How Many Squares Must a Binary Sequence Contain?

Aviezri S. Fraenkel[1] and *R. Jamie Simpson*[2]

Submitted: November 16, 1994; Accepted: December 11, 1994

Abstract. Let $g(n)$ be the length of a longest binary string containing at most n distinct squares (two identical adjacent substrings). Then $g(0) = 3$ (010 is such a string), $g(1) = 7$ (0001000) and $g(2) = 18$ (010011000111001101). How does the sequence $\{g(n)\}$ behave? We give a complete answer.

1. Introduction

A binary word (or string) containing no square (a pair of identical adjacent subwords) has maximum length 3; in fact, the only squarefree words of length 3 are 010 and its 1-complement 101. A computer disclosed that a binary word containing at most 1 square has maximum length 7: the only words of length 7 with only 1 square are

$$0001000, \qquad 0100010, \qquad 0111011$$

and their 1-complements and the reverse of 0111011 and its 1-complement. Further, a binary word containing at most 2 distinct squares has maximum length 18; the only words of length 18 which contain only 2 distinct squares are

$$010011000111001101$$

and its 1-complement (which is also its reverse).

In general, let $g(k)$ denote the length of a longest binary word containing at most k distinct squares. "Distinct" means that the squares are of different shape, not just translates of each other. We have seen that $g(0) = 3$, $g(1) = 7$, $g(2) = 18$. This data raises the following natural questions.

1 Department of Applied Mathematics & Computer Science, The Weizmann Institute of Science, Rehovot 76100, Israel. Email: fraenkel@wisdom.weizmann.ac.il . Work done while visiting Curtin University.

2 School of Mathematics, Curtin University, Perth WA 6001, Australia. Email: tsimpsonr@cc.curtin.edu.au

1. Is the set of values of the sequence $S = \{g(k) : k = 0, 1, \ldots\}$ infinite or finite?
2. What's the value of $g(3)$?

Regarding the first of these questions, Entringer, Jackson and Schatz [1974] considered the conjecture that S is infinite, citing a reference "which... seems to say that [this] conjecture... is true". They then go on to show that S is finite, by proving that $g(5) = \infty$, i.e., there exists an infinite binary sequence with only 5 squares!

It has been shown many times that there exist infinite squarefree *ternary* sequences. See e.g., Thue [1912], Morse and Hedlund [1944], Hawkins and Mientka [1956], Leech [1957], Novikov and Adjan [1968], Pleasants [1970], Burris and Nelson [1971/72], del Junco [1977], Ehrenfeucht and Rozenberg [1983]. (Currie [1993] wrote: "One reason for this sequence of rediscoveries is that nonrepetitive sequences have been used to construct counterexamples in many areas of mathematics: ergodic theory, formal language theory, universal algebra and group theory, for example...".) Actually, Thue [1912] showed more: there exists a doubly infinite squarefree ternary sequence which also avoids the 2 triples $a_1 a_3 a_1$ and $a_2 a_3 a_2$. See Berstel [1992, §4.2] for an exposition of the full result, and Berstel [$\geq$ 1995] for an English translation of Thue's papers.

Roth [1991] has proved that given any alphabet Σ of more than 2 letters, any given pattern, such as a square, is avoidable over Σ, if and only if there exists an infinite binary word in which any morphism of that pattern is of bounded length.

Seen in this light, the result of Entringer *et al.* [1974] is not surprising. But it brings into even sharper focus the second question, because it makes us wonder about the values of $g(3)$ and $g(4)$.

We give a complete answer by showing that $g(3) = \infty$. In §2, after establishing some notation and definitions, we construct an infinite binary sequence, and in §3 we prove that it contains only the 3 squares 0^2, 1^2 and $(01)^2$.

We also remark that questions regarding squares in sequences arise in molecular biology, where they are known as *repeats*, or *tandem repeats*. In fact, the most frequent repeat in the human genome seems to be the *binary* word GT, with high *copy number* (the number of times GT is repeated). Trifonov [1989] argues that the copy number influences the functions of DNA chains adjacent to the repeated word, such as their binding power and gene expression; it can even cause certain diseases if too high or too low; and it also influences the unwinding capability of the DNA helix. Algorithms for identifying repeats and databases of repeats in the human genome are maintained by Milosavljević [$\geq$ 1995].

Since the copy number at a given site changes from one individual to another, the copy number has also been used in DNA-*fingerprinting*. This application appears to have been originated by Alec Jeffreys' group in Leicester. See e.g., Jeffreys, Wilson and Thein [1985] and Jeffreys, Turner and Debenham [1991]. Further elaborations on applications of DNA-fingerprinting to medicine and forensic medicine are given in Raskó and Downes [1995, ch. 6, especially p. 156; and ch. 12, especially pp. 379–380], where it is also stated that the human genome contains some 500,000 repeated words. (Keywords for human genome applications are VNTR (Variable Number Tandem Repeats) and mini- and microsatellite sequences for the basic subwords that are repeated.)

2. Construction of the Binary Sequence

We begin with some notation.

Denote by Σ^* the set of all *words* (finite or infinite strings, also called blocks) over the finite alphabet Σ, whose elements are *letters*. Given a finite word $\sigma = \sigma_1 \cdots \sigma_n \in \Sigma^*$, $\sigma_i \in \Sigma$ $(i \in \{1, \ldots, n\})$, the *length* of σ is $|\sigma| = n =$ number of letters in σ, counting multiplicities. Below we use the binary, ternary and quinary alphabets, denoted by $B = \{0, 1\}$, $T = \{a_1, a_2, a_3\}$, $Q = \{a_1, a_2, a_3, a_4, a_5\}$, respectively.

A *prefix* of a word is a subword at the *beginning* (left side) of the word; a *suffix* is a subword at the *end* (right side) of the word. Given words $x, y \in \Sigma^*$, we denote by xy the concatenation of these words, beginning with x and ending with y. Thus x^2 is the square xx. If x is a subword of y, we also write $x \subseteq y$.

A function $C: Q^* \to B^*$ is an *encoding* (a binary encoding of Q^*). Given a finite or infinite quinary word $q = q_1 q_2 \cdots \in Q^*$, $q_i \in Q$ $(i \in \{1, 2, \ldots\})$, C is defined by the *code* $C(q) = C(q_1)C(q_2)\cdots$, where the $C(a_i)$ are the given *codewords* $(i \in \{1, \ldots, 5\})$. Thus the codeword $C(a_i)$ is also the code of a_i. *Decoding* refers to the inverse function $C^{-1}: B^* \to Q^*$ if it exists. To *parse* any subword of a code means to identify beginnings and ends of all the codewords contained entirely in the subword.

We are now ready to describe the construction of the doubly infinite binary word which has only 3 squares. Since the construction involves infinite processes, we call it a procedure rather than an algorithm.

Procedure TQB. (1) Let $t \in T^*$ be a doubly infinite squarefree ternary word over $T = \{a_1, a_2, a_3\}$, which avoids $a_1 a_3 a_1$ and $a_2 a_3 a_2$.

(2) Replace every occurrence of $a_2 a_3$ in t by $a_2 a_4 a_3$, and every occurrence of $a_3 a_2$ by $a_3 a_5 a_2$. The result is a doubly infinite quinary word $q \in Q^*$.

TABLE 1. Possible pairs of q.

a_1a_2	a_2a_1	a_3a_1	a_4a_3
a_1a_3	a_2a_4	a_3a_5	a_5a_2

TABLE 2. Possible triples of q.

$a_1a_2a_1$	$a_2a_1a_2$	$a_3a_1a_2$	$a_4a_3a_1$
$a_1a_2a_4$	$a_2a_1a_3$	$a_3a_1a_3$	$a_5a_2a_1$
$a_1a_3a_5$	$a_2a_4a_3$	$a_3a_5a_2$	$a_5a_2a_4$

(3) Define $C(q)$ by

$$\begin{aligned}
C(a_1) &= 011\ 000\ 111\ 001\\
C(a_2) &= 011\ 100\ 011\ 001\\
C(a_3) &= 011\ 001\ 110\ 001\\
C(a_4) &= 011\ 0001\ 0111\ 001\\
C(a_5) &= 011\ 1001\ 0110\ 001.
\end{aligned}$$

From this encoding we see that $C(q)$ contains the squares 0^2, 1^2 and $(01)^2$. In the next section we show that $C(q)$ contains no other squares. The main idea is to establish an explicit bound on the length of the squares of $C(q)$. $\{$The name TQB of the procedure of course reminds us that in step 1 we have a Ternary sequence, in step 2 we create a Quinary sequence, and in step 3 a Binary sequence.$\}$

3. The Binary Sequence Contains Only 3 Squares

A single 0 sandwiched between 2 neighboring 1-bits will be called an *isolated* 0.

We begin by collecting some easily proved properties of the sequences q and $C(q)$ generated in Procedure TQB.

(i) All and only all the pairs and triples of q are listed in Tables 1 and 2 respectively.

(ii) The lengths of the $C(a_i)$ is 12 ($i \in \{1,2,3\}$) and 14 ($i \in \{4,5\}$). Only $C(a_4)$ and $C(a_5)$ contain isolated 0's; the only other isolated 0's are at the beginning of every codeword $C(a_i)$, in every concatenation $C(a_j)C(a_i)$. Hence the only distances between consecutive isolated 0's in $C(q)$ are 7 or 12. The sequence of these distances has the form

$$\cdots\ 7^2\ 12^{r_{-2}}\ 7^2\ 12^{r_{-1}}\ 7^2\ 12^{r_0}\ 7^2\ 12^{r_1}\ 7^2\ 12^{r_2}\ 7^2\ \cdots\ ,$$

where the r_i are *positive* integers (since a_4 and a_5 cannot be adjacent).

(iii) The doubly infinite sequence $C(q)$ can be parsed uniquely into codewords $C(a_i)$ ($i \in \{1, \ldots, 5\}$) by placing a comma in front of isolated 0's at distances 12 and 14 (skipping those isolated 0's which are at distance 7 from both of their preceding and succeeding isolated 0). Thus $C(q)$ can be decoded uniquely into q.

(iv) A codeword $C(a_i)$ is not a prefix or suffix of $C(a_j)$ for any $j \neq i$.

We show now that property (iii) can be strengthened: also certain finite, even short subwords of $C(q)$ can be parsed uniquely.

PROPOSITION 1. Any subword w of $C(q)$ which contains a codeword can be parsed uniquely, and so any codeword in w can be decoded uniquely.

PROOF. Suppose first that w contains no isolated 0. Then (ii) implies that $|w| = 12$ or 13, and the 12 left bits constitute a unique codeword. If w contains 2 isolated 0's at distance 12 then a unique codeword at length 12 can be identified, which induces a unique parsing on w. Unique parsing also results if w contains 3 isolated 0's at distances $7, 7$, when a unique codeword of length 14 can be identified. By (ii), the only remaining cases are 2 isolated 0's, z_1 and z_2, at distance 7, say with z_1 to the left of z_2, or else a single isolated 0, denoted by z.

If there are precisely 12 bits to the left of z_1 (or z), then they constitute a unique codeword. Similarly, if there are 11 or 12 bits to the right of z_2 (or z), then z_2 (or z) and the first 11 bits to its right constitute a unique codeword. So suppose that neither of these two cases holds. Then w must contain $C(a_4)$ or $C(a_5)$. In fact, either there are precisely 7 bits to the left of z_1 beginning in 01, which constitute the beginning of $C(a_4)$ or $C(a_5)$; or else there are precisely 6 or 7 bits to the right of z_2, the first 6 of which end in 01, which constitute the end of $C(a_4)$ or $C(a_5)$. In the case of z, there must be precisely 7 bits to the left of z beginning in 011 *and* precisely 6 or 7 bits to the right of z, the first 6 of which end in 001, which identifies $C(a_4)$ or $C(a_5)$ uniquely. ∎

In Table 3 the braces indicate illegal parsings; in fact, they violate the conditions, given at the end of the proof, which the bits near z_1, z_2 and z have to satisfy. By (i), Table 3 lists all the pairs containing a_4 or a_5.

We now come to the main result.

PROPOSITION 2. Let $C(q)$ be a doubly infinite binary word produced by Procedure TQB. Then every square of $C(q)$ is contained in some subword $C(q') \subseteq C(q)$ where $q' \subseteq q$ with $|q'| \leq 3$.

PROOF. Suppose $b_1 \cdots b_{2m} \subseteq C(q')$ is a (binary) square which intersects the code of $|q'| \geq 4$ letters of q. Denote the words $b_1 \cdots b_m$, $b_{m+1} \cdots b_{2m}$, $b_1 \cdots b_{2m}$ by w_L, w_R, $w = w_L w_R$ respectively. Observe that $|q'| \geq 4$ implies that either w_L or w_R contains a complete codeword, say c_1. Assume c_1 is contained in w_L, say.

TABLE 3. Encodings of the 4 pairs containing a_4 and a_5.

$$C(a_2a_4) = 011\ 10\overbrace{0\ 011\ 001 \mid 011\ 0001}\ 0111\ 001$$

$$C(a_3a_5) = 011\ 00\overbrace{1\ 110\ 001 \mid 011\ 1001}\ 0110\ 001$$

$$C(a_4a_3) = 011\ 0001\ \overbrace{0111\ 001 \mid 011\ 001\ 1}10\ 001$$

$$C(a_5a_2) = 011\ 1001\ \overbrace{0110\ 001 \mid 011\ 100\ 0}11\ 001$$

Suppose first that the leftmost bit of c_1 is at b_1. Since w is a square, the bits of c_1 appear also in w_R, with the leftmost bit at b_{m+1}. By (iv) and Proposition 1, actually c_1 appears in w_R, left-justified, and the complement of of this left-justified c_1 with respect to w_R is tiled uniquely with an integer number of codewords c_i. The same codewords then appear, shifted left by m places, in the complement of the left-justified c_1 of w_L with respect to w_L. Since the parsing is unique and w contains no part-codewords, the decoding exists, and so q contained a square, which is a contradiction. The same contradiction results if we assume that the rightmost bit of c_1 is at position b_m.

We may thus assume that c_1 is neither right- nor left-justified in w_L. Without loss of generality we may assume that c_1 is the leftmost codeword contained entirely in w_L. Since w is a square, Proposition 1 implies that c_1 also appears in w_R, at a unique location, namely right-shifted by m places from its location in w_L. Thus c_1 begins at some location $j+1 > m+1$, and so at location $j \geq m+1$, a codeword c_2 ends, which begins at some location $k \leq m$.

Suppose first that at least 8 of the bits of the suffix of c_2 are in w_R. We then use the following *left-shift argument*.

From the mapping C defined in Procedure TQB we see that a suffix of length ≥ 8 determines c_2 uniquely, when also the location j of the end of c_2 is given. (Knowing this location is crucial: note that the suffix of length 13 of $C(a_4)$ is identical to a subword of length 13 contained in the interior of $C(a_3a_5)$.) Since w is a square, it follows that at location $j-m \geq 1$ there is the end of the codeword c_2, which begins at location $k-m < 1$.

Again using the fact that w is a square we now have, in particular, $b_i = b_{i+m}$ for $i = k-m, \ldots, k-1$, i.e., we have another square

$$w' = b_{k-m} \cdots b_{k-1} b_k \cdots b_{k+m-1} = w'_L w'_R,$$

also of length $2m$, shifted left of w by $m-k$ bits, where $w'_L = b_{k-m} \cdots b_{k-1}$ and $w'_R = b_k \cdots b_{k+m-1}$. Now w'_R begins with a codeword and ends with one. As we saw above this implies that q has a square, which is a contradiction. This ends the left-shift argument.

We end the proof by considering four cases for the length of the suffix of c_2.

I. Assume that c_2 has a suffix of precisely 7 bits in w_R. The mapping C reveals that then c_2 is uniquely determined, except when $c_2 = C(a_1)$ or $C(a_4)$. When c_2 is uniquely determined, then the left-shift argument applies as above.

So assume first that $c_2 = C(a_1)$. If $C(a_1)$ intersects also the beginning of w_L, then the left-shift argument applies. Thus assume $C(a_4)$ intersects the beginning of w_L. By Table 1, $C(a_4)$ is followed by $C(a_3)$. Since w is a square, $C(a_3)$ must follow $C(a_1)$ in w_R. By Table 2, this $C(a_3)$ must be followed by $C(a_5)$. If this $C(a_5)$ is contained in w_R, then $C(a_5)$ must follow $C(a_3)$ in w_L. Thus $C(a_4a_3a_5)$ intersects w_L. This is a contradiction, since the triple $a_4a_3a_5$ doesn't appear in Table 2 (since t doesn't contain $a_2a_3a_2$). If $C(a_5)$ is not contained entirely in w_R, then the end of $C(a_3)$ and the beginning of $C(a_1)$ in w_L are adjacent bits. Since w is a square, the first 5 bits of $C(a_1)$ and $C(a_5)$ must then agree, but they don't.

Secondly, assume that $c_2 = C(a_4)$. If $C(a_4)$ also intersects the beginning of w_L, the left-shift argument applies. So assume that $C(a_1)$ intersects the beginning of w_L. By Table 2, $C(a_4)$ is followed by $C(a_3a_1)$ (since a_3a_2 cannot appear in q). Note that $C(a_3)$ must then be contained in both w_R and w_L. If $C(a_3a_1)$ is contained in w_R, then $C(a_3a_1)$ also appears after $C(a_1)$ in w_L. But then q and hence t contained $a_1a_3a_1$, which is a contradiction. If $C(a_1)$ is not contained entirely in w_R, then the end of $C(a_3)$ and the beginning of $C(a_4)$ in w_L must be adjacent bits. This is impossible, since q doesn't contain a_3a_4.

II. Assume that c_2 has a suffix of precisely 6 bits in w_R. Then case I applies a fortiori, and the same proof is valid. But now, in addition, $C(a_3)$ and $C(a_5)$ have the same suffix (of 6 bits).

Assume first that $c_2 = C(a_3)$. The only case that needs to be considered is when $C(a_5)$ intersects the beginning of w_L. It is followed by $C(a_2)$ (Table 1). Then $C(a_2)$ follows $C(a_3)$ in w_R, which is a contradiction, since q doesn't contain a_3a_2.

Secondly, assume that $c_2 = C(a_5)$. Then $C(a_5)$ has a prefix of length 8 in w_L, which is seen to be unique, so a right-shift argument, analogous to the left-shift argument, applies.

III. Assume that c_2 has a suffix of precisely 5 bits in w_R. Then case II applies a fortiori, but also $C(a_1), C(a_2)$ and $C(a_4)$ have the same suffix (of 5 bits).

Suppose first that $c_2 = C(a_1)$ and $C(a_2)$ intersects the beginning of w_L. Now Table 1 shows that $C(a_2)$ is followed by $C(a_1)$ or $C(a_4)$. The former is impossible since then q contains the square a_1^2, and the latter is impossible since then q contains a_1a_4. So assume $c_2 = C(a_2)$ and $C(a_1)$ intersects the beginning of w_L.

Now $C(a_1)$ is followed either by $C(a_2)$ or $C(a_3)$. The former is impossible, since q doesn't contain a square a_2^2, and the latter is impossible since q doesn't contain a_2a_3.

Secondly, assume that $c_2 = C(a_2)$ and $C(a_4)$ intersects the beginning of w_L. Now $C(a_4)$ is followed by $C(a_3)$, so $C(a_3)$ must follow $C(a_2)$ in w_R, which is impossible, since q doesn't contain a_2a_3. If $c_2 = C(a_4)$ and $C(a_2)$ intersects the beginning of w_L, we get the same contradiction.

IV. Assume that c_2 has a suffix of ≤ 4 bits in w_R. Then c_2 has a prefix of ≥ 8 bits at the end of w_L which determines c_2 uniquely, so a right-shift argument applies.

Thus the assumption $|q'| \geq 4$ leads to a contradiction in all cases, hence $|q'| \leq 3$. ∎

A computer program verified that for all the triples in Table 3, the only squares in the code of these triples are the obvious ones: 0^2, 1^2 and $(01)^2$. This completes our proof that $g(3) = \infty$.

Acknowledgment. We would like to thank Justin Carpenter for his invaluable help with the computations.

References

1. J. Berstel [1992], Axel Thue's work on repetitions in words, in: *Séries Formelles et Combinatoire Algébrique* (P. Leroux and C. Reutenauer, eds.), Publ. du LACIM, Vol. 11, Université de Québec, à Montréal, pp. 65-80.
2. J. Berstel [$\geq$ 1995], Axel Thue's papers on repetition in words: an English translation, Publ. du LACIM, Université de Québec, à Montréal.
3. S. Burris and E. Nelson [1971/72], Embedding the dual of π_∞ in the lattice of equational classes of semigroups, *Algebra Universalis* **1**, 248–153.
4. J.D. Currie [1993], Open problems in pattern avoidance, *Amer. Math. Monthly* **100**, 790–793.
5. A. del Junco [1977], A transformation with simple spectrum which is not rank one, *Canad. J. Math.* **29**, 655–663.
6. A. Ehrenfeucht and G. Rozenberg [1983], On the separating power of EOL systems, *RAIRO Inform. Théor.* **17**, 13–22.
7. R. Entringer, D. Jackson and J. Schatz [1974], On nonrepetitive sequences, *J. Combin. Theory* (Ser. A) **16**, 159–164.
8. D. Hawkins and W.E. Mientka [1956], On sequences which contain no repetitions, *Math. Student* **24**, 185–187.

9. A.J. Jeffreys, M. Turner and P. Debenham [1991], The efficiency of multilocus DNA fingerprint probes for individualization and establishment of family relationships, determined from extensive casework, *Am. J. Hum. Genet.* **48**, 824–840.

10. A.J. Jeffreys, V. Wilson and S.L. Thein [1985], Hypervariable 'minisatellite' regions in human DNA, *Nature* **314**, 67–73.

11. A.J. Jeffreys, V. Wilson and S.L. Thein [1985], Individual-specific 'fingerprints' of human DNA, *Nature* **316**, 76–79.

12. J.A. Leech [1957], A problem on strings of beads, *Math. Gaz.* **41**, 277–278.

13. A. Milosavljević [$\geq$ 1995], Repeat Analysis, Ch. 13, Sect. 4, Imperial Cancer Research Fund Handbook of Genome Analysis, Blackwell Scientific Publications, in press.

14. M. Morse and G.A. Hedlund [1944], Unending chess, symbolic dynamics and a problem in semigroups, *Duke Math. J.* **11**, 1–7.

15. P.S. Novikov and S.I. Adjan [1968], Infinite periodic groups I, II, III, *Izv. Akad. Nauk. SSSR Ser. Mat.* **32**, 212–244; 251–524; 709–731.

16. P.A.B. Pleasants [1970], Non-repetitive sequences, *Proc. Cambridge Phil. Soc.* **68**, 267–274.

17. I. Raskó and C.S. Downes [1995], *Genes in Medicine: Molecular Biology and Human Genetic Disorders*, Chapman & Hall, London.

18. P. Roth [1991], ℓ-occurrences of avoidable patterns, in: 8th Annual Sympos. Theoretical Aspects of Computer Science (STACS; C. Choffrut and M. Jantzen, eds.), Hamburg, Lect. Notes in Comp. Sci. 480, Springer-Verlag, pp. 42–49.

19. A. Thue [1912], Über die gegenseitige Lage gleicher Teile gewisser Zeichenreihen, *Norske Vid. Selsk. Skr., I. Mat. Nat. Kl. Christiania* **I**, 1–67.

20. E.N. Trifonov [1989], The multiple codes of nucleotide sequences, *Bull. Math. Biology* **51**, 417-432.