A Fourfold Congruence for Binary Words with Exactly Two Crossing Squares

Augusto Rehfeldt

Text generated by language models in the mathforge pipeline; not reviewed by a person.

Abstract

We consider binary words having exactly two square occurrences whose intervals overlap without either containing the other. Refined by word length, the two sorted root lengths, and intersection length, their counts are divisible by four except at four explicit profiles, where the residue is two. The proof classifies the possible crossing configurations and the reversal-fixed words, then applies reversal–complementation orbit counting. Exhaustive computations support the result through length 18, with an independent check through length 16. The supplied Lean 4 formalization compiles with one `sorry`; it is not a machine-checked proof.

1. Introduction

A square is a nonempty word repeated twice. Counting its **occurrences**, rather than its distinct spellings, makes location and interval geometry essential: even a constant word of length three contains two distinct squares.

Here we study words with exactly two square occurrences that cross: their intervals intersect, but neither contains the other. The main result is a congruence modulo four refined simultaneously by ambient length, both root lengths, and overlap length. Its proof combines a local rigidity observation—each root has length at most two—with the action generated by reversal and binary complementation.

The proposed new contribution is this profile-specific congruence and its four exceptional profiles, not the underlying symmetry argument. Originality remains provisional under the bounded search described in Section 5. No prefix-extension recurrence is established here.

2. Notation

A binary word of length n is a sequence $w_1 \cdots w_n$, with $w_i \in \{0, 1\}$. A square occurrence is a pair (i, r) satisfying

$$r \geq 1, \qquad 1 \leq i \leq n - 2r + 1, \qquad w_{i+j} = w_{i+r+j} \quad (0 \leq j < r).$$

Its root length is r, and its interval is

$$\{i, \dots, i + 2r - 1\}.$$

Distinct pairs are distinct occurrences, even if they spell the same word.

Overlap means nonempty intersection; containment includes equality. A crossing pair overlaps without either interval containing the other.

For $n \geq 1$, $1 \leq a \leq b$, and $\ell \geq 1$, let $C(n, a, b, \ell)$ count words having exactly two square occurrences, which cross, have sorted root lengths a, b, and have intersection length ℓ. This counts words, not occurrence pairs.

3. Results

Theorem (Fourfold congruence)**.** *For every integer $n \geq 1$, every pair of integers $1 \leq a \leq b$, and every integer $\ell \geq 1$,*

$$C(n,a,b,\ell) \equiv \begin{cases} 2 \pmod 4, & (n,a,b,\ell) \in \{(3,1,1,1),(5,1,1,1),(7,1,1,1),(5,2,2,3)\}, \\ 0 \pmod 4, & \textit{otherwise.} \end{cases}$$

The following proof is reproduced verbatim from the supplied verified proof.

Proof.

Let R denote reversal and let K denote complementation. Both operations preserve square occurrences, their root lengths, and the lengths of intersections of their intervals.

We first determine the possible overlapping configurations.

Lemma 1. *Every binary square of root length at least 3 contains a square occurrence whose interval is strictly smaller than its own interval.*

Proof. Consider the first four letters of the square. If two consecutive letters are equal, they form a square of root length 1. Otherwise those four letters alternate and form a square of root length 2. In either case, the resulting interval has length at most 4, whereas the original square has length at least 6.

□

Lemma 2. *In a word counted by $C(n,a,b,\ell)$, both square occurrences have root length at most 2.*

Proof. If one occurrence had root length at least 3, Lemma 1 would give another square properly contained in it. This contained occurrence cannot be the other prescribed occurrence, because neither prescribed interval contains the other. There would therefore be at least three occurrences, contrary to the definition of C.

□

Lemma 3. *The union of the two square intervals in an admissible word spells one of*

$$000, \qquad 01010, \qquad 00101, \qquad 01011,$$

or a complement of one of these words.

The first two configurations have profiles

$$(a,b,\ell) = (1,1,1), \qquad (2,2,3),$$

respectively. The last two have profile $(1,2,1)$*.*

Proof. Number positions in the union starting at 0, and put the leftmost occurrence first.

If both roots have length 1, crossing requires their starting positions to differ by 1. Their union is therefore xxx, giving 000 or 111.

Suppose both roots have length 2, and let d be the difference between their starting positions. Crossing requires $d \in \{1,2,3\}$. There can be no equal adjacent letters anywhere in their union: such letters would give a third square occurrence. Consequently the union must alternate.

If $d = 1$, its length is 5, giving 01010 or its complement. If $d = 2$ or 3, the alternating union has length at least 6, so it has length-4 square occurrences starting at positions $0, 1, 2$. This contradicts the requirement of exactly two occurrences.

Finally, suppose the root lengths differ. If the length-2 interval comes first, crossing forces the length-4 interval to start at position 1. The union has the form

$$xxyxy.$$

We must have $x \neq y$, since otherwise it contains more than two adjacent-equal occurrences. Thus it is 00101 or its complement.

If the length-4 interval comes first, crossing forces the length-2 interval to start at position 3. The union has the form

$$xyxyy,$$

and again $x \neq y$. This gives 01011 or its complement. The intersection lengths follow directly from these starting positions.

□

Lemma 4. *No admissible word is fixed by RK.*

Proof. Suppose $RK(w) = w$. Reversal sends an occurrence with interval $[i, j]$ to an occurrence with interval

$$[n + 1 - j, n + 1 - i],$$

and complementation does not change which positions form an occurrence. Hence RK permutes the two square occurrences.

If it fixes each occurrence, both intervals have midpoint $(n + 1)/2$. Two intervals with the same midpoint are nested, contradicting the required noncontainment.

It must therefore exchange the occurrences. Their root lengths are then equal. By Lemma 3 their union is either a constant word of length 3 or an alternating word of length 5. The union interval is invariant under reflection and has odd length, so its middle position is the middle position of the entire word. The equation $RK(w) = w$ would require the bit at that position to equal its complement, which is impossible.

□

Lemma 5. *The admissible words fixed by reversal are precisely*

$$000, \quad 10001, \quad 0100010, \quad 01010$$

and their complements.

Proof. Suppose $R(w) = w$. As in Lemma 4, reversal cannot fix both occurrences individually, because their intervals would have a common midpoint and would be nested. Thus reversal exchanges them. They have equal root lengths, and Lemma 3 shows that their union is a centered copy of 000, 111, 01010, or 10101.

First consider the core 000. Write the word as

$$u\,000\,R(u),$$

and let $k = |u|$. The two adjacent-equal occurrences inside 000 already account for both allowed squares. Thus every adjacent pair outside those two pairs must have unequal letters. Starting from the core and moving outward, the letters are consequently forced to alternate, beginning with 1.

For $k = 0, 1, 2$, this gives respectively

$$000, \qquad 10001, \qquad 0100010.$$

If $k \geq 3$, the three letters immediately preceding the core are 101; together with the first letter of the core they form 1010, an additional square. Hence $k \leq 2$.

Now consider the core 01010. Its two prescribed occurrences have root length 2, so no adjacent letters anywhere may be equal. Any extension must therefore continue the alternation. If there is a letter before the core, that letter and the first three core letters form 1010, an additional square. Thus no extension is possible.

Complementation gives the corresponding conclusions for the cores 111 and 10101.

It remains to verify that the listed words really have exactly two occurrences. For 000, only its two adjacent-equal pairs are squares. In 10001, the length-4 factors are 1000 and 0001, neither a square; its only adjacent-equal pairs are those inside 000.

In 0100010, the length-4 factors are

$$0100,\ 1000,\ 0001,\ 0010,$$

and the length-6 factors are 010001 and 100010. None is a square. Its only adjacent-equal pairs are again those inside 000.

Finally, 01010 has no adjacent-equal pair and has exactly the two length-4 squares 0101 and 1010. These checks exhaust all possible root lengths in the listed words. Complements have identical occurrence profiles.

□

Lemma 6. *For each fixed profile, the number of admissible words is congruent modulo* 4 *to the number of its reversal-fixed words.*

Proof. The commuting involutions R and K act on the finite set of words with that profile. No binary word is fixed by K, since complementation changes every bit. By Lemma 4, no admissible word is fixed by RK.

A reversal-fixed word therefore has orbit consisting of itself and its complement. Both words in this orbit are reversal-fixed, because R and K commute.

Every other word has four distinct orbit members

$$w, \quad R(w), \quad K(w), \quad RK(w).$$

Indeed, an equality between two of these would make w fixed by one of R, K, RK. Thus the words not fixed by reversal occur in groups of four, proving the congruence.

□

Conclusion

By Lemma 5, there are exactly two reversal-fixed admissible words in each of the profiles

$$(3,1,1,1), \qquad (5,1,1,1), \qquad (7,1,1,1), \qquad (5,2,2,3),$$

namely the displayed word and its complement. There are none in any other profile.

Lemma 6 therefore gives residue 2 modulo 4 for precisely these four profiles, and residue 0 modulo 4 otherwise. This proves the theorem. □

4. Computational verification

The following summarizes supplied verification reports; no new computations were performed for this paper.

Exhaustive search

Every binary word of lengths $1 \le n \le 18$ was enumerated. Square occurrences were recorded, rejecting a word upon finding a third. Words with exactly two occurrences were tested for crossing and counted by sorted root lengths and intersection length.

The tested parameter ranges were

$$1 \le a \le b \le \lfloor n/2 \rfloor, \qquad 1 \le \ell \le 2a - 1.$$

Parameters outside these bounds have count zero by interval geometry.

The report records:

- 524,286 binary words tested;
- 1,365 parameter cases tested;
- no counterexample;
- agreement of independent occurrence routines on all 510 words of lengths 1 through 8;
- the direct check $C(3, 1, 1, 1) = 2$.

Independent check

A separate reported check exhaustively enumerated words through length 16 and binary squares through root length 12. It confirmed the root-length restriction, crossing cores and profiles, absence of reverse-complement fixed words, exact reversal-fixed classification, profile-wise congruence, and actual group orbits. All reported checks passed. These finite checks support, but do not replace, the unrestricted proof.

Lean 4 status

Theorem	Lean 4 status	Back-translation verdict
Fourfold congruence	**compiles with `sorry`**	**FAITHFUL**

The compilation report contains one `sorry` and reports that `BinarySquares.main_theorem` depends on `sorryAx`, alongside `propext`, `Classical.choice`, and `Quot.sound`. It is therefore **not machine-checked sorry-free**.

The supplied back-translation matches the informal theorem: Boolean words are counted once; exactly two distinct square occurrences are required; crossing, sorted roots, overlap length, hypotheses, and all four exceptions agree. Zero-based starts and exclusive endpoints translate to the informal intervals, and $n \ge 1$ makes `n.toNat` exact. No differences or trivialization were reported.

A sorry-free Lean proof certifies the **Lean statement**, which still has to be read against the informal one; compilation alone does not establish that correspondence or prove a theorem containing `sorry`.

5. Limitations and open questions

The supplied informal referee report judged the proof valid and recorded no issues. No explicit `## GAP` markers occur in the supplied results. The remaining limitations are nevertheless substantive:

- The Lean proof contains an unproved step.
- Computational verification is finite.
- Novelty is provisional, with incomplete retrieval documentation and failed arXiv requests.
- The result gives residues, not exact counts or generating functions.
- Prefix-extension recurrences tracking suffixes that would create a third square remain undeveloped. Such recurrences, and extensions to larger alphabets or more square occurrences, are natural further questions.

Novelty

The evidence base is recorded as `retrieval`, not `memory-only`. The record lists one search round with the following exact queries, submitted against arXiv, Crossref, and OpenAlex:

1. `binary words exactly two square occurrences overlapping non-nested squares root lengths overlap length enumeration modulo 4 exceptional lengths 3 5 7`
2. `combinatorics on words enumeration binary words prescribed number of square occurrences overlapping repetitions congruences`
3. `binary words reversal complementation Klein four-group action orbit counting modulo 4 palindromes antipalindromes square occurrences`

All three arXiv requests timed out; no arXiv hits were recorded. The supplied retrieval list contains **24 returned entries: 14 from Crossref and 10 from OpenAlex**. Per-query counts and total index hit counts were not supplied, so these numbers describe recorded entries, not all possible matches. No matching result was identified in that record.

The record also proposes three follow-up queries, but does not document their execution: `"binary words" "exactly two squares" palindrome`, `"square occurrences" "modulo 4"`, and `"binary words" "two square occurrences" reversal`.

This is a **bounded automated search over those indexes—not a literature review**. It is blind to books, journals outside them, and anything phrased differently. The arXiv portion failed, and originality is not established. The supplied assessment labels the result "apparently new"; that is a provisional assessment, not a priority claim.

Models

A fourfold congruence with four exceptional overlap profiles (`verified`)

- Proposed the claim: gpt-6.1-sol
- Counterexample search: gpt-6.1-sol
- Wrote the proof: gpt-6.1-sol
- Referee: gpt-6.1-sol
- Independent check script: gpt-6.1-sol
- Lean formalization: gpt-6.1-sol
- Faithfulness judge: gpt-6.1-sol