by induction, the induction hypothesis being that the property is true for any $k < n$ for the lists $\mathcal{N}_k$ and $\mathcal{Z}_k$ (where $n \geqslant 5$). Consider the relation $\mathcal{Z}_n = 100\mathcal{Z}_{n-3} + 1010\mathcal{Z}_{n-4} + 00\mathcal{Z}_{n-2} + 010\mathcal{Z}_{n-3}$ and let $g0$ and $g1$ be in $\mathcal{Z}_n$. Because of the different prefixes on the right side, $g0$ and $g1$ belong to the same term of the right sum of lists, hence are consecutive words by the induction hypothesis. Now, because $\mathcal{N}_n = 100\mathcal{Z}_{n-3} + 1010\mathcal{Z}_{n-4}$, the same reasoning applies, and the statement is proved.

Hence, we know that the algorithm presented in the statement of the theorem indeed produces the list $\mathcal{N}'_n$. To prove the last statements in the theorem, consider the remark made right after the statement of Theorem 3.9. This remark shows that there exists a unique way to concatenate 0s and 1s to the right of the words in $\mathcal{N}'_n$ to get ZF-words of length n where two consecutive ones are of Hamming distance 1. Hence, because by Theorem 3.7 we have that in $\mathcal{N}_n$ two consecutive words are of Hamming distance 1, the concatenation process described in the remark indeed produces $\mathcal{N}_n$ from $\mathcal{N}'_n$, as required. □

Following [11] also leads to the following description that makes use of the Fibonacci substitution to describe the way digits are to be added to the elements of $\mathcal{N}_{n-1}$ to get $\mathcal{N}_n$.

Theorem 3.10. *Let σ be the* Fibonacci substitution *on the alphabet $\{\alpha, \beta\}$, defined by $\sigma(\alpha) = \alpha\beta$ and $\sigma(\beta) = \alpha$, and let $(\sigma_n)_{n\in\mathbb{N}} := \sigma^\infty(\alpha)$ be its fixed point. The sequence of rightmost digits in $\mathcal{G}$ is the sequence $(\tau_n)_{n\in\mathbb{N}^*}$ on the alphabet $\{0, 1\}$ defined by $\tau_1 = 1$ and, for any $n \geqslant 2$, $\tau_n = 1$ if and only if $\sigma_{\lfloor n/2 \rfloor - 1} = \beta$.*

In other words, the sequence of rightmost digits in $\mathcal{G}$ begins with a 1, then shows a sequence of 00s and 11s under the recoding of the fixed point of the Fibonacci substitution defined by $\alpha \longmapsto 00$ and $\beta \longmapsto 11$.

Proof. Write $(r_n)_{n\in\mathbb{N}^*}$ for the sequence of rightmost digits of $\mathcal{G}$. Our goal is to prove that $r_n = \tau_n$ for any n. This is easily checked for $n \leqslant 2$.

Let $n \geqslant 3$ and let k be the largest integer such that $F_{k+1} \leqslant n$ (so $g_n \in \mathcal{N}_k$). The word given by the rightmost letter of each element of $\mathcal{N}_m$ is $r_{F_{m+1}} \cdots r_{F_{m+2}-1}$, so the relation $\mathcal{N}_k = 10\overline{\mathcal{N}_{k-1}} + 10\overline{\mathcal{N}_{k-2}}$ gives $r_{F_{k+1}} \cdots r_{F_{k+2}-1} = r_{F_{k+1}-1} \cdots r_{F_k-1}$, so $r_n = r_{2F_{k+1}-1-n}$. Because $2F_{k+1} - 1 - n < n$, this equality, together with $r_1 = 1$ and $r_2 = 0$, is a complete recursive definition of $(r_n)_{n\in\mathbb{N}^*}$.

Because we also have $\tau_1 = 1$ and $\tau_2 = 0$, it only remains to show that $\tau_n = \tau_{2F_{k+1}-1-n}$ for any $n \geqslant 3$. This is a consequence of some standard properties of the Fibonacci substitution. First, the fixed point of σ can be described as the limit of the (eventually) increasing sequence $(B_j)_{j\in\mathbb{N}}$ of words in $\{\alpha, \beta\}$ such that $B_0 = \beta$, $B_1 = \alpha$, and $B_j = B_{j-1}B_{j-2}$ for any $j \geqslant 2$. For any $j \geqslant 2$, write B'_j for the block B_j in which the last two letters have been removed, and let us show that B'_j is palindromic for any j. It is true for B'_2 (which is empty) and for B'_3 (equal to α), so we can assume $j \geqslant 4$ so that all subsequent expressions are well-defined and correct. We write first B_j as $B_{j-2}B_{j-3}B_{j-2}$. Also, by an immediate induction, we have that $B_{j-2} = B'_{j-2}\alpha\beta$ for even values of j and $B_{j-2} = B'_{j-2}\beta\alpha$ for odd values of j. Hence, for j even (the case j odd being similar), $B_j = B'_{j-2}\alpha\beta B'_{j-3}\beta\alpha B'_{j-2}\alpha\beta$, so $B'_j = B'_{j-2}\alpha\beta B'_{j-3}\beta\alpha B'_{j-2}$, which is palindromic by induction.

Write C_j for the block obtained from B_j by the recoding $\alpha \longmapsto 00$ and $\beta \longmapsto 11$, and define C'_j from B'_j in the same way (or, equivalently, from removing from C_j its last four digits). The sequence $(\tau_n)_{n\in\mathbb{N}^*}$ is therefore the limit of the words $1C'_j$, where C'_j is palindromic. Also, by induction for any $j \geqslant 0$, we have $\mathrm{card}(B_j) = F_{j+1}$, so $\mathrm{card}(C'_j) = 2F_{j+1} - 4$ for $j \geqslant 2$.

Now, for any $n \geqslant 3$, let κ be an integer for which $n \leqslant 2F_{\kappa+1} - 3$ (that is, κ is chosen in such a way that there are at least n digits in the sequence $1C'_\kappa$). Because the initial 1 in $1C'_\kappa$ corresponds to τ_1, the symmetry given by the palindromicity of C'_κ means that $\tau_n = \tau_{2F_{\kappa+1}-1-n}$. This relation holds for any κ for which $n \leqslant 2F_{\kappa+1} - 3$. The value k defined previously as the largest integer such that $F_{k+1} \leqslant n$ is one of these κs, whenever we have $F_{k+2} - 1 \leqslant 2F_{k+1} - 3$ (because $n \leqslant F_{k+2} - 1$). This latter inequality holds for all $k \geqslant 4$. Thus, checking the equality $\tau_n = r_n$ for all $n \leqslant F_{4+2} - 1 = 7$, we get that both the sequences $(r_n)_{n\in\mathbb{N}^*}$ and $(\tau_n)_{n\in\mathbb{N}^*}$ have the same recursive definition with the same initial values, hence are equal. □

Eventually, let us write the complete "de-mirrored" algorithm given by Theorems 3.9 and 3.10 to get $\mathcal{N}_n$ from $\mathcal{N}_{n-1} = \{g_{F_n}, \ldots, g_{F_{n+1}-1}\}$:

- Initialization: $\mathcal{L} := \varnothing$
- for i from F_n to $F_{n+1} - 1$:
  - if g_i ends with a 1 then $\mathcal{L} := \mathcal{L} + \{g_i\}$
  - else $\mathcal{L} := \mathcal{L} + \{g_i\} + \{g_i\}$
- write $\mathcal{L} =: \{g_{F_{n+1}}, \ldots, g_{F_{n+2}-1}\}$
- for i from F_{n+1} to $F_{n+2} - 1$:
  - in $\mathcal{L}$, do $g_i := g_i\tau_i$
- return($\mathcal{L}$).

3.4. **The Hanoi-Fibonacci Graphs.**

In the present section, our aim is to investigate how to represent the Tower of Hanoi-Fibonacci by a graph $\mathcal{F}_n = (V_n, E_n)$, in which V_n contains the 3^n possible states of the puzzle, and E_n is the set of arcs (e, e') such that the move from e to e' is a Fibonacci move (Figure 3). Note that, contrary to the graph of the classic Tower of Hanoi, this graph is oriented because Fibonacci moves are not reversible (except for 1-Fibonacci moves).

Theorem 3.11. *For any $n \geqslant 2$, the unoriented graph that corresponds to $\mathcal{F}_n$ is nonplanar.*

Proof. Because $\mathcal{F}_n \subset \mathcal{F}_{n+1}$ for any n, it is sufficient to prove that $\mathcal{F}_2$ is nonplanar.

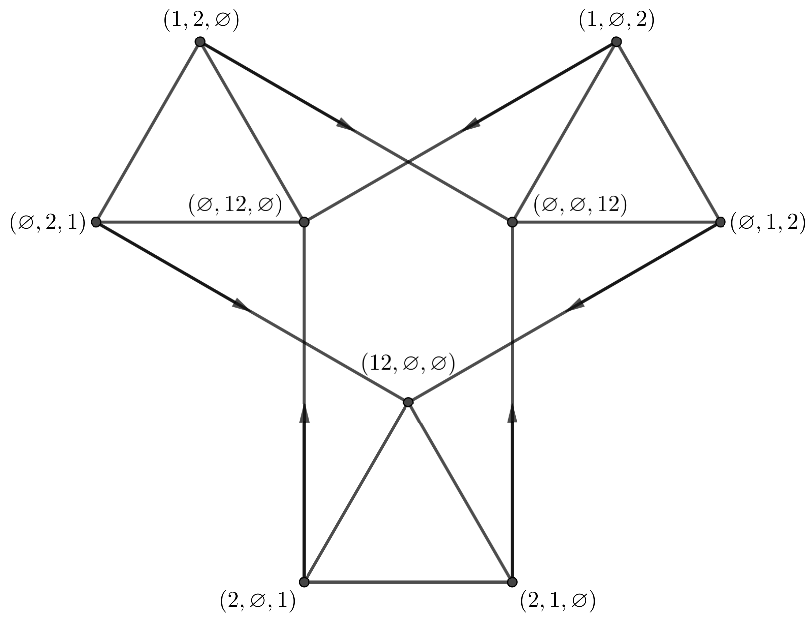

FIGURE 3. The graph $\mathcal{F}_2$ of the Tower of Hanoi-Fibonacci with 2 disks. (Line segments stand for arcs going both ways.)

Let us merge the vertices $(\varnothing, 2, 1)$ and $(1, 2, \varnothing)$ into a single vertex a, then the vertices $(1, \varnothing, 2)$ and $(\varnothing, 1, 2)$ to get a second vertex b, and eventually the vertices $(2, \varnothing, 1)$ and $(2, 1, \varnothing)$ to get a third vertex c. The graph thus obtained can be split into two subsets of vertices, $V = \{a, b, c\}$ (the merged "external" vertices) and $V' = \{(12, \varnothing, \varnothing), (\varnothing, 12, \varnothing), (\varnothing, \varnothing, 12)\}$ (the "internal" vertices). The set of edges of this new graph is the set of all possible edges between V and V', hence it is isomorphic to the complete bipartite graph $K_{3,3}$. Hence, $K_{3,3}$ is a minor of $\mathcal{F}_2$, so $\mathcal{F}_2$ is not planar. □

Theorem 3.12. *For any $n \geqslant 0$, $\mathcal{F}_n$ is strongly connected. In other words, any possible state of the puzzle can be attained from any other under the rules of the Tower of Hanoi-Fibonacci.*

Proof. We proceed by induction on n. Assume $\mathcal{F}_n$ is strongly connected, and consider $\mathcal{F}_{n+1}$. This latter graph contains exactly three copies of $\mathcal{F}_n$ that we denote by $\mathcal{F}_n^A$, $\mathcal{F}_n^B$, and $\mathcal{F}_n^C$ depending on the peg on which d_{n+1} is located in each. By the induction hypothesis, each of these three copies of $\mathcal{F}_n$ is strongly connected. Hence, to obtain the desired result, it is sufficient to prove that there exists an arc from some vertex of $\mathcal{F}_n^X$ to some vertex of $\mathcal{F}_n^Y$ for any different pegs X and Y. The Fibonacci move $(n+1) \sqcup \Delta_n \sqcup \varnothing \longrightarrow \varnothing \sqcup \Delta_{n-1} \sqcup n(n+1)$ provides such an arc. □

The previous drawing of $\mathcal{F}_2$ seems difficult to extend in a natural way to larger values of n, and a slight modification of Figure 1 seems more interesting for visualization purposes, even if it needs some specific coding to makes the arrow diagram handy. Also, as stated in Definition 3.1, it will be easier to work with a slightly modified version of a Fibonacci move, hereafter defined as

$$k\tilde{X} \sqcup \Delta_{k-1}\tilde{Y} \sqcup Z \longrightarrow \Delta_{k-2}\tilde{X} \sqcup \tilde{Y} \sqcup (k-1)kZ.$$

In this new version, the tower Δ_{k-2} ends up on $\tilde{X}$ instead of remaining on $\tilde{Y}$. This does not fundamentally change what precedes, and it is easy to check that all the results obtained under the initial Definition 3.1 of Fibonacci moves remain unchanged under the present variant. (In particular, because the graph $\mathcal{F}_2$ remains the same for this variant, the new graph is still nonplanar.)

Now, with this variant of Fibonacci moves, we can make use of the classic graph of Figure 1 to represent the Hanoi-Fibonacci puzzle. In this graph, we preserve the edges that represent 1-Fibonacci moves. The other edges are also preserved, but become arcs of a form that we will call *pseudo-arcs*. More precisely, the graph $\mathcal{G}_n$ of the classic puzzle is made of three copies of $\mathcal{G}_{n-1}$, together with three edges that make $\mathcal{G}_n$ connected. In $\mathcal{F}_n$, these edges are n*-pseudo-arcs*, represented as arrows labelled by $2^{n-2}+1$ (for $n \geqslant 2$). Let v be a vertex of $\mathcal{F}_n$, which is the origin of a k-pseudo-arc e (hence with $1 < k \leqslant n$). The k-Fibonacci move for the vertex v ends up on the vertex v' obtained by a jump of length $2^{k-2}+1$ in the direction of the arc, i.e., v' is the vertex of the graph at a distance $2^{k-2}+1$ from v (each arc or pseudo-arc counting for 1) attained by following the path of length $2^{k-2}+1$ defined by the geometrical direction defined by e (Figure 4).

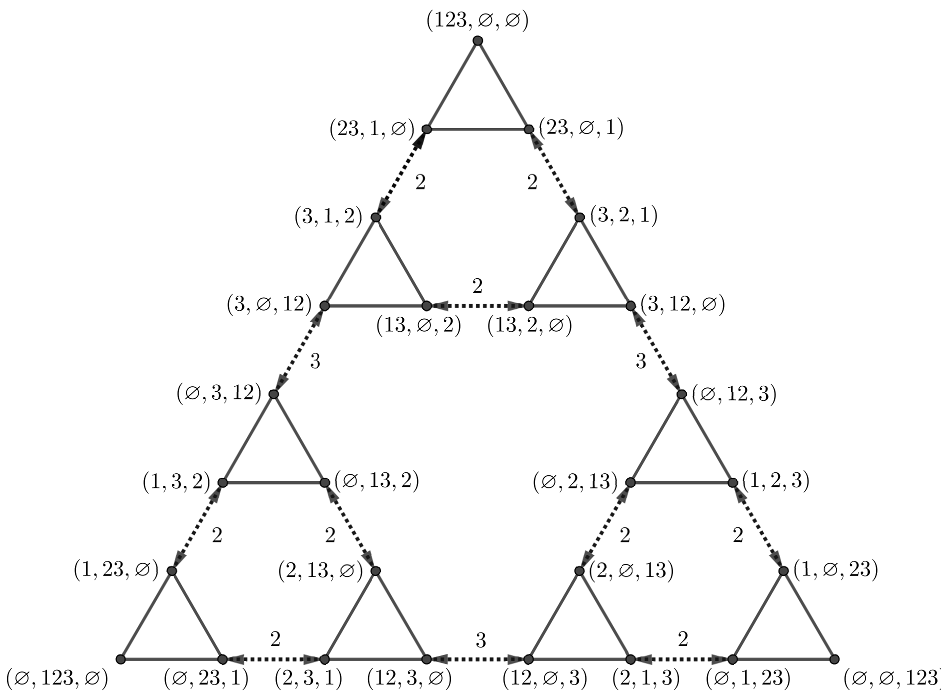

FIGURE 4. The graph $\mathcal{F}_3$ of the Tower of Hanoi-Fibonacci with 3 disks (under the variant of the Fibonacci moves) with its pseudo-arcs.

Theorem 3.13. *Under the previous definition of $\mathcal{F}_n$, if the vertex v is the origin of a k-pseudo-arc ($k \geqslant 2$), then the vertex v' is the state of the puzzle attained by the (only) possible k-Fibonacci move from the state v.*

Proof. Assume the result until $n-1$. The graph $\mathcal{F}_n$ contains three copies of $\mathcal{F}_{n-1}$, in each of which the property is true by induction. Therefore, it remains only to prove that the property is true also for the n-pseudo-arcs of $\mathcal{F}_n$. By symmetry, it is enough to consider the case of the n-pseudo-arc of origin $(n, \Delta_{n-1}, \varnothing)$. The n-Fibonacci move from this state leads to $(\Delta_{n-2}, \varnothing, (n-1)n)$. Also, under the classic rules of the Tower of Hanoi puzzle, going from $(n, \Delta_{n-1}, \varnothing)$ to $(\Delta_{n-2}, \varnothing, (n-1)n)$ with the optimal algorithm requires exactly $2^{n-2}+1$ moves, which are all on the same geometrical direction on $\mathcal{H}_n$, so we are done. $\square$

We deduce from this a combinatorial proof of the following equality (see Figure 5), which has some similarities with the classic sum of the nth row of Pascal's triangle being 2^n and the sum of its nth diagonal being F_n.

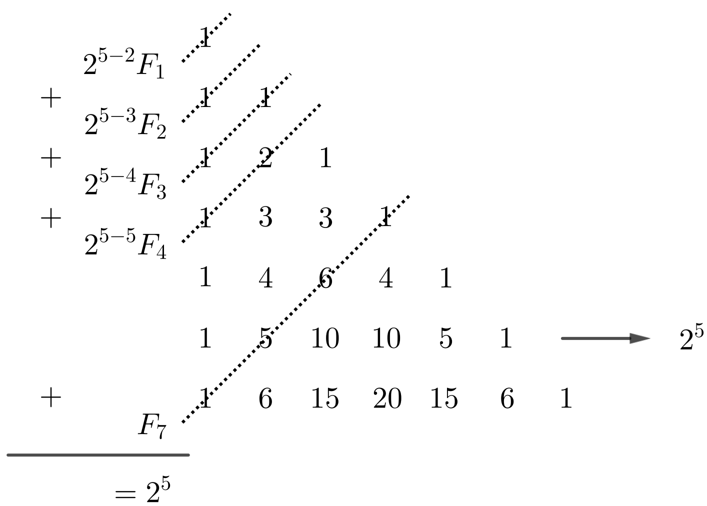

FIGURE 5. Vizualization of Corollary 3.14 for $n = 5$.

Corollary 3.14. *For any $n \geqslant 0$, we have*

$$2^n = F_{n+2} + \sum_{k=0}^{n-2} 2^k F_{n-1-k}.$$

Proof. With the notation of Section 2, we have $2^n - 1 = m_n$. Also, by Theorem 3.4 and the proof of Theorem 3.13, we have

$$\begin{aligned} m_n &= F_n + \sum_{k=3}^{n+1} (2^{k-3}+1) F_{n+2-k} \\ &= \sum_{k=2}^{n+1} F_{n+2-k} + \sum_{k=3}^{n+1} 2^{k-3} F_{n+2-k} \\ &= F_{n+2} - 1 + \sum_{k=0}^{n-2} 2^k F_{n-1-k}. \end{aligned}$$

$\square$

4. Some Generalizations and Questions

4.1. **Modifying the Fibonacci Moves.** Here, we briefly consider alternative ways of defining the allowed moves, extending in a natural way the Fibonacci moves. We write $\Delta_n^{n'}$ for the set of disks d_k with $n' \leqslant k \leqslant n$ (so $\Delta_n^{n'} = \Delta_n^1 = \Delta_n$ for $n' \leqslant 1$ and $\Delta_n^{n'} = \varnothing$ for $n' > n$).

Definition 4.1. *Let $p \geqslant 1$ and $q \geqslant 0$ be two integers. Let X and Y be two different pegs of some state such that, for some $k \in \Delta_n$, we have $X = \Delta_k^{k-p+1} X'$ and $Y = \Delta_{k-p} Y'$. Write Z for the third peg of the state. We define a* (p, q)-move *as a move that consists in putting simultaneously all the disks of $\Delta_k^{k-p+1-q}$ onto Z, i.e.,*

$$\Delta_k^{k-p+1} X' \sqcup \Delta_{k-p} Y' \sqcup Z \longrightarrow X' \sqcup \Delta_{k-p-q} Y' \sqcup \Delta_k^{k-p+1-q} Z.$$

We will talk about the (p, q)-Tower of Hanoi *for the Tower of Hanoi puzzle in which only (p, q)-moves are allowed.*

Note that the $(1, 0)$ case is the classic puzzle, and that the $(1, 1)$ one is the Tower of Hanoi-Fibonacci puzzle.

Theorem 4.2. *The (p, q)-Tower of Hanoi puzzle with n disks admits a solution for any $n \geqslant 0$. There exists only one optimal algorithm for it that needs exactly m_n (p, q)-moves, where the sequence $(m_n)_{n \in \mathbb{Z}}$ is defined by*

$$m_n = \begin{cases} 0, & \text{if } n \leqslant 0; \\ m_{n-p} + m_{n-p-q} + 1, & \text{if } n > 0. \end{cases}$$

Proof. For $n \leqslant p$, the (p, q)-move $(\Delta_n, \varnothing, \varnothing) \longrightarrow (\varnothing, \varnothing, \Delta_n)$ is allowed, so we have $m_n = 1$ for any $n \leqslant p$, which correspond to the formula stated in the theorem.

For $n > p$, the optimal solution is provided by the sequence of critical states, each of which needs, by induction, the number of moves written on its arrow:

$$(\Delta_n, \varnothing, \varnothing) \overset{m_{n-p}}{\longrightarrow} (\Delta_n^{n-p+1}, \Delta_{n-p}, \varnothing) \overset{1}{\longrightarrow} (\varnothing, \Delta_{n-p-q}, \Delta_n^{n-p+1-q}) \overset{m_{n-p-q}}{\longrightarrow} (\varnothing, \varnothing, \Delta_n).$$

To prove that this description is the (only) optimal solution, it is sufficient to prove that, in an optimal solution, the disk d_n, and therefore all the disks from d_{n-p+1} to d_n, move exactly once, which is done by a similar argument already given for the classic puzzle (Section 2) and the Hanoi-Fibonacci one (Theorem 3.2). By summing the number of moves, we get the expected formula. □

There is no serious doubt that generalizations of Zeckendorf-Fibonacci, Gray-like codes, and pseudo-arcs of the graph $\mathcal{H}_n$ can be given for (p, q)-moves, but some additional technicalities may be hard to overcome. For example, the case $p = q = 2$ provides a sequence m_n that is not strictly increasing (because $m_{2n-1} = m_{2n}$), hence a convenient numeration system derived from it is probably not as simple as the Zeckendorf one for the Fibonacci sequence in the case $p = q = 1$. The study of the corresponding graph for (p, q)-moves may be a little bit tricky as well.

4.2. **Restricting the Moves Between Pegs.** Possible variants on the classic puzzle consist in allowing moves only between some pegs. For example, in the *clockwise-cyclic* variant introduced in [1], additionally to the classic rules of Section 2, a disk can move only from A to B, from B to C, or from C to A.

Any variant of this type can be defined by an oriented graph with set of vertices $\{A, B, C\}$, an arc XY standing for the moves from the peg X to the peg Y that are allowed. The sensible variants of this kind (i.e., for which the puzzle is solvable for any n) are the ones for which the corresponding graph is strongly connected [5, Proposition 8.4]. We will not consider all possible cases here, but only mention briefly the *linear* variant, in which the allowed moves are those from A to B, from B to A, from B to C, and from C to B. It is well-known that, for such a restriction, the optimal algorithm for the classic puzzle needs $3^n - 1$ moves, so, because the number of distinct states is 3^n, the linear puzzle also provides the "worst solution", that is, the longest solution that does not come back to any state already met.

Now, consider the linear variant for the Tower of Hanoi-Fibonacci, in which a k-Fibonacci move is allowed if and only if it makes d_k going from A to B, from B to A, from B to C, or from C to B. Write again m_n for the minimal number of moves to solve this variant with n disks. The optimal solution is then given by the following recursive description (for $n \geqslant 3$):

$$(\Delta_n, \varnothing, \varnothing) \xrightarrow{m_{n-1}} (n, \varnothing, \Delta_{n-1}) \xrightarrow{1} (\varnothing, \Delta_n^{n-1}, \Delta_{n-2}) \xrightarrow{1}$$
$$(\Delta_{n-1}^{n-2}, n, \Delta_{n-3}) \xrightarrow{m_{n-3}} (\Delta_{n-1}, n, \varnothing) \xrightarrow{1} (\Delta_{n-2}, \varnothing, \Delta_n^{n-1}) \xrightarrow{m_{n-2}} (\varnothing, \varnothing, \Delta_n).$$

Hence, the sequence $(m_n)_n$ is given by $m_0 = 0$, $m_1 = 2$, $m_2 = 5$, and, for any $n \geqslant 3$, $m_n = m_{n-1} + m_{n-2} + m_{n-3} + 3$ (a kind of a *Tribonacci* sequence). This time, the proof that such a description provides the (only) optimal algorithm is a consequence of that, in an optimal algorithm, the disks d_n moves exactly twice.

Regarding the other variants derived from the restriction of moves between pegs, there is probably no specific difficulty to address them in the context of Fibonacci moves (or (p, q)-moves), apart from some of these variants already involve linear recurring sequences of order 6 in the classic Tower of Hanoi, so are possibly tiresome to describe in our even more technical context.

More interesting would be to find a general way to derive the sequence of moves (or at least the number of moves) from the conjunction of the two kinds of rules. For example, is it possible to deduce the previous Tribonacci sequence directly from what we separately know from the linear variant of the classic puzzle and from our study of the Tower of Hanoi-Fibonacci, instead of the recursive description we presented?

4.3. **Further Questions.** We could also consider even more general rules for the moves. For example, we could allow moves of the form $kX' \sqcup \Delta_{k-1} Y' \sqcup Z \longrightarrow X' \sqcup \Delta_{k-3}(k-1) Y' \sqcup (k-2)kZ$, and so on. One may wonder if two different rules can lead to the same sequence, hence asking for the links between these rules.

Eventually, a deeper work would be to obtain a theoretical way to find from a linear recurring sequence, some natural rules for the Tower of Hanoi for which the number of moves of the optimal algorithm would be given by the sequence. This will probably involve a more precise definition of a "natural rule". (For example, we may ask whether we can always restrict the study to Markovian moves, i.e., moves for which their legality depends only on the initial and final states.) In a sense, answering this question would truly complete Lucas's original assertion.

References

[1] M. D. Atkinson, *The cyclic Tower of Hanoi*, Inf. Process. Lett., **31** (1981), 118–119.

[2] J. Bennish, *The Tower of Hanoi problem and mathematical thinking*, Missouri J. Math. Sci., **11** (1999), 164–166.

[3] A. Bernini, S. Bilotta, R. Pinzani, and V. Vajnovszki, *Two Gray codes for q-ary k-generalized Fibonacci strings*, Proceedings of ICTCS 2013, 54–59.

[4] A. M. Hinz and P. K. Stockmeyer, *Discovering Fibonacci numbers, Fibonacci words, and a Fibonacci fractal in the Tower of Hanoi*, The Fibonacci Quarterly, **57.5** (2019), 72–83.

[5] A. M. Hinz, S. Klavžar, and C. Petr, *The Tower of Hanoi — Myths and Maths*, second ed., Birkhäuser, Cham (Switzerland), 2018.

[6] A. M. Hinz, *Pascal's Triangle and the Tower of Hanoi*, Am. Math. Mon., **99.6** (1992), 538–544.

[7] S. Klavžar and U. Milutinović, *Graphs $S(n,k)$ and a variant of the Tower of Hanoi problem*, Czechoslov. Math. J., **47.122** (1997), 95–104.

[8] N. Claus (= É. Lucas), *La Tour d'Hanoï, Jeu de calcul*, Science et Nature, **1** (1884), 127–128.

[9] N. Claus (= É. Lucas), *La Tour d'Hanoï, Véritable casse-tête annamite*, P. Bousrez, Tours, 1883.
[10] É. Lucas, *Récréations mathématiques, volume III*, Gauthier-Villars et Fils, Paris, 1893.
[11] B. Rittaud, *Reflected Gray codes with Fibonacci constraints, and generalizations* (submitted).
[12] B. Rittaud, *A numeration system and a Gray code given by a variant of the Tower of Hanoi*, Numeration 2017, Rome, 5–10, `http://logica.uniroma3.it/jn17/numeration2017.pdf`.
[13] É. Zeckendorf, *Représentation des nombres naturels par une somme de nombres de Fibonacci ou de nombres de Lucas*, Bull. Soc. R. Sci. Liège, **41** (1972), 179–182.

MSC2020: 00A08, 05C20, 11B39, 68R15, 94B25.

Laboratoire Analyse, Géométrie et Applications (CNRS UMR 7539), Institut Galilée, Université Sorbonne Paris Nord, 93430 Villetaneuse, France
Email address: `rittaud@math.univ-paris13.fr`