On Minimal Words With Given Subword Complexity

Ming-wei Wang*
Department of Computer Science
University of Waterloo
Waterloo, Ontario N2L 3G1
CANADA
`m2wang@neumann.uwaterloo.ca`

Jeffrey Shallit*
Department of Computer Science
University of Waterloo
Waterloo, Ontario N2L 3G1
CANADA
`shallit@graceland.uwaterloo.ca`

Submitted: May 28, 1998; Accepted: July 15, 1998.

Abstract

We prove that the minimal length of a word S_n having the property that it contains exactly F_{m+2} distinct subwords of length m for $1 \le m \le n$ is $F_n + F_{n+2}$. Here F_n is the nth Fibonacci number defined by $F_1 = F_2 = 1$ and $F_n = F_{n-1} + F_{n-2}$ for $n > 2$. We also give an algorithm that generates a minimal word S_n for each $n \ge 1$.

1991 Mathematics Subject Classification: Primary 68R15; Secondary 05C35.

0 Introduction

In this paper we solve a particularly interesting case of the following more general problem. Let $f : \mathbb{N} \longrightarrow \mathbb{N}$ be a non-decreasing function. Given a word w, a *subword* of w is any contiguous block of symbols of w. For each word w over some fixed finite alphabet, we define $P_w(n)$ to be the number of distinct subwords of w of length n. We say that f is *feasible* if for each integer $N \ge 1$ there exists at least one word $w = w(N)$ such that $P_w(n) = f(n)$ for $1 \le n \le N$. Such words $w(N)$ are said to possess the property $\mathcal{P}_{f(N)}$. At the present, there is no known simple characterization of the class of feasible functions. If f is feasible, let us call a shortest word w possessing property $\mathcal{P}_{f(N)}$ a *minimal word of order* N with respect to f. Then several natural questions can be asked.

*Supported in part by a grant from NSERC Canada.

1. What is the length of a minimal word of order N?
2. Is there a reasonably efficient algorithm that finds such minimal words?
3. For each order how many minimal words are there?

We show that the function $f(n) = F_{n+2}$ is feasible, give an algorithm that finds a minimal word of order n for each n and show that the length of a minimal word of order n is $F_n + F_{n+2}$ for $n > 1$. However, the question of a complete enumeration of all minimal words of order n is still open. Here the F_i are the Fibonacci numbers defined by $F_1 = F_2 = 1$ and $F_n = F_{n-1} + F_{n-2}$ for $n > 2$. Previously Good [G46] showed that the length of a shortest word containing as subwords all 2^n binary words of length n is $2^n + n - 1$. In the same year de Bruijn [B46] gave a complete enumeration of all such words (see also [B75]).

The converse problem is usually formulated as finding the function f when given a set of words w. When the words w are the prefixes of some infinite sequence S, the function f is one measure of the complexity of S, and is usually referred to as the subword complexity of S. For related results on subword complexity see the survey article of Allouche [A94].

The proof of our results centers on a detailed analysis of a version of the *de Bruijn graph* which appeared first implicitly in [F94] and explicitly in [R83]. Good [G46] and de Bruijn [B46] independently defined a version of these graphs in 1946. See Fredricksen [F82] for more references for the de Bruijn graph. Observe that $f(1) = F_3 = 2$, which means the number of distinct subwords of length 1 is 2. Thus we need only consider binary words over $\{0, 1\}$ in the rest of this paper.

We divide the presentation of the proof into 4 parts:

1. Existence
2. Structure of the word graph
3. Lower bound on the length
4. Algorithm that generates words which achieve the lower bound

1 Existence

In this section we establish the existence of words with property $\mathcal{P}_{f(n)}$ for each n. The method we employ leads to the de Bruijn graphs. We will define these graphs in this section and use them to prove our result in subsequent sections.

Lemma 1.1 *Let S_n denote the set of words of length n that omit 11. Then $|S_n| = F_{n+2}$ for all integers $n \geq 1$.*

Proof: We proceed by induction. The case $n = 1$ and $n = 2$ are trivial. For the inductive step note that S_n can be partitioned into two sets $S_{n,0}$ and $S_{n,1}$ where $S_{n,0}$ contains words that begin with 0 and $S_{n,1}$ contains words that begin with 1. Since no word of S_n contains 11, it is easy to see that $|S_{n,1}| = |S_{n-2}|$ and $|S_{n,0}| = |S_{n-1}|$. Thus we have $|S_n| = |S_{n-1}| + |S_{n-2}|$.

The Fibonacci numbers satisfy the same recurrence relation. Since we verified the initial condition $S_1 = F_3$ and $S_2 = F_4$, the lemma is proved. ■

Remark: Let w be a word of S_i. Then $w0^{j-i}$ is a member of S_j if $j \geq i$. Hence every word of S_i occurs as a subword of some word of S_j if $i \leq j$.

Theorem 1.1 *There exist finite words with property $\mathcal{P}_{f(n)}$ for each $n > 0$.*

Proof: Let $S_n = \{w_1, w_2, ..., w_m\}$. Then the word $w_1 0 w_2 0 ... w_n$ possesses property $\mathcal{P}_{f(n)}$ by Lemma 1.1 and the remark above. ■

Note that Theorem 1.1 gives an upper bound of $nF_{n+2} + n - 1$ for the length of a minimal word of order n. The next theorem shows that the above construction is essentially unique.

Theorem 1.2 *For all $n > 2$, any finite word w possessing property $\mathcal{P}_{f(n)}$ omits either 00 or 11.*

Proof: Since $n > 2$, $P_w(2) = 3$, and so w omits either 00, 01, 10, or 11. If it omits 01 then $w \in 1^*0^*$ and hence all subwords of w of length 3 are contained in $\{111, 110, 100, 000\}$. This implies $P_w(3) \leq 4$. However $P_w(3) = F_5 = 5$, a contradiction. A similar argument shows w cannot omit 10. Therefore w omits either 00 or 11. ■

1.1 Word graph

We define the particular kind of de Bruijn graphs that we use below. An example is shown in Figure 1.

Definition 1.1 *For $n > 0$, the word graph G_n is a directed graph with labeled edges defined as follows.*

- *The vertices of G_n are all words of length n that omit 11.*
- *The edges of G_n consist of all pairs of vertices (aw, wb) with label b such that $aw \neq wb$ and $a, b \in \{0, 1\}$.*

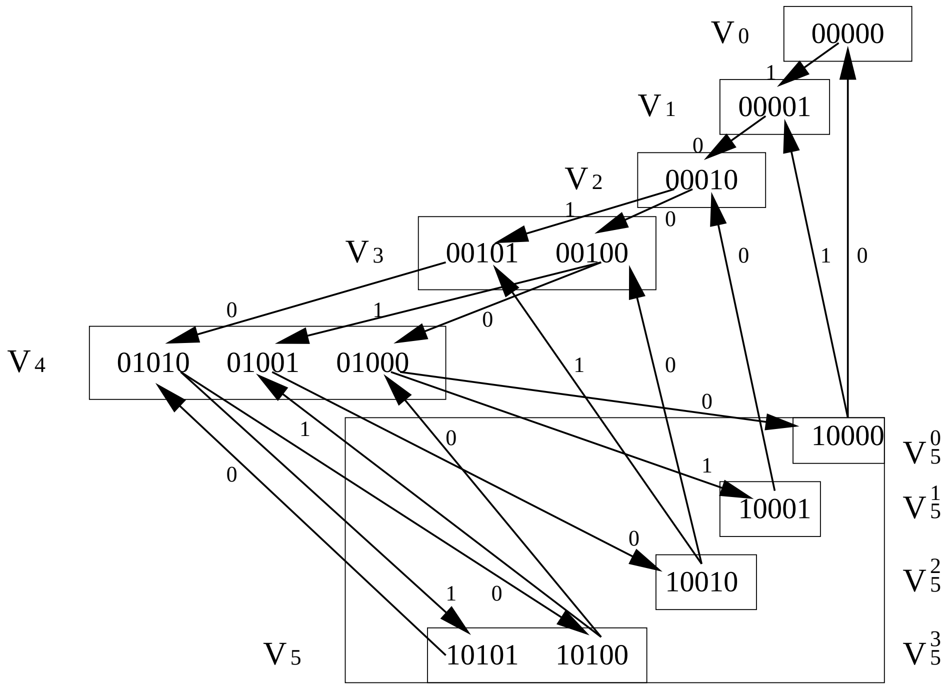

Figure 1: The directed graph G_5.

Let $L(n)$ be the minimal length of a word w possessing property $\mathcal{P}_{f(n)}$. A *walk* in a graph G is a sequence of vertices $\{P_1, P_2, ..., P_m\}$ of G such that (P_i, P_{i+1}) is an edge in G for $1 \leq i \leq m-1$. Note that a walk may repeat both vertices and edges. Let $l(n)$ be the length (number of edges traversed) of a shortest walk through G_n which visits every vertex of G_n at least once. Then Theorem 1.1 and Theorem 1.2 together imply that for $n > 2$, $L(n) = l(n) + n$. In subsequent sections we prove that $l(n) = F_n + F_{n+2} - n$.

2 Structure of the word graph

We summarize few properties of G_n in the following lemma. These properties can be seen more easily by contemplating Figure 1. We say that a graph G is *n-partite* if the vertices of G can be partitioned into n sets such that there are no edges between any pair of vertices in the same partition.

Lemma 2.1 *Let $G_n = G = (V, E)$ be a word graph, and $n > 2$. Then G has the following properties.*

1. *V can be partitioned into disjoint subsets $V_0, V_1, \ldots, V_n$ where V_i consists of words that begin with exactly $(n-i)$ 0's. In addition, V_n can be partitioned into $n-1$ disjoint subsets $V_n^0, \ldots, V_n^{n-2}$ where each V_n^i consists of words of V_i with the first character changed to 1.*

2. *We have $|V_0| = 1$, $|V_i| = F_i$ for $1 \leq i \leq n$, $|V_n^0| = 1$ and $|V_n^i| = F_i$ for $1 \leq i \leq n-2$.*

3. G is an $(n+1)$-partite graph with the V_i's as partitions.

4. For $1 \le i \le n-1$, each vertex in V_i has in-degree 2 and out-degree 1 or 2; each vertex in V_n has in-degree 1 and out-degree 1 or 2.

5. Vertices in V_i point only to vertices in V_{i+1} for $0 \le i \le n-1$; vertices in V_n^i point only to vertices in V_{i+1} for $0 \le i \le n-2$ with the exception that V_n^0 also points to V_0.

These properties of G_n are immediate from the definition. We omit the proof here.

3 Lower Bound

In this section we prove that $l(n) \ge F_n + F_{n+2} - n$ for $n > 1$. Due to certain boundary conditions, results in this section are proved for $n > 2$. The case $n = 2$ can be proved by inspecting G_2 in Figure 2.

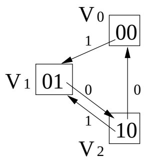

Figure 2: The directed graph G_2.

The following lemma is an easy consequence of parts (1) and (2) of Lemma 2.1.

Lemma 3.1 *$F_{n+2} = 1 + \sum_{i=1}^{n} F_i$ for $n \ge 1$.*

Now let $G = G_n$ be a word graph. By a *complete walk* of G we mean a walk through G that visits each vertex of G at least once. We begin by proving a lower bound on the length of a special type of complete walk of G_n. Then we will sketch the proof that the lower bound thus obtained is a lower bound for all complete walks of G_n.

Lemma 3.2 *For $n > 2$, if a complete walk of $G = G_n$ starts in V_n and ends in $W = V_n^0 \cup V_n^1 \cup V_0 \cup V_1$, then it has length $\ge F_{n+2} + F_n - n$.*

Proof: Define V_i and V_n^i as in Lemma 2.1. Fix an arbitrary complete walk P in G with the appropriate start and end points. Let y_i be the total number of visits by P to vertices of V_i for $0 \le i \le n$. Let x_i be the number of visits to vertices of V_n^i for $0 \le i \le n-2$.

Since P starts in V_n and ends in W, it follows that all visits to V_{i+1} $(2 \le i \le n-2)$ must be preceded by a visit to either V_i or V_n^i, and all visits to V_i and V_n^i are followed by a visit to V_{i+1}. Hence we see that $y_i + x_i = y_{i+1}$ or equivalently $y_i = y_{i+1} - x_i$ for $2 \le i \le n-2$. Furthermore since P starts in V_n, using part 5 of Lemma 2.1 we have $y_n = y_{n-1} + 1$ or equivalently $y_{n-1} = y_n - 1$. Since $y_n = \sum_{i=0}^{n-2} x_i$ by definition, we have $y_{n-1} = y_n - 1 = (\sum_{i=0}^{n-2} x_i) - 1$.

Now for $2 \leq j \leq n-2$, we claim that

$$y_j = (\sum_{i=0}^{j-1} x_i) - 1 \qquad (2 \leq j \leq n-1) \tag{1}$$

The above system of equations can be established by a "downward induction" as follows. First note that we already have $y_{n-1} = (\sum_{i=0}^{n-2} x_i) - 1$, so inductively assume $y_j = (\sum_{i=0}^{j-1} x_i) - 1$ for $3 \leq j \leq n-1$. Now since $y_{j-1} = y_j - x_{j-1}$ we have by the induction hypothesis,

$$\begin{aligned} y_{j-1} &= y_j - x_{j-1} \\ &= (\sum_{i=0}^{j-1} x_i) - 1 - x_{j-1} \\ &= (\sum_{i=0}^{j-2} x_i) - 1 \end{aligned}$$

Thus by induction, (1) is proved.

Now we estimate the value of y_j for each j. Since P is a complete walk, by part 2 of Lemma 2.1 we have $x_0 \geq |V_n^0| = 1$ and $x_i \geq |V_n^i| = F_i$ for $1 \leq i \leq n-2$. Therefore using the system of equations in (1) we obtain the following system of estimates for y_j $(2 \leq j \leq n-1)$.

$$\begin{aligned} y_j &= (\sum_{i=0}^{j-1} x_j) - 1 \\ &\geq 1 + (\sum_{i=1}^{j-1} F_j) - 1 \\ &= F_{j+1} - 1 \qquad \text{(By Lemma 3.1)} \qquad (2 \leq j \leq n-1) \end{aligned} \tag{2}$$

Trivially we also have $y_0 \geq |V_0| = 1$, $y_1 \geq |V_1| = 1$ and $y_n \geq |V_n| = F_n$. Now the length of P can be bounded from below by these estimates as follows.

$$\begin{aligned}
|P| &= (\sum_{i=0}^{n} y_i) - 1 \\
&= y_0 + y_1 + (\sum_{j=2}^{n-1} y_j) + y_n - 1 \\
&\geq 1 + 1 + (\sum_{j=2}^{n-1}(F_{j+1} - 1)) + F_n - 1 \qquad (3) \\
&= 2 + (F_{n+2} - 3) - (n-2) + F_n - 1 \qquad \text{(By Lemma 3.1)} \\
&= F_n + F_{n+2} - n
\end{aligned}$$

Since P is arbitrary, we see that $F_n + F_{n+2} - n$ is a lower bound for this type of complete walk. ■

Now we sketch the proof that $F_n + F_{n+2} - n$ is a lower bound for all complete walks of G_n. Suppose P is a complete walk of G_n that either does not start in V_n or does not end in W. We associate the numbers a and b with the start and end points of P respectively as follows. The number a is the index of the partition where P starts, i.e. P starts in V_a. The number b is slightly more complicated. If P ends in V_i $(0 \leq i \leq n-1)$, then $b = i$. Otherwise P ends in V_n^i $(0 \leq i \leq n-2)$, and we let $b = i$. In other words, we do not worry about where P starts in V_n but we do worry about where P ends in V_n. There are four cases.

1. $a = b + 1$. Then we have $y_i + x_i = y_{i+1}$ for $2 \leq i \leq n-1$. Therefore the system of equations in (1) of Lemma 3.2 is in this case replaced by

$$\begin{aligned}
y_j &= y_{j+1} - x_j \\
&= \sum_{i=0}^{j-1} x_i \qquad (2 \leq j \leq n-1) \qquad (4)
\end{aligned}$$

   By same method as in (2) and (3) of Lemma 3.2, we arrive at a lower bound of $F_n + F_{n+2} - 2$.

2. $2 \leq a \leq b$. Then we have $y_{a-1} + x_{a-1} + 1 = y_a$ and $y_b + x_b = y_{b+1} + 1$ and $y_i + x_i = y_{i+1}$ for $i \neq a-1$ or b, $2 \leq i \leq n-1$. In this case (1) is replaced by

$$\begin{aligned}
y_j &= y_{j+1} - x_j &&= \sum_{i=0}^{j-1} x_i \qquad b < j \leq n-1. \\
y_b &= y_{b+1} - x_b + 1 &&= (\sum_{i=0}^{b-1} x_i) + 1
\end{aligned} \qquad (5)$$

$$
\begin{array}{rclcll}
y_j &=& y_{j+1} - x_j &=& (\sum_{i=0}^{j-1} x_i) + 1 & a \le j \le b-1 \\
y_{a-1} &=& y_a - x_{a-1} - 1 &=& \sum_{i=0}^{a-2} x_i & \\
y_j &=& y_{j+1} - x_j &=& \sum_{i=0}^{j-1} x_i & 2 \le j \le a-2
\end{array}
$$

and (2) is replaced by

$$
\begin{array}{rclll}
y_j &=& \sum_{i=0}^{j-1} x_i & \ge F_{j+1} & b < j \le n-1. \\
y_b &=& (\sum_{i=0}^{b-1} x_i) + 1 & \ge F_{b+1} + 1 & \\
y_j &=& (\sum_{i=0}^{j-1} x_i) + 1 & \ge F_{j+1} + 1 & a \le j \le b-1 \\
y_{a-1} &=& \sum_{i=0}^{a-2} x_i & \ge F_a & \\
y_j &=& \sum_{i=0}^{j-1} x_i & \ge F_{j+1} & 2 \le j \le a-2
\end{array} \tag{6}
$$

Finally in place of (3) we have

$$
\begin{array}{rcl}
|P| &=& (\sum_{i=0}^{n} y_i) - 1 \\
&=& y_0 + y_1 + (\sum_{j=2}^{a-1} y_j) + (\sum_{j=a}^{b} y_j) + (\sum_{j=b+1}^{n-1} y_j) + y_n - 1 \\
&\ge& 1 + 1 + (\sum_{j=2}^{a-1} F_{j+1}) + (\sum_{j=a}^{b} (F_{j+1} + 1)) + (\sum_{j=b+1}^{n-1} F_{j+1}) + F_n - 1 \\
&=& 2 + (\sum_{j=2}^{n-1} F_{j+1}) + (b - a + 1) + F_n - 1 \\
\text{(By Lemma 3.1)} &=& 2 + (F_{n+2} - 3) + (b - a + 1) + F_n - 1 \\
&=& F_n + F_{n+2} + b - a - 1
\end{array} \tag{7}
$$

Thus we obtain a lower bound of $F_n + F_{n+2} + b - a - 1$.

3. $a > b+1$. If $b \ge 2$, then this case is similar to case 2 with the equation $y_b = y_{b+1} - x_b + 1$ switching position with the equation $y_{a-1} = y_a - x_{a-1} - 1$ in (5). The lower bound

derived is again $F_n + F_{n+2} + b - a - 1$. If $b = 0$ or 1, then we have $a \leq n - 1$ and the equations in (5) become

$$\begin{aligned} y_j &= y_{j+1} - x_j &&= \sum_{i=0}^{j-1} x_i && a \leq j \leq n-1. \\ y_{a-1} &= y_a - x_{a-1} - 1 &&= (\sum_{i=0}^{a-2} x_i) - 1 && \\ y_j &= y_{j+1} - x_j &&= (\sum_{i=0}^{j-1} x_i) - 1 && 2 \leq j \leq a-2 \end{aligned} \tag{8}$$

and we can derive a lower bound of $F_n + F_{n+2} - a$.

4. $a = 0$ or $a = 1$. This is similar to case 2 except that the equations in (5) involving y_0 and y_1 are invalid. We remove the invalid equations from (5). Then if $b \geq 2$, we can work through (2) and (3) of Lemma 3.2 as we have done for case 2 and obtain a lower bound of $F_n + F_{n+2} + b - 3$. If $b = 0$ or 1, then (5) reduces to (4) and we get the same lower bound of $F_n + F_{n+2} - 2$.

In all cases, if $n > 2$, we found a larger lower bound. Therefore we may take $F_n + F_{n+2} - n$ as a lower bound for all complete walks of G_n, for $n > 2$. As we mentioned at the beginning of this section, this bound also holds for $n = 2$.

We can say rather more.

Corollary 3.2.1 *For $n > 2$, P is a minimal complete walk of G_n of length $F_n + F_{n+2} - n$ if and only if P starts in V_n, ends in W and visits each vertex of $V_n \cup W$ exactly once. Furthermore one of the following two conditions holds:*

1. *P starts in V_n^0 and ends in V_n^1.*

2. *P starts in V_n^1 and ends in V_1.*

Proof: Observe that the lower bounds we obtained for complete walks that either do not start in V_n or do not end in W are $> F_n + F_{n+2} - n$. Therefore from the proof of Lemma 3.2 we see that P is a complete walk of length $F_n + F_{n+2} - n$ if and only if P starts in V_n, ends in W and visits each vertex of V_n exactly once. So what remain to be shown are the two conditions on the start and end points of P.

Where could P end? P could not end in V_n^0 because otherwise vertices in $V_0 \cup V_1$ are not visited by P. P could not end in V_0 because then the only way to reach V_1 is from V_n^0. But V_0 is only reachable from V_n^0. Hence the single vertex in V_n^0 is visited more than once, contradicting our assumption about P.

Next we show that P must start in $V = V_n^0 \cup V_n^1$. To see this let us define $w_1, ..., w_{n-2}$ inductively as follows: $w_1 =$ parent of the single vertex in V_n^0, $w_j =$ parent of w_{j-1} that is not in V_n for $2 \leq j \leq n-2$. We claim that if P starts in $V_n \setminus V$, then all of w_j $(1 \leq j \leq n-2)$ are visited more than once. We prove this by induction. First, since w_1 has as its children the two vertices of V and they are only reachable from w by part 5 of Lemma 2.1, w_1

must be visited more than once. Now inductively assume w_j is visited more than once for $1 \leq j < n-2$. Note that w_j is one of the two children of w_{j+1}. As w_j is visited more than once by the induction hypothesis, the total number of visits to the two children of w_{j+1} is greater than 2. But the other parent w of w_j is in V_n and thus is visited only once. So w_{j+1} must be visited more than once. Thus by induction, our claim is proved. Now suppose P ends in V_1. Observe that w_{n-2} is the single vertex in V_2. Thus w_{n-2} is reachable only from V_1 and V_n^1. Therefore since P ends in V_1, w_{n-2} is visited more than once implies that the single vertex of V_n^1 is visited more than once, a contradiction. Similarly if P ends in V_n^1 we see that the single vertex of V_1 is visited more than once which again contradicts our assumption about P.

Lastly, we prove the connection between the start and the end points of P. Suppose P starts in V_n^0. Then since V_0 and V_1 consist of the two children of V_n^0, we see that P must end in V_n^1, because ending in V_1 would imply either V_n^0 is visited more than once or the only vertices visited by P are those of $V_n^0 \cup V_0 \cup V_1$. In either case we arrive at a condition incompatible with our assumptions about P. Similarly, if P starts in V_n^1 then P must end in V_1. ∎

4 Algorithm

We now give an algorithm that traces a complete walk in G that satisfies the conditions of Corollary 3.2.1. It will then follow that our lower bound is achievable. Consequently the shortest word satisfying $\mathcal{P}_{f(n)}$ is of length $F_n + F_{n+2}$ for $n > 1$. As in Section 3, we will assume $n > 2$ throughout this section unless otherwise specified. The case $n = 2$ is seen to be true by inspection.

We now introduce an order on the vertices of G to facilitate descriptions of the algorithm and its proof. We think of G as drawn in $n+1$ levels with V_0 at the top and V_n at the bottom. Within each level, the vertices are ordered by their value as integers in binary. Large vertices are placed to the left. We view G as a tree with root V_0 and "leaves" V_n except that there are back edges from the "leaves" to the interior vertices. See Figure 1 for an example.

Now we can describe the algorithm as

Traverse ($\mathcal{T}$)
Input: G_n.
Output: A complete walk of G_n.
Begin

1. Start at 10...0 (the single vertex of V_n^0).
2. Go to 0...0 (the single vertex of V_0).
3. If current vertex has only one child (or equivalently current vertex ends in 1) then
   visit it.
   else if the left child of the current vertex has not been visited then
   visit the left child (left child is the one that ends in 1).
   else
   visit the right child (right child is the one that ends in 0).
4. If the current vertex is 10...01 (the single vertex of V_n^1) then
   Stop.
   else
   Repeat step 3.

End

An example walk traced out by the algorithm on G_5 in Figure 1 is as follows: $10000 \xrightarrow{0} 00000 \xrightarrow{1} 00001 \xrightarrow{0} 00010 \xrightarrow{1} 00101 \xrightarrow{0} 01010 \xrightarrow{1} 10101 \xrightarrow{0} 01010 \xrightarrow{0} 10100 \xrightarrow{1} 01001 \xrightarrow{0} 10010 \xrightarrow{0} 00100 \xrightarrow{0} 01000 \xrightarrow{1} 10001$ which gives the word 100000101010010001.

We will henceforth refer to the algorithm just stated as algorithm $\mathcal{T}$. The following lemma gives the basic property of $\mathcal{T}$.

Lemma 4.1 *The number of times $\mathcal{T}$ visits a vertex v of a word graph $G = G_n$ is at most the number of children of v.*

Proof: Suppose to the contrary, there exists v that is visited more often than the number of its children and let us call such a vertex *selfish*. As $\mathcal{T}$ runs sequentially we can pick the first selfish vertex v. Could $v \in V_0 \cup V_1$? If $v \in V_0$, then since the only edge to v is from the start vertex and the start vertex cannot be visited more than once because it is the sibling of the vertex at which $\mathcal{T}$ terminates, we have that v is unselfish, a contradiction. Similarly it is easy to see that $v \notin V_1$.

Now suppose $v \in V_k$, $1 < k \leq n$. There are two cases.

Case 1: v ends in 1. In this case v's parent(s) has (have) two children and v is the left child of its parent(s). By step 3 of $\mathcal{T}$, v is never visited more than once. As v has at least one child, v cannot be selfish.

Case 2: v ends in 0. Since v ends in 0 and $k > 1$, v has two children. If $k = n$, then by Lemma 2.1 part 4, v has one parent. So if v is visited more than two times, so is v's parent. But the parent becomes selfish first. This contradicts that v is the first selfish vertex. If $k < n$, then v has two parents $v_1 \in V_n$ and $v_2 \notin V_n$. There are two cases here. By Lemma 2.1, both parents share the same children.

Case 2a: If v is the only child of both parents, then since v is visited more than two times, one of the parents has been visited more than once and it becomes selfish before v. This is a contradiction.

Case 2b: If v's parents have two children, then by step 3 of $\mathcal{T}$ v must be the right child. Recalling that v has 2 children, we have the total number of times v and its left sibling were visited is > 3. Since v is the first selfish vertex, the parent $v_2 \notin V_n$ can be visited at most two times, then $v_1 \in V_n$ must have been visited at least two times. However v_1 has only one parent. So v_1's parent, say w, is visited at least twice. So w must have two children. But then v_1 must be the right child of w and w must have been visited at least three times. Namely, w becomes selfish before v does. This is a contradiction.

Since k is arbitrary, we conclude that such v does not exist. So the lemma is proved. ■

The previous lemma implies the following property of $\mathcal{T}$.

Corollary 4.1 *$\mathcal{T}$ visits each vertex of V_n at most once.*

Proof: Let us call a vertex $v \in V_n$ *popular* if v is visited by $\mathcal{T}$ more than once. We will show that all vertices of V_n are unpopular. Consider an arbitrary vertex $v \in V_n$. Let w be a parent of v. Note that v has only one parent by part 4 of Lemma 2.1. So w is unique. If v is the only child of w, then the unselfishness of w implies the unpopularity of v. If w has two children then there are two cases.

Case 1: v ends in 1. Then v has only one child. So v is not popular since by Lemma 4.1 v is not selfish.

Case 2: v ends in 0. Then v must be the right child of w. Since w is unselfish, step 3 of $\mathcal{T}$ implies that v is not popular. Thus the lemma is proved. ■

By Lemma 4.1, the two vertices in $V_0 \cup V_1$ are unselfish. Since each of the two vertices has only one child, their unselfishness implies that they are visited by $\mathcal{T}$ at most once. Thus $\mathcal{T}$ visits each vertex in $W \cup V_n$ at most once. If we now prove that $\mathcal{T}$ indeed traces out a complete walk, then by Corollary 3.2.1, we conclude that it is a complete walk of length $F_n + F_{n+2} - n$ and the problem is solved. To do so we investigate how visitation of one vertex affects another vertex. It turns out that certain pairs of vertices are closely related in their visitation status. Such pair of vertices appears frequently in the sequel. Thus we define them as follows.

Definition 4.1 *Suppose v is not the start vertex (10...0), then we say that w is the* rightmost descendant *of v if w satisfies the following three conditions.*

1. *$w \neq v$.*
2. *$w \in V_n$ and all edges on the shortest path from v to w have label 0.*
3. *w is the only vertex in V_n not equal to v on the shortest path from v to w.*

Two examples are shown in Figure 3. As usual we let the *distance* from a vertex v to a vertex w to be the length (number of edges) of the shortest path from v to w. Observe that if v' is the rightmost descendant of v and the distance from v to v' is greater than 1, then v' is also the rightmost descendant of the right child of v.

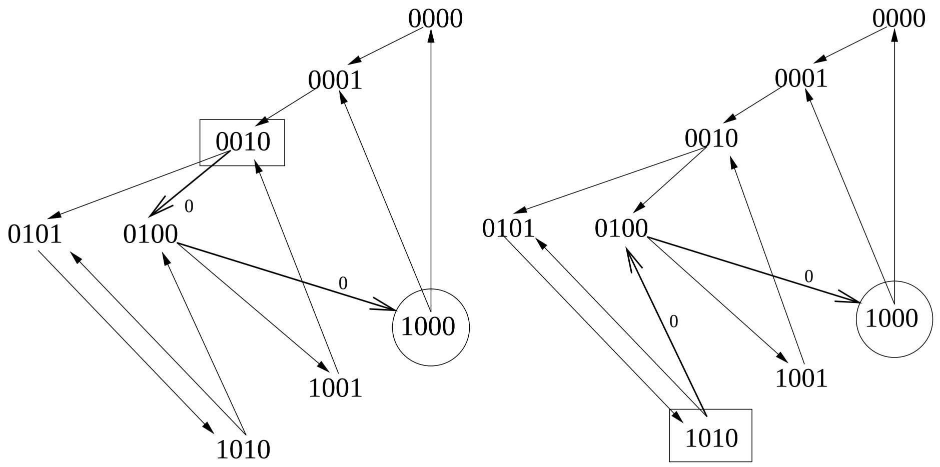

Figure 3: Circled vertices are the rightmost descendants of boxed vertices.

The basic relation between v and its rightmost descendant is contained in the following lemma.

Lemma 4.2 *Suppose vertex v ends in 0, and its rightmost descendant v' is not the start vertex. Then*

1. *If $v \in V_n$ is not visited by $\mathcal{T}$, then v' is not visited by $\mathcal{T}$.*
2. *If $v \notin V_n$ is visited by $\mathcal{T}$ only once, then v' is not visited by $\mathcal{T}$.*

Proof: By step 1, step 2 and the first application of step 3 of $\mathcal{T}$, it is clear that all vertices in $V_0 \cup V_1$ are visited by $\mathcal{T}$. Therefore we assume $v \notin V_0 \cup V_1$ in the rest of the proof. We proceed by induction on the distance d from v to v'.

$d = 1$: In this case we have $v \notin V_n$ and v has two children. Since v' is not the start vertex and is the right child of v, then by step 3 of $\mathcal{T}$, v' is not visited because v is visited only once.

$d = k > 1$: Since v ends in 0, $v \notin V_0 \cup V_1$, v has two children. Since $k > 1$, the right child w of v is not in V_n. Let u be the other parent of w. If v is in V_n and is not visited by $\mathcal{T}$, then Lemma 4.1 implies that w is visited only once because w is u's right child and u is unselfish. Suppose $v \notin V_n$ and is visited by $\mathcal{T}$ only once. Then the other parent u of w is in V_n and Corollary 4.1 implies that w is visited only once due to the unpopularity of u. Since the distance from w to v' is less than d, by the induction hypothesis v' is not visited. This proves the lemma. ∎

Next we need

Lemma 4.3 *Suppose the rightmost descendant v' of v is the start vertex. Then v is visited by $\mathcal{T}$ at most once.*

Proof: If $v \in V_n$, then Corollary 4.1 implies this lemma. So suppose $v \notin V_n$. We prove this case by induction on the distance from v to v'. It is easy to see that the result holds for vertices of $V_0 \cup V_1 \cup V_2$ since the only vertices in V_n that contain an edge to them are the start and end vertex. Therefore we inductively assume the lemma is true for distance $d > n-3$. For distance $d \leq n-3$, v must be the right child of its parents because the start vertex is the rightmost vertex of V_n and $d \leq n-3$. Since v has at most 2 parents, if v is visited more than once, one of its parents must be visited more than once because v is the right child of both parents. But this contradicts the induction hypothesis. So v is visited at most once. This proves the lemma. ■

Lemma 4.3 can be sharpened to

Lemma 4.4 *Suppose the rightmost descendant v' of v is the start vertex, then v is visited by $\mathcal{T}$ exactly once.*

Proof: The cases where $v \in V_0 \cup V_1$ are trivially true. We prove the other cases by induction on the distance d from v to v'.

$d = 1$: In this case v is the parent of the end vertex. Since there are only finitely many vertices in G_n, by Lemma 4.1, $\mathcal{T}$ terminates. Therefore v is visited exactly once.

$d > 1$: There are two cases here.

Case 1: $v \notin V_n$. Suppose v has not been visited. Then Corollary 4.1 and step 3 of $\mathcal{T}$ implies that the right child of v is not visited. This contradicts the induction hypothesis. So by Lemma 4.3, v is visited exactly once.

Case 2: $v \in V_n$. If v is not visited, then by case 1 above, the right child of v is not visited. This again contradicts the induction hypothesis. So v is visited exactly once. By induction, the lemma is proved. ■

Putting previous results together, we can now prove the main theorem.

Theorem 4.1 *$\mathcal{T}$ traces out a minimal complete walk in G_n of length $F_n + F_{n+2} - n$.*

Proof: We first prove that all vertices of V_n are visited. Assume to the contrary that some vertices of V_n are not visited. Pick the rightmost such vertex v. Note that $v \notin V_n^0 \cup V_n^1$ as these are the start and end points of $\mathcal{T}$. For vertices $v \in V_n \setminus (V_n^0 \cup V_n^1)$ there are two cases.

Case 1: v ends in 1. Then v's parent has two children and v is the left child. By step 3 of $\mathcal{T}$, if v is not visited, then v's right sibling cannot have been visited. This is impossible as v is supposed to be the rightmost unvisited vertex.

Case 2: v ends in 0. Then let v' be the rightmost descendant of v. If v' is not the start vertex, then by Lemma 4.2 v' is not visited. Note that v' is obtained from v by removing a positive number of symbols from the left end of v and appending the same number of zeros to the right end of v. Therefore by repeated applications of Lemma 4.2, we get a sequence of vertices v', v'', ... such that eventually we arrive at an unvisited vertex $v^{(m)}$ whose rightmost descendant $v^{(m+1)}$ is the start vertex. However Lemma 4.4 now implies that $v^{(m)}$ is visited by $\mathcal{T}$, a contradiction. This proves that $\mathcal{T}$ visits all vertices of V_n.

Now we assume inductively that $\mathcal{T}$ visits all vertices of V_k, $2 < k \leq n$. Suppose some vertices of V_{k-1} are not visited. Pick the rightmost such vertex. There are two cases.

Case 1: v ends in 1. The same argument as in case 1 above shows this case is impossible.

Case 2: v ends in 0. Since $v \notin V_0 \cup V_1$, v has two children. Then Corollary 4.1 implies that the right child of v is not visited. This contradicts the inductive hypothesis. So this case is also impossible.

Thus $\mathcal{T}$ visits all vertices of V_{k-1}. By induction, $\mathcal{T}$ visits all vertices of V_i, $2 \leq i \leq n$. By step 1 and step 2 of $\mathcal{T}$, it also visits the vertices of V_0 and V_1. Further by Lemma 4.1, each vertex in $V_0 \cup V_1$ is visited exactly once. So it traces a complete walk in G_n which satisfies the condition of Corollary 3.2.1. Therefore we conclude that the path it traced is a minimal complete walk of length $F_n + F_{n+2} - n$. ■

A table of words produced by the algorithm for $1 \leq n \leq 7$ is given below. The cases $n = 1$ and $n = 2$ are produced by hand.

1	10
2	1001
3	1000101
4	10000101001
5	100000101010010001
6	10000001010100101000100100001
7	10000000101010100101000101000010010010001000001

5 Remarks on generalizations

It would be interesting to see if some of the ideas presented here could be used to obtain more general results on feasible functions. In particular lower bounds on the length of minimal words for f satisfying a linear recurrence. It seems that the idea of partitioning the vertices of the word graph into "levels" may be useful in this context.

6 Acknowledgement

Theorem 4.1 proved a conjecture on length which originated from David Swart's computation of the minimal words of order n for $1 \leq n \leq 6$.

References

[A94] Allouche, J.-P. Sur la complexité des suites infinies, *Bull. Belg. Math. Soc.* **1** (1994), 133-143.

[B46] de Bruijn, N. G. A combinatorial problem, *Nederl. Akad. Wetensch. Proc.* **49** (1946), 758–764.

[B75] de Bruijn, N. G. Acknowledgement of priority to C. Flye Sainte-Marie on the counting of circular arrangements of $2n$ zeros and ones that show each n-letter word exactly once, *TH-Report 750WSK-06, Eindhoven University of Technology the Netherlands* (1975), 1-14.

[G46] Good, I. J. Normally recurring decimals, *J. London Math. Soc.* **21** (1946), 167–169.

[F82] Fredricksen, H. A survey of full length nonlinear shift register cycle algorithms, *SIAM Rev.* **24** (1982), 195-221.

[F94] Flye-Sainte Marie, C. Solution to problem number 58, *L'Intermédiaire des Mathématiciens* **1** (1894), 107–110.

[R83] Rauzy, G. Suite à termes dans un alphabet fini, *Séminaire de Théorie des Nombres de Bordeaux* (1982-1983), n° 25.