Counting overlapping pairs of strings

Eric Rivals, Pengfei Wang

Counting overlapping pairs of strings

Eric Rivals
LIRMM, Université Montpellier, CNRS, Montpellier, France

Pengfei Wang
LIRMM, Université Montpellier, CNRS, Montpellier, France

Abstract

A correlation is a binary vector that encodes all possible positions of overlaps of two words, where an overlap for an ordered pair of words (u, v) occurs if a suffix of u matches a prefix of v. As multiple pairs can have the same correlation, it is relevant to count how many pairs of words share the same correlation depending on the alphabet size and word length n. We exhibit recurrences to compute the number of such pairs – which is termed *population size* – for any correlation; for this, we exploit a relationship between overlaps of two words and self-overlap of one word. This theorem allows us to compute the number of pairs with a longest overlap of a given length and to show that the expected length of the longest border of two words asymptotically diverges, which solves two open questions raised by Gabric in 2022. Finally, we also provide bounds for the asymptotic of the population ratio of any correlation. Given the importance of word overlaps in areas like word combinatorics, bioinformatics, and digital communication, our results may ease analyses of algorithms for string processing, code design, or genome assembly.

2012 ACM Subject Classification Mathematics of computing → Discrete mathematics

Keywords and phrases combinatorics, correlation, overlap, border, lattice, asymptotics, bounds, expectation, word

Digital Object Identifier 10.4230/LIPIcs...

Category Regular Paper

Funding E. Rivals and P. Wang are supported by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 956229.

1 Introduction

A word u overlaps a word v if a suffix of u equals a prefix of v. The shared suffix-prefix is called a *border* for the ordered pair of words (u, v) (note that other authors call this a *right border*, see [7]). If (u, v) has no border it is said *unbordered*. The pair (u, v) is said *mutually unbordered* if both (u, v) and (v, u) lack a border. Conversely, if both (u, v) and (v, u) have a border, then the pair is said to be *mutually bordered*. These notions generalize to pairs of words, the well studied notions of border, bordered and unbordered words, that were originally defined for single words.

Overlapping and unbordered words are central in many applications: bioinformatics, pattern matching, or code design. Computing overlaps between all pairs of sequencing reads is one step of the genome assembly task [10, 21]; several algorithms solve it in optimal time [11, 34, 16, 33]. The notion of borders is core in word combinatorics [20, 19], the design of pattern matching algorithms [15, 32], and in the statistical analysis of pattern finding and discovery [23, 5]. For instance, questions in vocabulary statistics deal with the distributions of the number of missing words or of common words in random texts [25, 26], which depend on the overlap structure of words, and find applications in bioinformatics [31] or in the test of random number generators [24]. Set of mutually unbordered words serve as code for synchronization purposes in network communication. A seminal construction algorithm appeared in 1973 [22], and others brought recent improvements in the design of cross bifix-free codes [3, 1] or non-overlapping code [2], a topic of combinatorial interest [4].

The combinatorics of single (not pair) bordered and unbordered words over a q-ary alphabet has been studied in depth. For instance, the recurrence for counting the number of unbordered words of length n over a q-ary alphabet was first given in [22], while the recurrence for counting the number of bordered words (termed "overlapping sequences" in some articles) is proven in [18]. From this, the probability that a random word of length n is unbordered was shown to converge when n tends to ∞ in [22], while Holub and Shallit have shown that expected maximum border length for words of length n over a q-ary alphabet also converges [14]. In a related area (but somehow more distant from our topic), other works have investigated unbordered factors of words [13, 17], a topic introduced by Ehrenfeucht and Silberger in [6].

Recently, building on ideas similar to those used in [22], Gabric gave three recurrences to count bordered, mutually bordered, mutually unbordered pairs of words of length n over a k-ary alphabet [7]. In his conclusion, he raised challenging open questions: 1/ count the number of pairs having a longest border of length i (with i satisfying $0 < i < n$), and 2/ what is the expected length of the longest border of a pair of words? We address and solve both questions in our work (see Section 5).

Example: Consider the binary alphabet $\{a, b\}$ and the following three words denoted by u, v, w: `abaaa`, `aaabb`, and `abbbb`. The pairs (u, v) and (v, w) both have a longest border of length 3, but (u, v) has 3 distinct non empty borders `aaa`, `aa`, and `a`, while (v, w) has only one `abb`. The pairs (v, u) and (w, v) have no borders, which illustrates the assymetry of this notion.

First, this example illustrates that the possibilities of overlap of a pair (u, v) depends on the self-overlapping structure of their longest border (compare `aaa` with `abb`). Second, it shows that the self overlap structure of the border limits the number of words having such a shared suffix-prefix, and thus the number of pairs of words to count. Indeed, only words of length 5 having a suffix (resp. prefix) such as `aaa` or `bbb`, can participate in a pair having as much and as long borders as (u, v). These observations suggest that to answer the open

question raised by Gabric, one may have to account for the complete overlap structure of a pair of words.

Other authors have proposed to encode the starting position of such overlaps in a binary vector called a *correlation* [8]. In our example, the correlation of the pair (u, v) is `00111`, while that of (v, w) is `00100`. For any word z, the correlation of (z, z) is called the *autocorrelation* of z. Clearly, multiple pairs can have the same correlation, and hence there are less correlations of length n than pairs of words of length n.

Fortunately, one can build on earlier studies of set of autocorrelations, denoted Γ_n, and the set of correlations, denoted Δ_n, for all possible strings of length n [8, 9, 28, 29]. It is known the self overlap structure of a word [8], as well as the overlap structure of a pair of words [30], does not depend on the alphabet size (provided the alphabet has at least two letters – a unary alphabet makes these questions trivial). Combining a characterization of Δ_n provided in [30] and algorithm for enumerating Γ_n [27], we can enumerate Δ_n to get the list of all correlations of length n.

With the terminology used in [8, 29, 26], we exhibit two solutions to compute the population size of any correlation, that is the number of pairs of words having the same correlation (in Section 3). For this, we exploit two recurrences to compute the population size of autocorrelations [8, 28]. With this in hand, we derive in Section 5 a formula for the abovementioned open question 1/ (Corollary 25), and show that expected length of the longest border of words of length n asymptotically diverges (open question 2/ - Theorem 26). Besides this, we provide bounds for the asymptotic of the population ratio of any correlation (Theorem 23 Section 4), which extend the result known for autocorrelations [8]. Finally, we conclude with some open questions (Section 6).

2 Preliminaries

Let Σ be a finite *alphabet*, that is a set of *letters* of cardinality σ. We call a sequence of elements of Σ a *string* or a *word*. The empty word is denoted by ε. We denote by Σ^* the set of all finite strings over Σ, and by Σ^n the set of all strings of length n over Σ, with $n \in \mathbb{N}$. For a string x, $|x|$ denotes the *length* of x. For two strings x, y, we denote their concatenation by xy, and the k-fold concatenation of x with itself by x^k for any $k > 0$. For any $L \subset \Sigma^*$, we define $x.L$ as $\{xy : y \in L\}$.

Let u be a string of Σ^n. We index the letters of u from 0 to $n-1$: $u = u[0] \dots u[n-1]$. The ith letter of u is denoted by $u[i]$. We also denote by $u[i..j]$ for any $0 \leq i \leq j < n$ the substring of u starting at position i and ending at position j. A substring is said to be *proper* iff $j - i + 1 < n$. Moreover, $u[0..j]$ is a prefix, $u[i..n-1]$ is a suffix of u.

2.1 Definitions of borders and correlation for pairs of strings

To study overlaps between two words, we consider ordered pairs of strings: we denote a pair of strings $(u, v) \in \Sigma^n \times \Sigma^m$, which differs from the pair (v, u).

▶ **Definition 1** (Border of pair of strings). *A border of a pair of strings $(u, v) \in \Sigma^n \times \Sigma^m$ is any string that is a non-empty suffix of u, and a non-empty prefix of v. If a border exists, (u, v) is said* bordered, *otherwise it is* unbordered.

A pair may have multiple borders, and in general the set of borders for (u, v) differs from that of (v, u). In his article, Gabric refers to a border of (u, v) as a right border and to a border of (v, u) as a left border; we use a different terminology.

pos.	0	1	2	3	4	5	6	7	8	9	10	
u	a	a	b	b	a	b	-	-	-	-	-	t
v	b	a	b	b	a	a	-	-	-	-		0
	-	b	a	b	b	a	a	-	-	-	-	0
	-	-	b	a	b	b	a	a	-	-	-	0
	-	-	-	b	a	b	b	a	a	-	-	1
	-	-	-	-	b	a	a	b	a	b	-	0
	-	-	-	-	-	b	a	b	b	a	a	1

Table 1 Example of correlation for the pair $(u, v) := (aabbab, babbaa)$ of length 6. All possible shifts of v to the right of u are displayed on distinct lines: those at which an overlap exists are colored in blue. The last column shows $c(u, v)$ written top-down, with 1 bits corresponding to borders also colored in blue.

Guibas & Odlyzko [9] proposed to encode in a binary vector the positions in u at which a border is starting, and they named this notion the *correlation* of a pair of strings. From now on, for the sake of simplicity, we focus on pairs of strings of equal length, denoted n, although our results can be generalized to the case of unequal lengths.

▶ **Definition 2** (Correlation). *Let $(u, v) \in \Sigma^n \times \Sigma^n$. The correlation of (u, v), denoted $c(u, v)$, is a binary vector of length n (i.e., $c(u, v) \in \{0, 1\}^n$) satisfying $\forall i \in [0, \dots, n-1]$*

$$c(u,v)[i] = \begin{cases} 1 & \text{if } u[i..n-1] = v[0..n-i-1] \\ 0 & \text{otherwise.} \end{cases}$$

Generally $c(u, v) \neq c(v, u)$. For any length $n \in \mathbb{N}$, we denote the set of all correlations for words of length n by Δ_n, and its cardinality by δ_n as in [29].

▶ **Definition 3** (Δ_n and δ_n). *Let $n \in \mathbb{N}$. The set of all correlations of words of length n is:*

$$\Delta_n := \{t \in \{0,1\}^n : \exists (u,v) \in \Sigma^n \times \Sigma^n : c(u,v) = t\},$$

and its cardinality is denoted δ_n.

▶ **Example 4.** Consider the pair of strings $(u, v) = (aabbab, babbaa)$ of length 6 over the binary alphabet $\{a, b\}$. The pair (u, v) has a border starting at position 3 in u, and a shorter border starting at position 5. Its correlation is $c(u, v) = 000101$. See Table 1. Of course, a permutation of the alphabet (that is exchanging a with b and vice versa) yields a different pair of strings, which has the same correlation as (u, v). Thus, several pairs can share the same correlation.

A special case arises when u equals v. Then $c(u, u)$ is called the *autocorrelation* of u, which for clarity, we will denote by $a(u)$. We recall definitions of period, period set, and autocorrelation, as well as some known properties of autocorrelations that we use later on. The proofs of properties can be found in [30, 8, 12].

▶ **Definition 5** (Period). *String $u = u[0..n-1]$ has period $p \in \{0, 1, \dots, n-1\}$ if and only if $u[0..n-p-1] = u[p..n-1]$, i.e. for all $0 \leq i \leq n-p-1$, we have $u[i] = u[i+p]$.*

The zero period is called *trivial*. The smallest non-trivial period of u is called its *basic period*. The *period set* of a string u is the set of all its periods and is denoted by $P(u)$.

▶ **Definition 6** (Autocorrelation). *For every string $u \in \Sigma^n$, its autocorrelation is the string $a(u) \in \{0,1\}^n$ such that*

$$a(u)[i] = \begin{cases} 1 & \text{if } i \in P(u) \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in \{0,\dots,n-1\}.$$

For all possible strings of length $n \in \mathbb{N}$, the set of autocorrelations, denoted by Γ_n, is defined as: $\Gamma_n := \{s \in \{0,1\}^n : \exists u \in \Sigma^n : a(u) = s\}$. We denote by κ_n the cardinality of Γ_n. Clearly, $\Gamma_n \subset \Delta_n$. When $n = 0$ we consider that $\Gamma_n = \{\varepsilon\}$.

▶ **Lemma 7.** *Let $s \in \Gamma_n$ and $u \in \Sigma^n$ such that $a(u) = s$. Let $0 \leq p \leq q < n$ such that $s[p] = 1$. Then, $s[q] = 1$ iff $u[p..n-1]$ has period $(q-p)$ (equivalently the $(q-p)$ bit in $a(u[p..n-1])$ equals 1).*

▶ **Lemma 8.** *Let $s \in \Gamma_n$. For all p satisfying $0 \leq p < n$, and $s[p] = 1$, it follows that $s[kp] = 1$ for all $k \in [2,\dots,\lfloor \frac{n}{p} \rfloor]$.*

▶ **Lemma 9.** *Let $\pi(u)$ be the basic period of $u \in \Sigma^n$, and p be a non-trivial period. Then either $p = k \cdot \pi(u)$, $k \in [1,\dots,\lfloor \frac{n}{\pi(u)} \rfloor]$ or $p > n - \pi(u)$.*

2.2 Set of all correlations of length n and its characterization

The first characterization of autocorrelations was given by Guibas and Odlyzko in their seminal paper [8]. They studied the cardinality of Γ_n and provided a lower and an upper bound for $\log(\kappa_n)/\log_2(n)$, and conjectured that their lower bound was also an upper bound. They also proposed an algorithm to compute the number of strings in Σ^n that share the same period set, which they termed the *population* of an autocorrelation. A key result of their work is the *alphabet independence* of Γ_n: Any alphabet with $\sigma > 1$ gives rise to the same set of autocorrelations, i.e., to Γ_n.

Rivals et al. [30] have characterized Δ_n and exhibited its relation to the sets Γ_j for $0 \leq j \leq n$, which is stated below.

▶ **Lemma 10** (Lemma 21 [30]). *The set of correlations of length n is of the form*

$$\Delta_n = \left\{ 0^{(n-j)}s, \quad \text{with } s \in \Gamma_j \quad \text{and} \quad j \in [0,\dots,n] \right\}.$$

Lemma 10 gives us the **structure of any correlation** for any pair of strings (u,v) of length n: it starts with a series of 0, until the leftmost 1, which marks the position in u of the longest border of pair (u,v). Let z denote this border and j denote its length. The above characterization is based on the fact that the suffix of length j of $c(u,v)$ (the one starting with the leftmost 1) must be the autocorrelation of z. Indeed, each border of z is also a border of (u,v). If $j = 0$, then z is empty string and $c(u,v) = 0^n$. Of course, if $u = v$, then the correlation of (u,v) is the autocorrelation of u.

This characterization implies the following **partition** of Δ_n:

▶ **Corollary 11.** $\Delta_n = \bigcup_{j=0}^{n} \{0^{n-j}s \mid s \in \Gamma_j\} = \bigcup_{j=0}^{n} \left(0^{n-j}.\Gamma_j\right)$.

Since correlations (and autocorrelations) are binary encoding of a set of positions, we can get the *intersection (or union)* of two correlations by taking their logical AND (or OR). For legibility, for $t, t' \in \Delta_n$ we denote their intersection by $t \cap t'$ and their union by $t \cup t'$. We use such notation to investigate the algebraic structure of Δ_n in Appendix A.

Rivals et al. [30] studied the cardinalities of Γ_n and Δ_n, and proved the asymptotic convergence of ratios involving κ_n and δ_n towards the same limit when n tends to infinity. Precisely, $\frac{\ln \delta_n}{\ln^2(n)} \to \frac{1}{2\ln(2)}$ when $n \to \infty$.

It is interesting to study the algebraic structure of Δ_n. In Appendix A, we show that Δ_n is a lattice under set inclusion, and that it does not satisfy the Jordan-Dedekind condition. The example 12 and Figure 1 illustrate the lattice structure of Δ_n for $n = 4$.

▶ **Example 12.** Figure 1 illustrates the lattice structure with Δ_4, for strings of length $n = 4$. From Corollary 11, one has $\Delta_4 = \Gamma_4 \cup (0.\Gamma_3) \cup (00.\Gamma_2) \cup (000.\Gamma_1) \cup \{0000\}$.

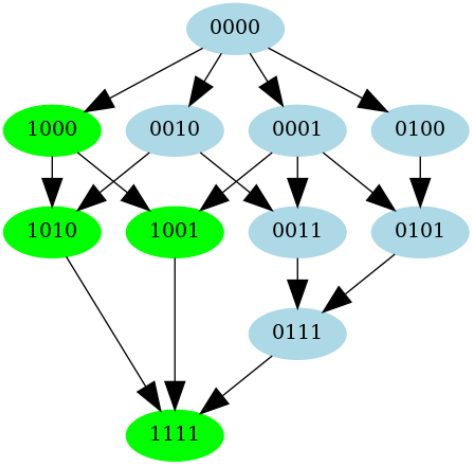

Figure 1 The lattice of Δ_4: each node contains a correlation as a binary vector. The elements of Γ_4 are colored in green. Since between 0000 and 1111 there are chains of different lengths (3 and 4), Δ_4 does not satisfy the Jordan-Dedekind condition.

Table 2 Pair population sizes on a binary alphabet for correlations of Δ_4; $\sigma = 2$ and $n = 4$.

correlation	pair population size
0000	74
0001	82
0010	30
0011	24
0100	16
0101	8
0111	6
1000	6
1001	6
1010	2
1111	2

3 Population size of a correlation

The population of a correlation $t \in \Delta_n$ is defined as: $POP(t) := \{(u, v) \in \Sigma^n \times \Sigma^n$ such that $c(u, v) = t\}$. We want to compute its cardinality, i.e. the *population size*, which we denote by $pop(t)$. For example, consider the correlation $t := 01010$ from Δ_5: over the alphabet $\Sigma = \{a, b\}$, we have $POP(t) = \{(ababa, babaa), (ababa, babab), (bbaba, babab), (bbaba, babaa), (aabab, ababa), (aabab, ababb), (babab, ababa), (babab, ababb)\}$ and $pop(t) = 8$.

Similarly, if s is an autocorrelation, that $s \in \Gamma_n$, we define the population of s as $POP(s) := \{u \in \Sigma^n$ such that $a(u, v) = s\}$ and denote by $pop(s)$ the cardinality of $POP(s)$.

Let us give an overview of our results and detail how they generalize or improve existing ones. First, for a given autocorrelation $t \in \Gamma_n$ there exists a linear time *realization* algorithm to build a binary string u such that $a(u) = t$ [27]. We will exhibit such a realization algorithm for any correlation $t \in \Delta_n$ in Section 3.1. In fact, this is related to counting not the pairs of $POP(t)$, but single strings either u or v, for which such pair exist. We show a formula to determine the cardinality of $POP_l(t) := \{u \in \Sigma^n : \exists v \in \Sigma^n$ such that $c(u, v) = t\}$ or of $POP_r(t) := \{v \in \Sigma^n : \exists u \in \Sigma^n$ such that $c(u, v) = t\}$. Note that as u and v play a

symmetrical role in $POP_l(t)$ and $POP_r(t)$, it implies that their cardinalities, denoted $pop_r(t)$ and $pop_l(t)$, must be equal. Clearly, $pop_r^2(t)$ is an upper bound for $pop(t)$.

Second, there exist two algorithms for computing the population size of an autocorrelation (i.e., when $t = a(u)$). A recurrence formula for the population size of an autocorrelation was proposed in [8][Theorem 7.1] and with it the authors investigated the asymptotics of the population size (Theorem 7.2)[1]. Another algorithm takes advantage of the fact that Γ_n, the set of autocorrelations of length n, forms a lattice with set inclusion [29]. We will review the recurrence formula from [8] (see page 9) and use it to propose one for correlations (Theorem 19 in Section 3.2). We will also review the recurrence formula from [29] (see page 10) and apply it to propose another one for correlations (Theorem 20).

3.1 Computing the single population size

First, we need a simple Lemma about occurrences of a suffix of a word.

▶ **Lemma 13.** *Let $i > 0$ and $j > 0$ be two integers. Let $u \in \Sigma^i$ and $v \in \Sigma^j$. If the first letter of v does not occur in u, then v occurs in uv only at position i.*

Let now us state the realization problem and describe our binary realization algorithm.
Problem: Consider the binary alphabet $\Sigma = \{a, b\}$. Let $n > 0$ and let $t \in \Delta_n$. Find a pair (u, v) of strings over Σ, such that $c(u, v) = t$.
Algorithm: If $t[0] = 1$, then t is an autocorrelation. Then, call the binary realization algorithm for autocorrelation with input t and return the obtained binary word [27]. If $t = 0^n$, the pair of strings shall not overlap at all. Thus $u := a^n$ and $v := b^n$ satisfy the correlation vector t. Otherwise, we know there exists $0 < j < n$ and $s \in \Gamma_j$ such that $t = 0^{n-j}s$. This is the **main case**.
Call the binary realization algorithm for autocorrelation with s as input, and denote by w the returned binary word. w has length j and must be the suffix of u and prefix of v. Without loss of generality, assume $w[0] = a$. Then, setting $u := b^{n-j}w$, and taking any v in the set $w.\Sigma^{(n-j)}$, we get

- w is border of (u, v), and thus s is a suffix of $c(u, v)$;
- w has only one occurrence in u by Lemma 13, and is thus the longest border of (u, v).

Hence, we get $c(u, v) = 0^{n-j}s$ as required. Finally, return (u, v) with $v := w.a^{n-j}$.

From this realization algorithm, in the **main case**, we see that for a fixed $t \in \Delta_n$, once w and u are chosen as above, there exist $\sigma^{(n-j)}$ pairs since v can be any word in $w.\Sigma^{(n-j)}$. This is a maximum for $pop_r(t)$ once w is fixed. Hence, we obtain the following Lemma to compute the single population size. A formal proof appears in Appendix B.

▶ **Lemma 14.** *Let $t := 0^{n-j}s$ be in Δ_n with $j \in [1, \ldots, n]$. Then the single population size of t satisfies: $pop_r(t) = pop(s) \cdot \sigma^{(n-j)}$.*

Remark: if $j = 0$, then the pair of strings (u, v) is unbordered. Note that if $vu \in \Sigma^{2n}$ with $|u| = |v| = n$ and is unbordered, then (u, v) is also unbordered. Therefore, all such pairs of strings (aka "bifix-free sequences") can be constructed by the algorithm of Nielsen [22].

[1] In their article, the authors mostly use the term "correlation" instead of autocorrelation.

3.2 Computing the pair population size

Before, finding a formula to compute $pop(t)$, i.e. the pair population size, of a correlation t in Δ_n, we show that $pop(t)$ is related to the population size of some autocorrelations of strings of length $2n$ in Theorem 17. To achieve this, we demonstrate two lemmas linking the borders of a pair (u, v) with the borders of the string vu.

▶ **Lemma 15.** *Let $t \in \Delta_n$ and let $(u, v) \in \Sigma^n \times \Sigma^n$ such that $c(u, v) = t$. Then, t is the suffix of length n of the autocorrelation of the word vu.*

Proof. Let $t \in \Delta_n$ and (u, v) be a pair of words as in the lemma. By Lemma 10, we know there exists $j \in [0, \dots, n]$ and $s \in \Gamma_j$ such that $t = 0^{(n-j)}s$. If we denote by z the longest border of (u, v), then $|z| = j$. We distinguish two cases depending on j.

Case 1. If $j = n$ then $u = v = z$, $t = s = a(u)$ and $vu = uu$. As the word uu has period $|u|$, then by Lemma 7, then t is the suffix of length n of $a(vu)$.

Case 2. Otherwise $j < n$. By hypothesis, there exist two words x and y of length $n - j$ such that $u = xz$ and $v = zy$. Hence, $vu = zyxz$ and z is a border of vu. As $|zyx| = 2n - j$, it implies that vu has period $2n - j$, and by Lemma 7 s is the suffix of length j of $a(vu)$. Let us show by contraposition that for any position $n \leq i < 2n - j$ the i-th bit of $a(vu)$ is `0`. Let i be a integer such that $n \leq i < 2n - j$ and assume the the i-th bit of $a(vu)$ equals `1`. Then, vu would have a border of length $2n - i$ with $n \geq 2n - i > j$, and this border would also be a border of (u, v), which contradicts the maximality of z. Hence, t is a suffix of $a(vu)$. ◀

▶ **Lemma 16.** *Let $w \in \Sigma^{2n}$; let u and v be words in Σ^n such that $w = vu$. If w has a border, then the pair of strings (u, v) is bordered.*

Proof. Let w, u, and v be as in the lemma. If $u = v$, then u is a border of the pair (u, u).

Otherwise, we have $u \neq v$. Let z be a border of w. We distinguish two cases based on $|z|$.

1. Case 1: $|z| \in [1, \dots, n - 1]$. Then, there exist two words x, y of length $n - |z|$ such that $v = zy$ and $u = xz$. Thus, z is a border of (u, v).
2. Case 2: $|z| \in [n + 1, \dots, 2n - 1]$. Then, w has a period $p := 2n - |z|$ and $p < n$ (the half $|w|$). According to properties of periods (Lemma 8), the integer $\lfloor \frac{2n}{p} \rfloor p$ is also period of w. Then, if we denote its corresponding border by z', we have $|z'| < n$, and we are back to case 1, with z' being a border of (u, v).

◀

Before stating the theorem on the population size of a correlation, we need a notation. Let $t \in \Delta_n$. We denote by $G(t)$ the set of all strings of length $2n$ whose autocorrelation has t as suffix, and by $g(t)$ the cardinality of $G(t)$. Formally, $G(t) := \{w \in \Sigma^{2n} : t \text{ is a suffix of } a(w)\}$.

The following theorem shows the relation between the number of pairs of strings of length n and the number of specific strings of length $2n$. For $t \in \Gamma_n$, $pop(t)$ can be directly calculated using Theorem 18. Therefore we consider $t \in \Delta_n$ but exclude those in Γ_n.

▶ **Theorem 17.** *Let $t \in \Delta_n \setminus \Gamma_n$. Then, $pop(t) = g(t)$.*

Proof. i/ Let us first prove that $pop(t) \leq g(t)$. Let $(u, v) \in POP(t)$. According to Lemma 15, the string vu belongs to $G(t)$. This implies that $pop(t) \leq g(t)$.
ii/ Let us prove that $pop(t) \geq g(t)$. Let $w \in G(t)$, and let u and v be strings of length n such that $w = vu$. As $t \in \Delta_n$, by Lemma 10, we know there exists $j \in [0, \dots, n]$ and $s \in \Gamma_j$ such that $t = 0^{(n-j)}s$.

If $j = 0$, then $t = 0^n$. We show that $a(w) = 10^{2n-1}$. Indeed, assume w has a period smaller than n, then by Lemma 8 it would also have periods $> n$, which contradicts $t = 0^n$. Thus, $c(u, v) = t$ and (u, v) belongs to $POP(t)$.

If $0 < j < n$. Then $w[0..j-1]$ is the longest border of w with length $j < n$. From Lemma 16 (specifically, case 1), we get that (u, v) belongs to $POP(t)$. In both cases, this implies that $g(t) \leq pop(t)$.

Combining both inequalities, we get $pop(t) = g(t)$, which concludes the proof. ◀

Now we will calculate the number of pairs of strings of length n with the correlation $t = 0^{n-j}s \in \Delta_n$ where $s \in \Gamma_j$, i.e., the population size of t. Thanks to Theorem 17, we provide two different approaches of computing $pop(t)$: the first one, based on the recurrence for the population size of autocorrelation using their recursive structure [8, Predicate Ξ], is presented in Section 3.2.1. The second approach based on the recurrence for the population size of autocorrelation which exploits the lattice structure of Γ_n [29], is shown in Section 3.2.2.

3.2.1 Recurrence based on the recursive structure of autocorrelations

We review the recurrence formula given by Guibas & Odlyzko. Let $s \in \Gamma_j$. They define the autocorrelation of length n denoted as $s_n := 10^{n-j-1}s$, and the sequence ψ for $k \in \mathbb{Z}$ depending on s as

$$\psi[k] := \begin{cases} 0 & \text{for } k > j \\ s[j-k] & \text{for } 1 \leq k \leq j \\ \sigma^{-k} & \text{for } k < 1. \end{cases}$$

We will use this definition of s_n in many places. The sequence ψ partitions $\mathbb{N}$ into three distinct ranges. For $k < 1$, $\psi[k]$ equals σ^{-k}. In the interval $1 \leq k \leq j$, $\psi[k]$ equals 1 if $(j-k)$ is a period in s, and 0 otherwise. For any $k > j$, $\psi[k]$ is consistently equal to 0. Theorem 18 states their recurrence for $pop(s_n)$.

▶ **Theorem 18** (Population size of an autocorrelation (Theorem 7.1 [8]))**.** *The number of strings of length n which have autocorrelation s_n satisfies the recurrence*

$$pop(s_n) + \sum_k pop(s_k)\psi[2k-n] = 2\psi[2j-n]pop(s),$$

where $pop(s_k) = 0$ for $k < j$.

We state our result regarding the population size of a correlation $t = 0^{n-j}s$ with s being fixed. See Table 2 for pair population sizes on a binary alphabet for all correlations in Δ_4. Note that if $j = n$, then the population size of t is the known population size of s.

▶ **Theorem 19** (Population size of a correlation (I))**.** *Let $\lambda, j, n \in \mathbb{N}$ satisfying $0 \leq \lambda, j < n$. Let $t := 0^{n-j}s$ be an element of Δ_n with $s \in \Gamma_j$. Then the population size of t satisfies the recurrence*

$$pop(t) = \sum_{\lambda=\lceil \frac{2n-j}{2} \rceil}^{n-1} pop(s_{(2n-\lambda)}) \cdot s[j+2\lambda-2n] + pop(s_{2n}).$$

Proof. Let $w \in G(t)$ and define the integer λ as $\lambda := \max\{0 \leq i < n : a(w)[i] = 1\}$. According to Theorem 17, we know that $pop(t) = g(t)$. Thus we are left to show

$$g(t) = \sum_{\lambda=\lceil \frac{2n-j}{2} \rceil}^{n-1} pop(s_{(2n-\lambda)}) \cdot s[j+2\lambda-2n] + pop(s_{2n}).$$

Define $s_{(2n,\lambda)} := *^{\lambda}10^{2n-\lambda-j-1}s \in \{0,1\}^{2n}$, where $*^{\lambda}$ is a short cut notation for any word in $\{0,1\}^{\lambda}$. Note that this defines a binary vector of length $2n$, which may belong to Γ_{2n} depending on the value of λ. Let us partition the set $G(t)$ into its subsets $POP(s_{(2n,\lambda)})$, where $s_{(2n,\lambda)} \in \Gamma_{2n}$

$$G(t) = \bigsqcup_{\lambda\in[0,\dots,n-1]:s_{(2n,\lambda)}\in\Gamma_{2n}} POP(s_{(2n,\lambda)}).$$

Taking the cardinalities, for $s_{(2n,\lambda)} \in \Gamma_{2n}$ we get

$$g(t) = \sum_{\lambda=0}^{n-1} pop(s_{(2n,\lambda)}) = \sum_{\lambda=1}^{n-1} pop(s_{(2n,\lambda)}) + pop(s_{(2n,0)}).$$

We distinguish different cases depending on λ.

1. When $\lambda = 0$. The autocorrelation of w satisfies $a(w) = s_{(2n,0)} = s_{2n} = 10^{2n-j-1}s \in \Gamma_{2n}$. Thus the number of strings w having the autocorrelation $s_{(2n,0)}$ equals the population size of s_{2n}, i.e., $pop(s_{(2n,0)}) = pop(s_{2n})$.
2. When $\lambda \in [1,\dots,n-1]$. Recall $s_{(2n-\lambda)} = 10^{2n-\lambda-j-1}s$, then we have $s_{(2n,\lambda)} = *^{\lambda}s_{(2n-\lambda)} \in \{0,1\}^{2n}$. Note that not all $s_{(2n,\lambda)}$ belongs to Γ_{2n}, but all $a(w)$ must have the form $s_{(2n,\lambda)}$. We will identify all elements $a(w)$ in Γ_{2n} that take the form $s_{(2n,\lambda)}$.
   By the definition of λ, we know $\lambda < |w|/2$, which indicates that at most one $a(w)$ can possibly exist for a given $s_{(2n-\lambda)}$. Note that $a(w)[2\lambda] = 1$ where $2\lambda \in [2n-j,\dots,2n-2]$, this implies $\lambda \geq \lceil (2n-j)/2 \rceil$. Denote by $\pi(w)$ the basic period of w, then $a(w)$ could be decomposed as $a(w) = (10^{\pi(w)-1})^{\alpha}s_{(2n-\lambda)}$ where $\alpha = \lambda/\pi(w)$ (a proper positive divisor) by Lemma 9. By Lemma 7, such an $a(w)$ exists precisely if $s[2\lambda-(2n-j)] = s[j+2\lambda-2n] = 1$ since $a(w)[2n-j] = 1$ and $j+2\lambda-2n \in [0,\dots,j-1]$. Thus we have

$$\sum_{\lambda=1}^{n-1} pop(s_{(2n,\lambda)}) = \sum_{\lambda=\lceil\frac{2n-j}{2}\rceil}^{n-1} pop(s_{(2n-\lambda)})s[j+2\lambda-2n].$$

Combining the two cases, we get $pop(t) = \sum_{\lambda=\lceil\frac{2n-j}{2}\rceil}^{n-1} pop(s_{(2n-\lambda)}) \cdot s[j+2\lambda-2n] + pop(s_{2n})$. ◀

Observe that in Theorem 19, calculating the population size of $t = 0^{n-j}s$ requires to compute $pop(s_{(2n-\lambda)})$ for all $\lambda \in [\lceil\frac{2n-j}{2}\rceil,\dots,n-1] \cup \{0\}$ by Theorem 18. Therefore, we provide another recurrence on t which calculates $pop(t)$ relying only on s. See details in Appendix C, Theorem 30.

3.2.2 Recurrence based on the lattice structure

As Γ_n equiped with inclusion is a lattice [29, Theorem 3.1], the successor of an autocorrelation s is a more constrained autocorrelation, i.e. one that contains more periods than s. One can use this relationship to compute population size. From the proof of Theorem 19, we know the autocorrelation of $w \in G(t)$ satisfies the form: $a(w) = (10^{\pi(w)-1})^{\lambda/\pi(w)}s_{(2n-\lambda)}$ where $s[j+2\lambda-2n] = 1$ for all $\lambda \in [\lceil\frac{2n-j}{2}\rceil,\dots,n-1] \cup \{0\}$. Clearly $\pi(w)|\lambda$. Thus, we can provide another recurrence by using the notion of *number of free characters (nfc for short)* introduced in [29]. The *nfc* of an autocorrelation $s \in \Gamma_n$ is the maximum number of positions in a string u with $a(u) = s$ that are not determined by the periods. For instance, the *nfc* of $100001001 \in \Gamma_9$ is 4 since a string u with $a(u) = 100001001$ must satisfy *character equations*: $u[0] = u[3] = u[5] = u[8], u[1] = u[6]$, and $u[2] = u[7]$. Thus $u = u[0]u[1]u[2]u[0]u[4]u[0]u[1]u[2]u[0]$ where $u[0], u[1], u[2], u[4] \in \Sigma$. Theorem 20 states the recurrence on population sizes.

▶ **Theorem 20.** *(Population size of an autocorrelation 2 (Theorem 6.1 [29])) Let $n \in \mathbb{N}$ and $s_{(k)}$ be the kth ($k = 1, \ldots, \kappa_n$) autocorrelation of Γ_n. Let $\rho_{(k)}$ denote the number of free characters of $s_{(k)}$. The population size $pop(s_{(k)})$ satisfies the recurrence*

$$pop(s_{(k)}) = \sigma^{\rho_{(k)}} - \sum_{j : s_{(k)} \subset s_{(j)}} pop(s_{(j)}).$$

The proof of Theorem 20 relies on the *nfc* of a given autocorrelation, on the lattice structure of Γ_n, and on the following idea. Consider the set $\mathcal{A}$ of words that satisfy the *character equations* imposed by autocorrelation $s_{(k)}$. As a word in $\mathcal{A}$ can satisfy additional character equations, $\mathcal{A}$ contains all words whose autocorrelation is $s_{(k)}$, but also words whose autocorrelations are $s_{(j)}$ with $j : s_{(k)} \subset s_{(j)}$. We reuse this idea to compute $pop(t)$.

Let $\lambda_1, \ldots, \lambda_m$ be the proper positive divisors of λ. From the proof of Theorem 19, the autocorrelation of $w \in G(t)$ could be decomposed based on λ_m, λ for $\lambda \in [\lceil \frac{2n-j}{2} \rceil, \ldots, n-1]$ as follows:

$$a(w) = s_{(2n,\lambda,\lambda_m)} = (10^{\frac{\lambda}{\lambda_m}-1})^{\lambda_m} 10^{2n-\lambda-j-1} s.$$

Let $s \in \Gamma_j$ and consider correlation $t := 0^{n-j}s$ as fixed. Recall that $g(t)$ is the cardinality of the set $G(t)$ of all strings of length $2n$ whose autocorrelation has t as suffix. To state our second formula for the population size of t, we assume that all autocorrelations of Γ_{2n} have been calculated. For consistency, we use the notation $\rho_{(.)}$ to refer to the number of free characters of an autocorrelation (as in Theorem 20).

▶ **Theorem 21.** *Let $n \in \mathbb{N}$. Let $\rho_{(2n,\lambda,\lambda_m)}$ denote the number of free characters of $s_{(2n,\lambda,\lambda_m)}$ and $\rho_{(j)}$ be the number of free characters of s_{2n}. The population size $pop(t)$ satisfies:*

$$pop(t) = \sum_{\lambda=\lceil \frac{2n-j}{2} \rceil}^{n-1} \sum_{\lambda_m} (\sigma^{\rho_{(2n,\lambda,\lambda_m)}} - \sum_{v \in \Gamma_{2n} : s_{(2n,\lambda,\lambda_m)} \subset v} pop(v)) + \sigma^{\rho_{(j)}} - \sum_{y \in \Gamma_{2n} : s_{2n} \subset y} pop(y).$$

Proof. By Theorem 19 and Theorem 17 we know:

$$pop(t) = g(t) = \sum_{\lambda=\lceil \frac{2n-j}{2} \rceil}^{n-1} pop(s_{(2n,\lambda)}) + pop(s_{2n}) \tag{1}$$

$$= \sum_{\lambda=\lceil \frac{2n-j}{2} \rceil}^{n-1} \sum_{\lambda_m} pop(s_{(2n,\lambda,\lambda_m)}) + pop(s_{2n}). \tag{2}$$

From Theorem 20 we get:

$$pop(s_{(2n,\lambda,\lambda_m)}) = \sigma^{\rho_{(2n,\lambda,\lambda_m)}} - \sum_{v \in \Gamma_{2n} : s_{(2n,\lambda,\lambda_m)} \subset v} pop(v) \tag{3}$$

and

$$pop(s_{2n}) = \sigma^{\rho_{(j)}} - \sum_{y \in \Gamma_{2n} : s_{2n} \subset y} pop(y). \tag{4}$$

Combining Equations (2), (3), and (4), we obtain the desired result. ◀

4 Asymptotics on the population ratios

The population ratio of a correlation $t \in \Delta_n$ is $pop(t)/\sigma^{2n}$. Here, we study the asymptotic lower and upper bounds for this ratio. Before stating our result, we give several definitions introduced by Guibas & Odlyzko [8]. Recall that Theorem 18 on the population size of an autocorrelation s_n relies on a sequence $\psi[k]$. They define three generating functions (with dummy variable z) two for $pop(s_n)$ and $\psi[k]$, and introduce $\widetilde{pop}(z)$, which is the normalization of $pop(z)$ by $pop(s)$. Their definitions are as follows:

$$pop(z) = \sum_{n=0}^{\infty} pop(s_n) z^{-n}; \quad \psi(z) = \sum_{n=0}^{\infty} \psi[k] z^{-n}; \quad \widetilde{pop}(z) = \frac{pop(z)}{pop(s)}.$$

Thus Theorem 18 can be rewritten as:

$$\widetilde{pop}(z) + \psi(z)\widetilde{pop}(z^2) = 2\psi(z)z^{-2j}. \tag{5}$$

Hence, the asymptotics of $pop(s_n)$ as $n \to \infty$ with s being fixed follows.

▶ **Theorem 22** (Asymptotics on the population sizes (Theorem 7.2 [8])). *Let μ be any small positive complex number. The population size of s_n divided by the population size of s over an alphabet of cardinality $\sigma \geq 2$ satisfies*

$$\frac{pop(s_n)}{pop(s)} = \left(\frac{2}{\sigma^{2j}} - \widetilde{pop}(\sigma^2)\right)\sigma^n + O((\sigma+\mu)^{\frac{n}{2}}),$$

where $\widetilde{pop}(\sigma^2)$ satisfies the Functional Equation (5).

Denote $c = \frac{2}{\sigma^{2j}} - \widetilde{pop}(\sigma^2)$. Note that c is the asymptotic limit of $pop(s_n)/(pop(s)\sigma^n)$; thus $c \cdot pop(s)$ provides the limiting value of $pop(s_n)/\sigma^n$. Here, we state our result on the population size of correlation $t \in \Delta_n$ with s being assumed fixed. See Table 3 for some interesting cases on the limiting values of $pop(s_n)/\sigma^n$ and asymptotic bounds on $pop(t)/\sigma^{2n}$.

▶ **Theorem 23** (Asymptotics on the population ratios). *Let μ be any small positive complex number. Let $t := 0^{n-j}s \in \Delta_n$ with $j \in [0, \ldots, n-1]$. Over an alphabet of cardinality $\sigma \geq 2$, the ratio $pop(t)/pop(s)$ satisfies the asymptotic inequality:*

$$c \cdot \sigma^{2n} + O((\sigma+\mu)^n) \leq \frac{pop(t)}{pop(s)} < \frac{c \cdot \sigma}{\sigma - 1} \cdot \sigma^{2n} + O(n(\sigma+\mu)^n). \tag{6}$$

In particular, we have the asymptotic bounds on the population ratio $pop(t)/\sigma^{2n}$

$$c \cdot pop(s) \leq \lim_{n\to\infty} \frac{pop(t)}{\sigma^{2n}} < \frac{c \cdot \sigma}{\sigma - 1} \cdot pop(s). \tag{7}$$

Proof. By Theorem 19 on the population size of t, for $\lambda \in [\lceil \frac{2n-j}{2} \rceil, \ldots, n-1] \cup \{0\}$, we have.

$$\frac{pop(t)}{pop(s)} = \frac{\sum_\lambda pop(s_{(2n-\lambda)}) \cdot s[j + 2\lambda - 2n] + pop(s_{2n})}{pop(s)}. \tag{8}$$

Then (8) could be bounded above and below by:

$$\frac{pop(s_{2n})}{pop(s)} \leq \frac{\sum_\lambda pop(s_{(2n-\lambda)}) \cdot s[j + 2\lambda - 2n] + pop(s_{2n})}{pop(s)} < \frac{\sum_{\lambda=0}^{n} pop(s_{(2n-\lambda)})}{pop(s)}. \tag{9}$$

From Theorem 22, for any $\lambda \in [0, \ldots, n]$ we have:

$$\frac{pop(s_{(2n-\lambda)})}{pop(s)} = c \cdot \sigma^{2n-\lambda} + O((\sigma+\mu)^{\frac{2n-\lambda}{2}}). \tag{10}$$

Plugging in (10) in the left hand side of (9) we get

$$\frac{pop(s_{2n})}{pop(s)} = c \cdot \sigma^{2n} + O((\sigma+\mu)^n)$$

and in the right hand side of (9) we obtain:

$$\begin{aligned}\sum_{\lambda=0}^{n} \frac{pop(s_{(2n-\lambda)})}{pop(s)} &= c \cdot \sum_{i=n}^{2n} \sigma^i + \left(O((\sigma+\mu)^n) + O((\sigma+\mu)^{\frac{2n-1}{2}}) + \cdots + O((\sigma+\mu)^{\frac{n}{2}})\right) \\ &= \frac{c\sigma}{\sigma-1} \cdot \sigma^{2n} + O(n(\sigma+\mu)^n)\end{aligned}$$

Combining both equations, we obtain (6):

$$c \cdot \sigma^{2n} + O((\sigma+\mu)^n) \leq \frac{pop(t)}{pop(s)} < \frac{c \cdot \sigma}{\sigma-1} \cdot \sigma^{2n} + O(n(\sigma+\mu)^n).$$

Multiplying (6) by $pop(s)/\sigma^{2n}$, we get the desired bounds (7) on the asymptotic behavior of the population ratio $pop(t)/\sigma^{2n}$. ◀

Table 3 For $\sigma = 2, 3$ and 24, we give the limiting values of $pop(s_n)/\sigma^n$ (column 3) and asymptotic bounds on $pop(t)/\sigma^{2n}$ (column 4) for some autocorrelations s. The limiting values of $pop(s_n)/\sigma^n$ in column 3 are taken from [8]. Note that the lower bound in column 4 coincides with the value in column 3 (for a given s and σ). The correlations ε and $0^{n-1}1$ are the most "popular" for $\sigma = 2$, but it is not possible to distinguish which one is the most popular. It differs from the autocorrelation case, where $10^{n-2}1$ is the most popular. For $\sigma \geq 3$, the correlation of word pairs without overlaps (i.e., 0^n) is the most popular, as in the autocorrelation case.

Alphabet Size σ	Autocorrelation s	$pop(s_n)/\sigma^n$	$pop(t)/\sigma^{2n}$
2	ε	0.268	[0.268, 0.536)
	1	0.300	[0.300, 0.600)
	10	0.110	[0.110, 0.220)
	11	0.089	[0.089, 0.178)
3	ε	0.557	[0.557, 0.836)
	1	0.283	[0.283, 0.424)
	10	0.072	[0.072, 0.108)
	11	0.032	[0.032, 0.048)
24	ε	0.957	[0.957, 0.999)
	1	0.042	[0.042, 0.044)

5 Solutions to Gabric's open questions

In the article about bordered and unbordered pairs of words [7], the author raises two challenging open questions: 1/ *How many pairs of length-n words have a longest border of fixed length j?* and 2/ *what is the expected length of the longest border of a pair of words?*. Note that with his terminology, a border is either a right-border or a left-border depending

on the order of words in the pair. As the words play symmetrical roles in the definition of border, the counts for question 1/ are equal.

For question 1/, we answer a more complex question than the one asked by Gabric: *How many pairs of length-n words have a longest border within the fixed length range* $[i..k]$. From the characterization of the set of correlations (Lemma 10), we know that correlations are partitioned by their longest border (Corollary 11). To consider pairs with longest border of length in the range $[i..k]$, we must count pairs having a correlation t in the subset $\left\{\bigcup_{j=i}^{k}(0^{n-j}.\Gamma_j)\right\}$ of Δ_n. With the recurrence that computes the population size for any correlation t (Theorem 19), it suffices to sum up $pop(t)$ over all t in this subset to answer our question, which yields Theorem 24. By shrinking the range to a single value, we exactly answer Gabric's first question, as addressed in Corollary 25.

For question 2/, we take the average over this same subset as shown below in the equation of $E(X)$ on page 15. This provides a general formula and allows us to investigate the limit of this expectation, and to show in Theorem 26 that it diverges when the string length n tends to infinity.

5.1 Counting pairs of strings with a longest border of fixed length range

▶ **Theorem 24.** *Let* $L_{[i..k]}$ *be the number of pairs of strings of length* n *that have a longest border within the fixed length range* $[i..k]$ *where* $i \leq k \in \{0,\ldots,n-1\}$. *Let* $j \in \{i,\ldots,k\}$. *Let* s *be any autocorrelation of* Γ_j. *Let* $t := 0^{n-j}s \in (0^{n-j}.\Gamma_j)$. *Let* $s_{(2n-\lambda)} = 10^{2n-\lambda-j-1}s \in \Gamma_{(2n-\lambda)}$ *where* $\lambda \in [\lceil \frac{2n-j}{2} \rceil,\ldots,n-1] \cup \{0\}$. *Then*

$$L_{[i..k]} = \sum_{t\in(\cup_{j=i}^{k}(0^{n-j}.\Gamma_j))} pop(t)$$
$$= \sum_{\lambda=\lceil\frac{2n-j}{2}\rceil}^{n-1} \sum_{s\in(\cup_{j=i}^{k}\Gamma_j)} pop(s_{(2n-\lambda)}) \cdot s[j+2\lambda-2n] + \sum_{s\in(\cup_{j=i}^{k}\Gamma_j)} pop(s_{2n}).$$

In particular, $L_{[0..k]}$ represents the number of pairs of strings of length n that have a longest border of length at most k, and $L_{[i..n-1]}$ counts the number of pairs of (distinct) strings of length n that have a longest border of length at least i. By restricting the length range to a single value j, we get the following corollary that answers Gabric's first question.

▶ **Corollary 25.** *Let* L_j *be the number of pairs of strings of length* n *that have a longest border of length* j. *Let* s *be any autocorrelation of* Γ_j. *Let* $t := 0^{n-j}s \in (0^{n-j}.\Gamma_j)$. *Let* $s_{(2n-\lambda)} = 10^{2n-\lambda-j-1}s \in \Gamma_{(2n-\lambda)}$ *where* $\lambda \in [\lceil \frac{2n-j}{2} \rceil,\ldots,n-1] \cup \{0\}$. *Then*

$$L_j = \sum_{t\in(0^{n-j}.\Gamma_j)} pop(t) = \sum_{\lambda=\lceil\frac{2n-j}{2}\rceil}^{n-1} \sum_{s\in\Gamma_j} pop(s_{(2n-\lambda)}) \cdot s[j+2\lambda-2n] + \sum_{s\in\Gamma_j} pop(s_{2n}).$$

5.2 Expected value of the longest border of a pair of words

In [7], Gabric considers a fixed alphabet size σ and a Bernoulli i.i.d model for random words. In this model, the probability that a character occurs at any position is independent of other positions and equals $1/\sigma$. For a fixed word length n, the probability of any pair of words (u, v) both of length n is $1/\sigma^{2n}$. Gabric shows that, for the expected length of the **shortest border** of a pair of words converges to a constant. In contrast, we show that the asymptotic expected length of the **longest border** actually diverges.

Define X to be the length of the longest border of a pair of strings (u, v). Then, the expectation of X is

$$E(X) = \sum_{j=0}^{n-1} j \cdot Pr(X = j) = \sum_{j=1}^{n-1} j \cdot \frac{L_j}{\sigma^{2n}} = \sum_{j=1}^{n-1} j \cdot \frac{\sum_{t \in (0^{n-j}.\Gamma_j)} pop(t)}{\sigma^{2n}}.$$

▶ **Theorem 26.** *The asymptotic expected length of the longest border of a pair of strings (u, v) diverges.*

Proof. The asymptotic expected length of the longest border of (u, v) is:

$$E_\infty(X) = \lim_{n\to\infty} \sum_{j=1}^{n-1} j \cdot \frac{\sum_{t \in (0^{n-j}.\Gamma_j)} pop(t)}{\sigma^{2n}}.$$

We claim that $\sum_{t \in (0^{n-j}.\Gamma_j)} pop(t)/\sigma^{2n} \geq c$ when $n \to \infty$, where c is a positive constant as defined in Section 4. To see this, note that $\sum_{t \in (0^{n-j}.\Gamma_j)} pop(t)/\sigma^{2n}$ satisfies the following equation by Corollary 25.

$$\begin{aligned} \frac{\sum_{t \in (0^{n-j}.\Gamma_j)} pop(t)}{\sigma^{2n}} &= \frac{\sum_{\lambda=\lceil \frac{2n-j}{2} \rceil}^{n-1} \sum_{s \in \Gamma_j} pop(s_{(2n-\lambda)}) \cdot s[j + 2\lambda - 2n] + \sum_{s \in \Gamma_j} pop(s_{2n})}{\sigma^{2n}} \\ &\geq \frac{\sum_{s \in \Gamma_j} pop(s_{2n})}{\sigma^{2n}}. \end{aligned} \tag{11}$$

By Theorem 22 and since $\sum_{s \in \Gamma_j} pop(s) \geq 1$, the right side of (11) asymptotically satisfies:

$$\frac{\sum_{s \in \Gamma_j} pop(s_{2n})}{\sigma^{2n}} = \frac{\left(c \cdot \sigma^{2n} + O((\sigma + \mu)^n)\right) \cdot \sum_{s \in \Gamma_j} pop(s)}{\sigma^{2n}} \geq c + O(\sigma^{-n}).$$

Therefore, we obtain:

$$\frac{\sum_{t \in (0^{n-j}.\Gamma_j)} pop(t)}{\sigma^{2n}} \geq c \text{ when } n \to \infty. \tag{12}$$

Which means that there exists a very large N such that when $n > N$, (12) is true. After substituting (12) to the asymptotic expectation formula, we conclude

$$E_\infty(X) = \lim_{n\to\infty} \sum_{j=1}^{n-1} j \cdot \frac{\sum_{t \in (0^{n-j}.\Gamma_j)} pop(t)}{\sigma^{2n}} \geq \lim_{n\to\infty} \sum_{j=N}^{n-1} c \cdot j \to \infty.$$

◀

6 Conclusion

Our work focuses on counting ordered pairs of words (u, v) that satisfy a given correlation, which is a binary vector that encodes all overlaps of u over v. The set of such pairs is called the population of the correlation, and their number the population size. The main results on population size are stated in Theorems 19 and 21. With these at hands, one can count the number of pairs of length-n words with a longest border of length j as asked by Gabric (his open question 1/) [7], or the expected length of this longest border across all pairs of length-n words (his open question 2/), since the longest border is encoded in the correlation. Thus, the answer to open question 1/ is for instance a corollary of Theorem 24, which answers a more

complex question. This emphasizes the importance of accounting for the complete overlap structure of a pair of words when investigating such questions. Another result illustrating this fact is the asymptotic divergence of the expected length of the longest border (Gabric's open question 2/ – see Theorem 26). This result contrasts with the convergence of the longest border of single words of length n [14].

We conclude our work by proposing one conjecture and one open question:

1. We conjecture that population ratio $pop(t)/\sigma^{2n}$ converges, and its asymptotic behavior equals the limiting value of $pop(s_n)/\sigma^{2n}$: $\lim_{n\to\infty} pop(t)/\sigma^{2n} = \lim_{n\to\infty} pop(s_n)/\sigma^{n}$.
2. What is the variance or distribution of the length of the longest border of a pair of words?

References

1 Dragana Bajic and Tatjana Loncar-Turukalo. A simple suboptimal construction of cross-bifix-free codes. *Cryptography and Communications*, 6(6):27–37, 2014. doi:10.1007/s12095-013-0088-8.

2 Elena Barcucci, Antonio Bernini, and Renzo Pinzani. A strong non-overlapping dyck code. In Nelma Moreira and Rogério Reis, editors, *Developments in Language Theory*, pages 43–53, Cham, 2021. Springer International Publishing.

3 Stefano Bilotta, Elisa Pergola, and Renzo Pinzani. A new approach to cross-bifix-free sets. *IEEE Transactions on Information Theory*, 58(6):4058–4063, 2012. doi:10.1109/TIT.2012.2189479.

4 Simon R. Blackburn, Navid Nasr Esfahani, Donald L. Kreher, and Douglas R. Stinson. Constructions and bounds for codes with restricted overlaps. *IEEE Transactions on Information Theory*, 70(4):2479–2490, 2024. doi:10.1109/TIT.2023.3304712.

5 Isa Cakir, Ourania Chryssaphinou, and Marianne Månsson. On A conjecture by eriksson concerning overlap in strings. *Comb. Probab. Comput.*, 8(5):429–440, 1999. URL: http://journals.cambridge.org/action/displayAbstract?aid=46697.

6 Andrzej Ehrenfeucht and D.M. Silberger. Periodicity and unbordered segments of words. *Discrete Mathematics*, 26(2):101–109, 1979. URL: http://dx.doi.org/10.1016/0012-365X(79)90116-X, doi:10.1016/0012-365x(79)90116-x.

7 Daniel Gabric. Mutual borders and overlaps. *IEEE Transactions on Information Theory*, 68(10):6888–6893, 2022. doi:10.1109/TIT.2022.3167935.

8 Leonidas J. Guibas and Andrew M. Odlyzko. Periods in strings. *Journal of Combinatorial Theory, Series. A*, 30:19–42, 1981. doi:10.1016/0097-3165(81)90038-8.

9 Leonidas J. Guibas and Andrew M. Odlyzko. String overlaps, pattern matching, and nontransitive games. *Journal of Combinatorial Theory, Series A*, 30(2):183–208, 1981. doi:10.1016/0097-3165(81)90005-4.

10 Dan Gusfield. *Algorithms on Strings, Trees, and Sequences - Computer Science and Computational Biology*. Cambridge University Press, 1997. URL: https://doi.org/10.1017/cbo9780511574931, doi:10.1017/CBO9780511574931.

11 Dan Gusfield, Gad M. Landau, and Baruch Schieber. An efficient algorithm for the All Pairs Suffix-Prefix Problem. *Inf Proc Letters*, 41(4):181–185, 1992. URL: http://www.sciencedirect.com/science/article/pii/002001909290176V, doi:10.1016/0020-0190(92)90176-V.

12 Vesa Halava, Tero Harju, and Lucian Ilie. Periods and binary words. *Journal of Combinatorial Theory, Series A*, 89(2):298–303, 2000. doi:10.1006/jcta.1999.3014.

13 Tero Harju and Dirk Nowotka. Periodicity and unbordered segments of words. *Bulletin EATCS*, 80:162–167, 2003.

14 Stepan Holub and Jeffrey O. Shallit. Periods and borders of random words. In Nicolas Ollinger and Heribert Vollmer, editors, *33rd Symposium on Theoretical Aspects of Computer Science, STACS 2016, February 17-20, 2016, Orléans, France*, volume 47 of *LIPIcs*, pages 44:1–44:10. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.STACS.2016.44.

15 D.E. Knuth, J.H. Morris, and V.R. Pratt. Fast pattern matching in strings. *SIAM Journal of Computing*, 6:323–350, 1977.

16 Jihyuk Lim and Kunsoo Park. A fast algorithm for the All-Pairs Suffix–Prefix problem. *Theoretical Computer Science*, 698:14–24, 2017. URL: http://www.sciencedirect.com/science/article/pii/S0304397517305510, doi:10.1016/j.tcs.2017.07.013.

17 Alexander Loptev, Gregory Kucherov, and Tatiana Starikovskaya. On maximal unbordered factors. In Ferdinando Cicalese, Ely Porat, and Ugo Vaccaro, editors, *Combinatorial Pattern Matching - 26th Annual Symposium, CPM 2015, Ischia Island, Italy, June 29 - July 1, 2015, Proceedings*, volume 9133 of *Lecture Notes in Computer Science*, pages 343–354. Springer, 2015. doi:10.1007/978-3-319-19929-0_29.

18 O. P. Lossers. Overlapping binary sequences (G. blom). *SIAM Rev.*, 37(4):619–620, 1995. doi:10.1137/1037139.

19 M. Lothaire, editor. *Algebraic combinatorics on Words*. Cambridge University Press, second edition, 1997.

20 M. Lothaire, editor. *Combinatorics on Words*. Cambridge University Press, second edition, 1997.

21 Veli Mäkinen, Djamal Belazzougui, Fabio Cunial, and Alexandru I. Tomescu. *Genome-Scale Algorithm Design: Biological Sequence Analysis in the Era of High-Throughput Sequencing*. Cambridge University Press, 2015. doi:10.1017/CBO9781139940023.

22 Peter Tolstrup Nielsen. A note on bifix-free sequences (corresp.). *IEEE Trans. Inf. Theory*, 19(5):704–706, 1973. doi:10.1109/TIT.1973.1055065.

23 Peter Tolstrup Nielsen. On the expected duration of a search for a fixed pattern in random data (corresp.). *IEEE Transactions on Information Theory*, 19(5):702–704, 1973. doi:10.1109/TIT.1973.1055064.

24 Ora E. Percus and Paula A. Whitlock. Theory and Application of Marsaglia's Monkey Test for Pseudorandom Number Generators. *ACM Transactions on Modeling and Computer Simulation*, 5(2):87–100, April 1995. doi:10.1145/210330.210331.

25 Sven Rahmann and Eric Rivals. Exact and efficient computation of the expected number of missing and common words in random texts. In Raffaele Giancarlo and David Sankoff, editors, *Combinatorial Pattern Matching, 11th Annual Symposium, CPM 2000, Montreal, Canada, June 21-23, 2000, Proceedings*, volume 1848 of *Lecture Notes in Computer Science*, pages 375–387. Springer, 2000. doi:10.1007/3-540-45123-4_31.

26 Sven Rahmann and Eric Rivals. On the distribution of the number of missing words in random texts. *Combinatorics, Probability and Computing*, 12(01), Jan 2003. doi:10.1017/s0963548302005473.

27 Eric Rivals. Incremental algorithms for computing the set of period sets. HAL, 2024. lirmm-04531880, 22 pages.

28 Eric Rivals and Sven Rahmann. Combinatorics of Periods in Strings. In F. Orejas, P. Spirakis, and J. van Leuween, editors, *ICALP 2001, Proc. of the 28th International Colloquium on Automata, Languages and Programming, (ICALP), Crete, Greece, July 8-12, 2001*, volume 2076 of *Lecture Notes in Computer Science*, pages 615–626. Springer Verlag, 2001. doi:10.1007/3-540-48224-5_51.

29 Eric Rivals and Sven Rahmann. Combinatorics of periods in strings. *Journal of Combinatorial Theory, Series A*, 104(1):95–113, 2003. doi:10.1016/s0097-3165(03)00123-7.

30 Eric Rivals, Michelle Sweering, and Pengfei Wang. Convergence of the Number of Period Sets in Strings. In Kousha Etessami, Uriel Feige, and Gabriele Puppis, editors, *50th International Colloquium on Automata, Languages, and Programming (ICALP 2023)*, volume 261 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 100:1–100:14, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL: https://drops.dagstuhl.de/opus/volltexte/2023/18152, doi:10.4230/LIPIcs.ICALP.2023.100.

31 Stéphane Robin, François Rodolphe, and Sophie Schbath. *DNA, Words and Models*. Cambrigde University Press, 2005.

32 William F. Smyth. *Computating Pattern in Strings*. Pearson - Addison Wesley, 2003.

33 William H. A. Tustumi, Simon Gog, Guilherme P. Telles, and Felipe A. Louza. An improved algorithm for the All-Pairs Suffix–Prefix problem. *J. of Discrete Algorithms*, 37:34–43, 2016. URL: http://www.sciencedirect.com/science/article/pii/S1570866716300053, doi:10.1016/j.jda.2016.04.002.

34 Niko Välimäki, Susana Ladra, and Veli Mäkinen. Approximate All-Pairs Suffix/Prefix overlaps. *Inf. Comput.*, 213:49–58, 2012. doi:10.1016/j.ic.2012.02.002.

A Structure of Δ_n

In this section, we show that Δ_n is a lattice under set inclusion, and that it does not satisfy the Jordan-Dedekind condition. The Jordan-Dedekind condition requires that all maximal chains between the same two elements have the same length. This extends to Δ_n the findings of Rivals & Rahmann [29] who proved similar results for Γ_n.

First let us now show that Δ_n is closed by intersection, for any $n > 0$.

▶ **Lemma 27.** *Let t and $t' \in \Delta_n$. Then $(t \cap t') \in \Delta_n$.*

Proof. Let $t, t' \in \Delta_n$. By Lemma 10, we can write $t = \mathtt{0}^{n-i}s_i, t' = \mathtt{0}^{n-j}s_j, s_i \in \Gamma_i, s_j \in \Gamma_j, i, j \in [0, \ldots, n-1]$. We claim that if $(s_i \cap s_j) \in \Gamma_{\min(i,j)}$, then $(t \cap t') \in \Delta_n$. We distinguish two cases: If $\mathbf{i = j}$, then $(s_i \cap s_j) \in \Gamma_j$ by Lemma 3.3 from [29]. Thus $\mathtt{0}^{n-i}(s_i \cap s_j) \in \left(\mathtt{0}^{n-j}.\Gamma_j\right) \subset \Delta_n$.

Otherwise, $\mathbf{i \neq j}$, and without loss of generality, we suppose $i < j$. Let string $U \in \Sigma^i$, and string $V \in \Sigma^j$ such that $a(U) = s_i, a(V) = s_j$. Denote $V = V_1V_2$ where $|V_1| = i, |V_2| = j - i$. Let $W = (\Sigma \times \Sigma)^i$ such that $W[k] = (U[k], V[k]), k \in [0, i-1]$. It follows that $a(W) \in \Gamma_i$ (by Lemma 3.3 [29]). Note that $a(V) = a(V_1) \cup 0^i a(V_2)$. Then we have $(s_i \cap s_j) = a(U) \cap a(V) = a(U) \cap \left(a(V_1) \cup 0^i a(V_2)\right) = (a(U) \cap a(V_1) \cup (a(U) \cap 0^i a(V_2)) = a(W) \cup \emptyset = a(W) \in \Gamma_i$. Therefore $\mathtt{0}^{n-i}(s_i \cap s_j) \in \left(\mathtt{0}^{n-i}.\Gamma_i\right) \subset \Delta_n$. ◀

▶ **Theorem 28.** *$(\Delta_n, \subset)$ is a lattice.*

Proof. Note that $(\Delta_n, \subset)$ has null element $\mathtt{0}^n$, and universal element $\mathtt{1}^n$. By Lemma 27, Δ_n is closed under intersection. The meet $x \wedge y$ of x, y is their intersection, the join $x \vee y$ of x, y is the intersection of all elements containing x, y. The universal element ensures this intersection is not empty. ◀

By Lemma 10, we have Γ_n is strictly included in Δ_n. As any autocorrelation has its leftmost bit equal to $\mathtt{1}$, and only the autocorrelations have this property in Δ_n, it follows that only an autocorrelation can be a successor of an autocorrelation. Moreover, $\mathtt{10}^{n-1}$ is a successor of the null element $\mathtt{0}^n$. It follows that, between the null and universal element of Δ_n, there is a chain of length strictly smaller than n that goes through a chain between $\mathtt{10}^{n-1}$ and the universal element $\mathtt{1}^n$ and traverses only nodes that are autocorrelations, by Lemma 3.5 from [29] when $n > 6$. More exactly this chain has length $\lfloor n/2 \rfloor + 1$.

For any $n > 2$, the following chain $\mathtt{0}^n \prec \mathtt{0}^{n-1}\mathtt{1} \prec \ldots \prec \mathtt{0}^{n-i}\mathtt{1}^i$ (with i in $2, \ldots, n-2$) to $\mathtt{01}^{n-1}$, and finally to the universal element $\mathtt{1}^n$ exists in Δ_n. This chain is maximal has length n – which is the maximal length of a chain in Δ_n. Since there exist two maximal chains of different length between the null and universal elements of Δ_n, when $n > 6$, and visual inspection of Δ_4 and Δ_5 confirms the same property, we obtain this Theorem.

▶ **Theorem 29.** *For $n > 3$, the lattice Δ_n does not satisfy the Jordan-Dedekind condition.*

B Proof of Lemma 14

Proof. We prove by construction. $POP_r(t)$ is the set of strings v who have length n, and a prefix whose autocorrelation is s.

Denote $v = v_1v_2$, where $|v_1| = j, |v_2| = n - j$. Clearly, s is the autocorrelation of v_1. The population size $pop_r(t)$ equals all possible choices of v_1 times all possible choices of v_2. Note that all possible choices of v_1 is the population size of s, $pop(s)$, whereas v_2 can be arbitrary which implies all possible choices of v_2 is σ^{n-j}. Indeed, once a string v is given, we can

construct a corresponding string u as following: Denote $u = u_1v_1$ where $|u_1| = n - j$. We construct $u_1 = u[0, n-j-1]$ by choosing $u[i] \in \Sigma\backslash\{v[0]\}$, $i \in [0, n-j-1]$ meaning each letter in $u[0, n-j-1]$ differs from the first letter of v. It ensures that there is no overlap for (u, v) before the position $n - j$. ◀

C Population size of a correlation: recurrence II

▶ **Theorem 30.** *Let $k, \lambda, j, n \in \mathbb{N}$ satisfying $0 \leq \lambda, j < n$. Let $s \in \Gamma_j$ be a fixed element. Define $t := 0^{n-j}s$ to be an element of Δ_n. Then, $pop(t)$, the population size of t satisfies the recurrence*

$$\begin{aligned} pop(t) = &\sum_{\lambda=\lceil \frac{2n-j}{2} \rceil}^{n-1} 2pop(s)\psi[2j+\lambda-2n]\ s[j+2\lambda-2n] \\ &- \sum_{\lambda=\lceil \frac{2n-j}{2} \rceil}^{n-1} \Big(\sum_{k} pop(s_k)\psi[2k-2n+\lambda]\Big)\ s[j+2\lambda-2n] + pop(s_{2n}), \end{aligned}$$

where $pop(s_k) = 0$ for $k < j$, and ψ is defined as above.

Proof. We just substitute the recurrence on $s_{(2n-\lambda)}$ by Theorem 18 to Theorem 19

$$pop(s_{(2n-\lambda)}) = 2pop(s)\psi[2j+\lambda-2n] - \sum_{k} pop(s_k)\psi[2k-2n+\lambda].$$

◀