Two-dimensional pattern matching with k mismatches

Jonas Ellert[1], Paweł Gawrychowski[2], Adam Górkiewicz[3], and Tatiana Starikovskaya[4]

[1]?
[2]?
[3]?
[4]?

Abstract

[TODO]

1 Related Work

Hamming-Threshold PM in 1D

- Early solutions for pattern matching with mismatches use fast Fourier transform and run in $\mathcal{O}(n\sqrt{m \log m})$ time [1]. The first algorithms that explicitly exploit the threshold k run in time roughly $\mathcal{O}(nk)$ [39, 52] and use “kangaroo jumping”, a technique in which each text position is considered as a potential occurrence, and verified by jumping from mismatch to mismatch using a data structure for longest common extensions.

- Faster solution in $\mathcal{O}(n\sqrt{k \log k})$ or $\tilde{\mathcal{O}}(n + k^3 n/m)$ time [12] by still using convolutions, but also exploiting periodicity and using counting arguments. The latter result was first improved to $\tilde{\mathcal{O}}(n + k^2 n/m)$ time [29], and then further refined to $\tilde{\mathcal{O}}(n + kn/\sqrt{m})$ time [45], which is a smooth trade-off between the previous $\tilde{\mathcal{O}}(n\sqrt{k})$ and $\tilde{\mathcal{O}}(n + k^2 n/m)$ bounds. If Monte-Carlo randomization is allowed, then the time can be slightly improved to $\mathcal{O}(n + kn\sqrt{(\log m)/m})$ [24]. A significantly faster algorithm would imply unexpected consequences for the complexity of boolean matrix multiplication [45].

- The k-mismatch occurrences of the pattern either have a simple and exploitable structure, or the pattern is close to being periodic. This property is a powerful tool for pattern matching with mismatches, and was analyzed in [23, 25].

Exact PM in 2D Assuming $n \times n$ text and $m \times m$ pattern, $m \leq n$.

- Two-dimensional pattern matching dates back (at least) to the 1970s, when Richard Bird generalized the Knuth-Morris-Pratt algorithm [50] for one-dimensional pattern matching, achieving $\mathcal{O}(n^2 + m^2)$ time [21].

- By using multiple pattern matching on only $\mathcal{O}(n/m)$ rows of the text, Baeza-Yates and Régnier achieve average time complexity $\mathcal{O}(n^2/m)$, using $\mathcal{O}(m^2)$ space and still running in $\mathcal{O}(n^2)$ time in the worst case [18]. This was followed by more results optimizing the average time complexity [46, 62].

- Crochremore at al. [35] solve two-dimensional pattern matching space efficiently, using an $\mathcal{O}(m^2)$ time and $\mathcal{O}(\log m)$ space preprocessing followed by an $\mathcal{O}(n^2)$ time and constant space search phase.

- On a PRAM, the problem can be solved in constant time and $O(n^2)$ work [34], after preprocessing the pattern in $\mathcal{O}(\log\log m)$ time and $\mathcal{O}(m^2)$ work [31].
- Other works studying the problem are, e.g., [5, 19, 40, 49, 63, 64]

Hamming-Threshold PM in 2D Assuming $n \times n$ text and $m \times m$ pattern, $m \leq n$, and threshold $k < m^2$.

- First result in 1987 by Krithivasan and Sitalakshmi [51], who show that $\tilde{\mathcal{O}}(kmn^2)$ time can be achieved.
- Faster algorithm by Rank and Heywood runs in $\tilde{\mathcal{O}}((k + m)n^2)$ time [60].
- Current state of the art is $\mathcal{O}(kn^2)$ [8] **J:** check
- More work [17, 46, 58] is aimed at optimizing the average time complexity. This has also been done for the setting in which arbitrary rotations of the pattern are allowed [38].
- Dynamic and online algorithms [30]
- Measures like edit distance are more complicated to define for two-dimensional strings [16].

2D Periodicity

- Two-dimensional periodicity was introduced in [3, 4] and refined in [40]. All periods (and witnesses for all non-periods) can be computed in $\mathcal{O}(n^2)$ time [32]. Witnesses can be used to efficiently eliminate potential occurrences of the pattern. Since witness computation is costly for run-length encoded strings, Amir at al. [6] show additional properties of two-dimensional periodicities that allow the elimination of occurrences without witnesses.
- Famous periodicity theorems have been generalized to two dimensions, for example the Fine-Wilf theorem [37, 56] and the Lyndon-Schützenberger theorem [42, 53].
- Smallest rectangular cover in $\mathcal{O}(n^2)$ time, all aperiodic covers in $\mathcal{O}(n^2 \log n)$ time [27]. Covers of two-dimensional strings were also considered in [36, 41].
- All tile covers in $\mathcal{O}(n^{2+\epsilon})$ time [59].
- There can be $\Omega(n^3 \log n)$ tandem substrings (repeated rectangle, like square in a normal string) [13].
- All tandems can be computed in $\mathcal{O}(n^3 \log n)$ time [14].
- There can be $\Omega(n^2 \log n)$ two-dimensional runs (maximal repetitions) [43].
- There are at most $\mathcal{O}(n^2 \log^2 n)$ two-dimensional runs [26].
- All runs can be computed in $\mathcal{O}(n^2 \log^2 n)$ time [9].

Other 2D Shannanigans

- A survey by Rytter [61] discusses compressed representations of two-dimensional strings and the computation on such representations without explicit decompression. If the text is compressed as a two-dimensional SLP, then merely deciding if a pattern is present is NP-complete with respect to the compressed size [20]. If both text and pattern are compressed using a two-dimensional version of Lempel-Ziv compression, then exact matching can be done in $\mathcal{O}(n^2 + m^3)$ time, using additional working space proportional to the compressed representation of the pattern [10]. If text and pattern are given in two-dimensional run-length encoding, then the time for exact matching is linear in the size of the compressed text, and the space is linear in the size of the compressed pattern [11]. If multiple patterns are given, then they can be compressed into a space efficient self-index that allows fast dictionary matching [57].

- In parametrized matching, pattern occurrences have to be isomorphic via a bijection on the effective alphabets [2,33]. A key step for efficient parametrized matching in two dimensions is counting the number of distinct symbols in every $m \times m$ substring, which can be done in $\mathcal{O}(n^2)$ time [33].
- More work on structural elements in two-dimensional strings, e.g., frames (rectangles with matching outer columns and rows) [22], Lyndon words [55], motif patterns (allowing "don't cares", and satisfying special properties) [15], patterns with at least two occurrences [48], and palindromic substrings [54].
- There is an efficient algorithm for matching a half-rectangular pattern (fixed height, but irregular shape) in a text, allowing a fixed number of mismatches, insertions and deletions [7].

2 Introduction

[TODO]

We consider the one-dimensional all-substring Hamming distance problem (HD1D), where for a given text string T of length n and a string P of length m ($m < n$), we want to calculate the Hamming distance between P and every fragment T of length m.

We consider the two-dimensional all-substring Hamming distance problem (HD2D), where for a given 2D string T of size $n \times n$ and a string P of size $m \times m$ ($m < n$), we want to calculate the Hamming distance between P and every $m \times m$ fragment of T.

We also consider the bounded variants of HD1D and HD2D, where we are only required to calculate the distances which are not greater than k, for some parameter $k \in \mathbb{Z}^+$.

Theorem 5. *Bounded* HD2D *can be solved in* $\tilde{\mathcal{O}}((m^2 + mk^{5/4})n^2/m^2)$ *time.*

3 Preliminaries

For our purposes we will not use the standard definition of a two-dimensional string, where we associate it with a two-dimensional array of characters, and instead we will define it more broadly. Although we will occasionally use the array notation, we will do it exclusively for $n \times m$ strings. For any $n \in \mathbb{Z}^+$ we will denote $[n] = \{0, \ldots, n-1\}$. We will use the terms point and vector interchangeably. Our results hold under word-RAM model of computation.

Definition 1 (Two-dimensional string)**.** We define a **string** S as a partial function $\mathbb{Z}^2 \nrightarrow \Sigma$ which maps some arbitrary set of integer points, denoted as $\mathrm{dom}(S)$, to characters. For simplicity we will write $u \in S$ to denote that $u \in \mathrm{dom}(S)$. We say that a string S is **partitioned** into strings $R_1, \ldots, R_\ell$ when the sets $\mathrm{dom}(R_1), \ldots, \mathrm{dom}(R_\ell)$ partition $\mathrm{dom}(S)$ and $R_i(u) = S(u)$ for all $u \in R_i$. We call a string S **monochromatic** when $S(u) = \sigma$ for every $u \in S$ for some $\sigma \in \Sigma$ and we will write $\mathrm{C}(S)$ to denote the value σ. We say that S is $n \times m$ for some $n, m \in \mathbb{Z}^+$ when $\mathrm{dom}(S) = [n] \times [m]$. Physically we represent a string as a list of point-character pairs.

Definition 2 (Shifting)**.** For a set of points $V \subseteq \mathbb{Z}^2$ and a vector $u \in \mathbb{Z}^2$, we denote $V + u$ as $\{v + u : v \in V\}$. For a string S and a vector $u \in \mathbb{Z}^2$ we denote $S + u$ as a string R such that $\mathrm{dom}(R) = \mathrm{dom}(S) + u$ and $R(v) = S(v - u)$ for $v \in \mathrm{dom}(R)$. Intuitively, we shift the set of points while maintaining their character values.

Definition 3 (Hamming distance)**.** For a pair of strings S, R we define

$$\mathrm{Ham}(S,R) = |\,\{\, u : u \in \mathrm{dom}(S) \cap \mathrm{dom}(R), S(u) \neq R(u) \,\}\,|,$$

which corresponds to the number of mismatches between S and R.

Under such notation, the HD2D problem is equivalent to calculating the (bounded or unbounded) values of $\mathrm{Ham}(P+q,T)$ for all $q \in \mathbb{Z}^2$ such that $\mathrm{dom}(P+q) \subseteq \mathrm{dom}(T)$ (so for $q \in [n-m+1]^2$).

Definition 4 (Don't care symbol)**.** We define the **don't care** symbol as a special character which matches with every character. We will denote it with ?. Unless stated otherwise, we assume it is not allowed in Σ and in both HD1D and HD2D every character present in T and P matches only with itself.

Definition 5 (Vector operators)**.** For any $u \in \mathbb{Z}^2$ we refer to its coordinates as $u.x, u.y$. For $u, v \in \mathbb{Z}^2$ we denote $u \cdot v = u.x \cdot v.x + u.y \cdot v.y$ and $u \times v = u.x \cdot v.y - u.y \cdot v.x$. Note that alternatively $u \cdot v = |u||v| \cos\alpha$ and $u \times v = |u||v| \sin\alpha$ where α is the angle between u and v.

4 One-dimensional generalizations

In this section we explore some of the methods used for one-dimensional strings. Specifically, as our goal is to generalize the solution for pattern matching with k mismatches described in [44], we are especially interested in two-dimensional variants of the techniques that were used to solve the one-dimensional case.

Theorem 1. *Consider an algorithm $\mathcal{A}$ which solves* HD2D *(bounded or unbounded), but only when $2|n$ and $n \leq \frac{3}{2}m$. If its running time is $\mathcal{T}(m)$, then the general case can be solved in $\mathcal{O}(\mathcal{T}(m)n^2/m^2)$.*

Proof. Let $r = \lfloor m/2 \rfloor$ and let $n' = r+m-1$ or $r+m$ if $r+m-1$ is odd. We see that the set $N = [n']^2$ satisfies the conditions for the text domain. For any vector $q \in [n-m]^2$ we can find a vector u such that $r|u.x, r|u.y$ and $q - u \in [r]^2$, so we have $\mathrm{Ham}(P+q,T) = \mathrm{Ham}(P+q-u, T_u)$ where T_u is the restriction of $T-u$ to N. If $T-u$ is not defined for some $v \in N$, we can pad $T_u(v)$ with any character. We see that $\mathrm{dom}(P+q-u) \subseteq N = \mathrm{dom}(T_u)$. There are $\mathcal{O}(n^2/m^2)$ possible vectors u and we run $\mathcal{A}$ for every pair of T_u and P. $\square$

Theorem 2. *Consider an $n \times n$ string T, $m \times m$ string P and set of vectors Q such that $\mathrm{dom}(P+q) \subseteq \mathrm{dom}(T)$ for every $q \in Q$. There exists an algorithm which calculates $d_q = \mathrm{Ham}(P+q,T)$ for every $q \in Q$ in total time $\tilde{\mathcal{O}}(n^2 + \sum_{q \in Q} d_q)$.*

Proof. For the sake of clarity, we will temporarily switch to the classical array notation for strings. Let $T_0, \ldots, T_{n-m}$ denote an array of two-dimensional strings (arrays) such that $T_k[0\,..\,n-1, 0\,..\,m-1] = T[0\,..\,n-1, k\,..\,k+m-1]$. For every row $P[0], \ldots, P[m-1]$ of P and every row $T_k[0], \ldots, T_k[n-1]$ of every T_k we assign an integer identifier so that $\mathrm{Id}(P[i]) = \mathrm{Id}(T_k[j]) \Leftrightarrow P[i] = T_k[j]$ by using the KMR algorithm (described in [48]) in $\tilde{\mathcal{O}}(n^2)$.

We use the approach described in [39]. There exists a data structure (suffix array) which for a given one-dimensional array S allows us to detect all mismatches between any given two of its subarrays of equal length. It can be built in $\tilde{\mathcal{O}}(|S|)$ and the query time is $\tilde{\mathcal{O}}(d+1)$ where d is the number of mismatches. We construct the suffix array for the concatenation of the following arrays:

- the rows $P[i]$ for every i,
- the rows $T[i]$ for every i,

- the array $\mathrm{Id}(P[0])\,\mathrm{Id}(P[1])\ldots\mathrm{Id}(P[m-1])$,
- the arrays $\mathrm{Id}(T_k[0])\,\mathrm{Id}(T_k[1])\ldots\mathrm{Id}(T_k[n-1])$ for every k,

the total length of which is $\mathcal{O}(n^2)$. Let us consider a problem of detecting mismatches between P and some $T' = T[j\mathinner{.\,.}j+m-1, k\mathinner{.\,.}k+m-1]$. We can first find all row indices i for which $P[i] \neq T'[i]$ by finding all mismatches between $\mathrm{Id}(P[0])\ldots\mathrm{Id}(P[m-1])$ and $\mathrm{Id}(T_k[j])\ldots\mathrm{Id}(T_k[j+m-1])$, which we do with query to the data structure. For every such i we can then find all mismatches between $P[i]$ and $T'[i]$ by querying $P[i]$ and $T[i+j][k\mathinner{.\,.}k+m-1]$. If the distance between P and T' is d, the first query takes $\tilde{\mathcal{O}}(d+1)$ operations and all subsequent queries take $\tilde{\mathcal{O}}(d+1)$ operations in total. □

Lemma 1. HD1D *with don't care symbols can be solved in $\tilde{\mathcal{O}}(n|\Sigma|)$ by running $|\Sigma|$ instances of FFT.*

Lemma 2 ([47])**.** *There exists a $(1+\varepsilon)$-approximate algorithm which solves* HD1D *with don't care symbols in $\tilde{\mathcal{O}}(n)$.*

Theorem 3. HD2D *with don't care symbols can be solved in $\tilde{\mathcal{O}}(n^2|\Sigma|)$.*

Proof. We will again use the array notation. We construct one-dimensional strings $\bar{T}$ and $\bar{P}$ by concatenating subsequent rows $T[0], \ldots, T[n-1]$ of T and rows $P[0], \ldots, P[m-1]$ of P padded with don't care symbols:

$$\bar{T} = T[0]\ T[1]\ \ldots\ T[n-1],$$

$$\bar{P} = P[0]\ ?^{n-m}\ P[1]\ ?^{n-m}\ \ldots\ ?^{n-m}\ P[m-1].$$

We run the algorithm from Lemma 1. The distance between $T[i\mathinner{.\,.}i+m-1, j\mathinner{.\,.}j+m-1]$ and P is equal to the distance between $\bar{T}[in+j\mathinner{.\,.}in+j+nm-n+m-1]$ and $\bar{P}$. □

Theorem 4. *There exists a $(1+\varepsilon)$-approximate algorithm which solves* HD2D *with don't care symbols in $\tilde{\mathcal{O}}(n^2)$.*

Proof. Identical to Theorem 3, but we use the algorithm from Lemma 2 instead of Lemma 1. □

The same reduction as in Theorem 3 can be applied for every HD1D solution which allows don't care symbols. Unfortunately, the most effective known algorithms for bounded HD1D [28,44] rely on periodicity and inherently do not allow don't care symbols, thus, they cannot be easily generalized.

Observation 1. *Every* HD2D *solution which allows don't care symbols (eg. the algorithms from Theorem 3 and Theorem 4) can be extended to also calculate the Hamming distance for occurrences of P which are not entirely contained in T. It can be done by padding the text with don't care symbols and it does not change the complexity of the solution.*

5 Main result

In this section we provide a detailed proof of the following theorem:

Theorem 5. *Bounded* HD2D *can be solved in $\tilde{\mathcal{O}}((m^2+mk^{5/4})n^2/m^2)$ time.*

We show an algorithm which works in time $\tilde{\mathcal{O}}(m^2+mk^{5/4})$, assuming $2|n$ and $m < n \leq \frac{3}{2}m$. The solution for the general case follows from Theorem 1.

We start by running the algorithm from Theorem 4 with $\varepsilon = 1$. We construct the set Q as the set of such vectors $q \in \mathbb{Z}^2$ for which the estimated value of $\mathrm{Ham}(P+q, T)$ is at most $2k$. For every $q \in [n-m+1]^2 \setminus Q$ we say that $\mathrm{Ham}(P+q, T)$ equals ∞. The next step is to calculate the exact value of $\mathrm{Ham}(P+q, T)$ for every $q \in Q$.

Let us consider the case when $|Q| \leq 6m + m^2/k$. We can run the algorithm from Theorem 2 and by the fact that $\mathrm{Ham}(P+q, T) \leq 4k$ for every $q \in Q$, it will perform $\tilde{\mathcal{O}}(m^2 + mk)$ operations. We are left with the case when $|Q| > 6m + m^2/k$, in which we take advantage of the fact that some strings $P + q$ for $q \in Q$ must have a large overlap and small Hamming distance from each other, and thus P must be periodic.

5.1 Two-dimensional periodicity

In this section we introduce a range of new tools related to two-dimensional periodicity. We then select some special periods of the pattern and show how to decompose it into some regularly structured monochromatic strings.

Definition 6 (Periodicity)**.** Consider any vector $\delta \in \mathbb{Z}^2$. We say that a string S has an ℓ-period δ when

$$\mathrm{Ham}(S + \delta, S) \leq \ell.$$

Lemma 3. *For every $u, v \in Q$, the vector $u - v$ is an $8k$-period of P.*

Proof. $\mathrm{Ham}(P+u-v, P) = \mathrm{Ham}(P+u, P+v) \leq \mathrm{Ham}(P+u, T) + \mathrm{Ham}(P+v, T) \leq 4k+4k$. □

Theorem 9. *For a given $\ell \in \mathbb{Z}^+$ and a set of points $U \subseteq [\ell+1]^2$, such that $|U| > 12\ell$, there exist $s, t, s', t' \in U$, such that the following conditions hold for $w = t - s$ and $w' = t' - s'$:*

- $0 < |w||w'| = \mathcal{O}(\ell^2/|U|)$,
- $|\sin\alpha| \geq \frac{1}{2}$ *where α is the angle between w and w',*
- *$w, w', -w, -w'$ are all contained in different **quadrants**, defined as*

$$\begin{aligned}
\mathcal{Q}_1 &= (0, +\infty) \times [0, +\infty),\\
\mathcal{Q}_2 &= (-\infty, 0] \times (0, +\infty),\\
\mathcal{Q}_3 &= (-\infty, 0) \times (-\infty, 0],\\
\mathcal{Q}_4 &= [0, +\infty) \times (-\infty, 0).
\end{aligned}$$

Such w, w' can be found in $\tilde{\mathcal{O}}(|U|)$ operations.

Proof. See Section 5.4. □

We run the algorithm from Theorem 9 on the set Q (where $\ell = n - m \leq m/2$, thus $|Q| > 6m + m^2/k \geq 12\ell$). We obtain vectors $\varphi \in \mathcal{Q}_4$ and $\psi \in \mathcal{Q}_1$, which by Lemma 3 are $\mathcal{O}(k)$-periods of P. We will refer to those vectors throughout the rest of the description. Note that because $|Q| > 6m + m^2/k$, we have $0 \leq \varphi \times \psi \leq |\varphi||\psi| = \mathcal{O}(\min\{m, k\})$.

Definition 7 (Lattice congruency)**.** We define $\mathcal{L} = \{ s\varphi + t\psi : s, t \in \mathbb{Z} \}$. We say that two vectors $u, v \in \mathbb{Z}^2$ are **lattice-congruent** and denote $u \equiv v$ when $u - v \in \mathcal{L}$.

Lemma 4. *There exists a set of points $\Gamma \subseteq \mathbb{Z}^2$ such that $|\Gamma| = \mathcal{O}(\min\{m, k\})$ and every point $u \in \mathbb{Z}^2$ is lattice-congruent to exactly one point $\gamma \in \Gamma$.*

Proof. Let $p = \{ s\varphi + t\psi : s \in [0,1), t \in [0,1) \}$. We construct $\Gamma = p \cap \mathbb{Z}^2$. It is commonly known, that a simple polygon with integer vertices contains $\mathcal{O}(A)$ integer points in the interior or on the boundary, where A denotes its surface area. Observe that the points in Γ are contained in a parallelogram with vertices $(0,0), \varphi, \varphi + \psi, \psi$. Since its surface area is $\varphi \times \psi = \mathcal{O}(\min\{m, k\})$, we get $|\Gamma| = \mathcal{O}(\min\{m, k\})$.

Now consider any point $u \in \mathbb{Z}^2$. There exist some unique values $s, t \in [0,1)$ and $s', t' \in \mathbb{Z}$, such that $u = (s+s')\varphi + (t+t')\psi$. It is easy to see that $u \equiv s\varphi + t\psi$ and $s\varphi + t\psi \in \Gamma$. □

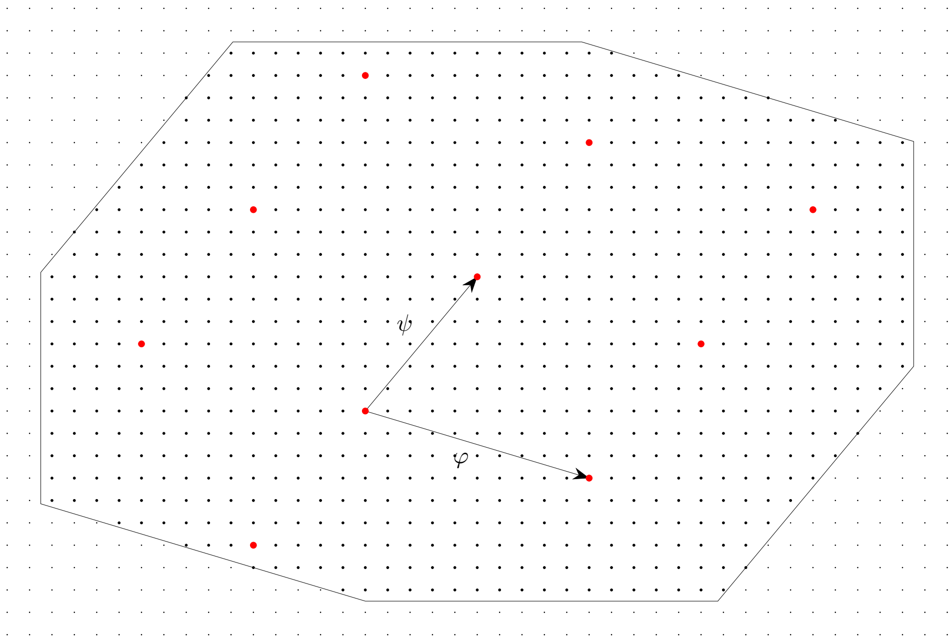

Figure 1: All the points in the polygon form a parquet and the red points form a subparquet.

Definition 8 (Parquet)**.** We call a set $U \subseteq \mathbb{Z}^2$ a **parquet** when there exist some values $x_0, x_1, y_0, y_1, \varphi_0, \varphi_1, \psi_0, \psi_1 \in \mathbb{Z}$, which we will call its **signature**, such that

$$U = [x_0, x_1] \times [y_0, y_1] \cap \{\, u : u \in \mathbb{Z}^2, \varphi \times u \in [\varphi_0, \varphi_1], \psi \times u \in [\psi_0, \psi_1] \,\}\,.$$

See Figure 1 for an illustration.

a) If additionally $x_1 - x_0 + 1 \geq |\varphi.x| + |\psi.x|$ and $y_1 - y_0 + 1 \geq |\varphi.y| + |\psi.y|$, then U is a **spacious** parquet.

b) If additionally $x_0, y_0 = -\infty$ and $x_1, y_1 = +\infty$, then U is a **simple** parquet.

Note that every simple parquet is spacious.

Definition 9 (Subparquet)**.** We call a set $V \subseteq \mathbb{Z}^2$ a **subparquet** when there exists a parquet U and a point $\gamma \in \mathbb{Z}^2$ such that

$$V = \{\, u : u \in U, u \equiv \gamma \,\}\,.$$

This is also illustrated in Figure 1. A signature of V consists of a signature of U and the vector γ. We call V a spacious/simple subparquet when there exists U which is (correspondingly) a spacious/simple parquet. We say that V is lattice-congruent to some $v \in \mathbb{Z}^2$ (denoted as $V \equiv v$) when $v \equiv \gamma$. We similarly define the lattice congruency between two subparquets.

Definition 10 (Parquet string)**.** We call a string S a spacious/simple (sub-)parquet string when $\mathrm{dom}(S)$ is a spacious/simple (sub-)parquet.

Theorem 12. *A given spacious/simple parquet string R with $\mathcal{O}(k)$-periods φ and ψ can be partitioned in time $\tilde{\mathcal{O}}(|\mathrm{dom}(R)| + k)$ into $\mathcal{O}(k)$ monochromatic spacious/simple subparquet strings, correspondingly.*

Proof. See Section 5.6. ∎

Since $|\varphi.x|, |\varphi.y|, |\psi.x|, |\psi.y| \leq n-m \leq m/2$, the $m \times m$ string P is a spacious parquet string and satisfies the assumptions of Theorem 12. We partition P into a set of strings $\mathcal{V}$. We then group the strings based on the single character they contain. Specifically, for every character $\sigma \in \Sigma$ present in P, we construct the set $\mathcal{V}_\sigma = \{\, V : V \in \mathcal{V}, \mathrm{C}(V) = \sigma \,\}$.

Theorem 11. *For a given set of monochromatic simple subparquet strings $\mathcal{S}$ we can calculate*

$$\sum_{S \in \mathcal{S}} \mathrm{Ham}(P + q, S)$$

for every $q \in Q$ in total time $\tilde{\mathcal{O}}(m^2 + \sum_{S \in \mathcal{S}} |\mathcal{V}_{\mathrm{C}(S)}|)$, assuming that the sets $\mathrm{dom}(S)$ for $S \in \mathcal{S}$ are some pairwise disjoint subsets of $\mathrm{dom}(T)$.

Proof. See Section 5.5. □

5.2 Text decomposition

Because the text is not necessarily periodic, we unfortunately cannot use the same approach as for the pattern. In this section we show how to decompose T using a similar, but more nuanced method.

Definition 11 (Active text)**.** We define the **active text** $T_{\mathbf{a}}$ as the restriction of T to

$$\bigcup_{q \in Q} \mathrm{dom}(P + q).$$

Observation 2. $\mathrm{Ham}(P + q, T) = \mathrm{Ham}(P + q, T_{\mathbf{a}})$ *for every $q \in Q$.*

Definition 12 (Peripherality)**.** For every point $u \in \mathbb{Z}^2$ we define its **border distance** as $\min\{\, |u - v| : v \in \mathbb{Z}^2 \setminus \mathrm{dom}(T_{\mathbf{a}}) \,\}$. We say that a set of points $U \subseteq \mathbb{Z}^2$ is d-**peripheral** for some $d \geq 0$, if the border distance of every $u \in U$ is not greater than d. We say that a string S is d-peripheral when $\mathrm{dom}(S)$ is d-peripheral.

Theorem 13. *For a given positive integer $\ell \leq m$, we can partition the active text in time $\tilde{\mathcal{O}}(m^2 + \ell k)$ into a set of $\mathcal{O}(\ell k)$ monochromatic simple subparquet strings and an $\mathcal{O}(m/\ell)$-peripheral string.*

Proof. See Section 5.7. □

We partition $T_{\mathbf{a}}$ using the algorithm from Theorem 13 with $\ell = mk^{-3/4}$ into a set of simple subparquet strings $\mathcal{S}$ and a $\mathcal{O}(k^{3/4})$-peripheral string F. For every $q \in Q$ we then have

$$\mathrm{Ham}(P + q, T_{\mathbf{a}}) = \mathrm{Ham}(P + q, F) + \sum_{S \in \mathcal{S}} \mathrm{Ham}(P + q, S).$$

By Theorem 11, we can calculate $\sum_{S \in \mathcal{S}} \mathrm{Ham}(P+q, S)$ for every $q \in Q$ in time $\tilde{\mathcal{O}}(m^2 + mk^{5/4})$, since $\sum_{S \in \mathcal{S}} |\mathcal{V}_{\mathrm{C}(S)}| \leq |\mathcal{S}||\mathcal{V}| = \mathcal{O}(mk^{5/4})$. In the next section (5.3) we will introduce Theorem 6, which states that for a d-peripheral string F, we can calculate $\mathrm{Ham}(P + q, F)$ for every $q \in Q$ in total time $\tilde{\mathcal{O}}(m^2 + mdk^{1/2})$. By substituting $d = \mathcal{O}(k^{3/4})$, we get the complexity of $\tilde{\mathcal{O}}(m^2 + mk^{5/4})$, which ends the main proof.

5.3 Text periphery

In this section we explore the properties of peripheral strings. We consider any $d > 0$ and a non-empty d-peripheral string S, such that $\text{dom}(S) \subseteq \text{dom}(T_{\mathbf{a}})$. We define a partitioning of S into strings $S_1, \ldots, S_4$, by splitting it through the middle with a horizontal and vertical line. Specifically

- S_1 is the restriction of S to $\{ n/2, \ldots, n-1 \} \times \{ n/2, \ldots, n-1 \}$ (upper right quarter),
- S_2 is the restriction of S to $\{ 0, \ldots, n/2-1 \} \times \{ n/2, \ldots, n-1 \}$ (upper left quarter),
- S_3 is the restriction of S to $\{ 0, \ldots, n/2-1 \} \times \{ 0, \ldots, n/2-1 \}$ (lower left quarter),
- S_4 is the restriction of S to $\{ n/2, \ldots, n-1 \} \times \{ 0, \ldots, n/2-1 \}$ (lower right quarter).

We will demonstrate some characteristics of S_1, and by symmetry, generalize them to S.

Lemma 5. *Assuming $d \leq m/4$, there does not exist $u \in S_1$ and $v \in T_a$ such that $v.x - u.x \geq d$ and $v.y - u.y \geq d$.*

Proof. Assume the contrary. Since $u \in S_1$, the border distance of u is at most d, so there exists $w \in \mathbb{Z}^2 \setminus \text{dom}(T_{\mathbf{a}})$, such that $u.x - d \leq w.x \leq u.x + d$ and $u.y - d \leq w.y \leq u.y + d$. Since $v \in T_a$, there exists $q \in Q$ such that $v \in [m]^2 + q$. We have

$$w.x \geq u.x - d \geq n/2 - m/4 \geq n - m > q.x$$

and

$$w.x \leq u.x + d \leq v.x \leq q.x + m - 1.$$

Similarly we can show that $q.y \leq w.y \leq q.y + m - 1$, and thus $w \in [m]^2 + q$. Since $[m]^2 + q \subseteq \text{dom}(T_{\mathbf{a}})$ and $w \notin T_{\mathbf{a}}$, we get a contradiction. $\square$

We now introduce two major theorems regarding peripheral strings, the first of which is proven in the next section (5.3.1):

Theorem 7. *We can calculate* $\text{Ham}(P + q, S)$ *for every $q \in Q$ in total time $\tilde{\mathcal{O}}(m^2 + md|\Sigma|)$, where $|\Sigma|$ is the number of different characters present in both P and S.*

Theorem 6. *We can calculate* $\text{Ham}(P + q, S)$ *for every $q \in Q$ in total time $\tilde{\mathcal{O}}(m^2 + mdk^{1/2})$.*

Proof. Recall the construction of the sets $\mathcal{V}_\sigma$ described in Section 5.1. We define $\sigma \in \Sigma$ to be a **frequent** character if $|\mathcal{V}_\sigma| \geq \sqrt{k}$ and if $|\mathcal{V}_\sigma| < \sqrt{k}$, we call it an **infrequent** character. We partition S into two strings F and I, based on character frequency, so that F consists of only the frequent characters and I consists of only the infrequent ones. For every $q \in Q$ we then have

$$\text{Ham}(P + q, S) = \text{Ham}(P + q, F) + \text{Ham}(P + q, I).$$

Observe that the number of different frequent characters is $\mathcal{O}(\sqrt{k})$, and thus, by Theorem 7, we can calculate $\text{Ham}(P + q, F)$ for every $q \in Q$ in total time $\tilde{\mathcal{O}}(m^2 + mdk^{1/2})$, since F is d-peripheral.

We partition I into $|\text{dom}(I)|$ strings, one per every $u \in I$. Specifically, let I_u be the restriction of I to $\{ u \}$ for every $u \in I$. We have $\text{Ham}(P + q, I) = \sum_{u \in I} \text{Ham}(P + q, I_u)$ for every $q \in Q$. By Definition 9, I_u are simple subparquet strings, and thus, we can by Theorem 11 calculate the results in $\tilde{\mathcal{O}}(m^2 + \sum_{u \in I} |\mathcal{V}_{I(u)}|)$. Since $I(u)$ is an infrequent character for every $u \in I$, we have $|\mathcal{V}_{I(u)}| < k^{1/2}$ for every $u \in I$. By Observation 4 we have $|\text{dom}(I)| = \mathcal{O}(md)$, and thus the total complexity is $\tilde{\mathcal{O}}(m^2 + mdk^{1/2})$. $\square$

5.3.1 Peripheral convolution

This section serves as the proof of the theorem we just used to prove Theorem 6:

Theorem 7. *We can calculate* $\mathrm{Ham}(P+q,S)$ *for every* $q \in Q$ *in total time* $\tilde{\mathcal{O}}(m^2 + md|\Sigma|)$*, where* $|\Sigma|$ *is the number of different characters present in both* P *and* S*.*

We base our approach on the simple method of calculating the Hamming distance by running an instance of FFT for each unique character. We will again utilize partitioning to reduce the problem to some smaller ones and then solve them naively. We will take advantage of the fact that the points close to the border can overlap only with a small subset of points from the pattern when considering the occurrences fully contained in the active text.

Recall that $\mathrm{Ham}(P+q,S) = \mathrm{Ham}(P+q,S_1) + \cdots + \mathrm{Ham}(P+q,S_4)$. We will only show how to calculate $\mathrm{Ham}(P+q,S_1)$ for every $q \in Q$, since the other cases are symmetric. Consider a string P_0, defined as the restriction of P to $[m-d]^2$ and a string P_1, defined as the restriction of P to $\mathrm{dom}(P) \setminus \mathrm{dom}(P_0)$. Since the strings P_0 and P_1 partition P, we have

$$\mathrm{Ham}(P+q,S_1) = \mathrm{Ham}(P_0+q,S_1) + \mathrm{Ham}(P_1+q,S_1).$$

Definition 13. (width & height) For a non-empty set $U \subseteq \mathbb{Z}^2$ we define its **width** as $\max\{\, u.x - v.x + 1 : u,v \in U \,\}$ and its **height** as $\max\{\, u.y - v.y + 1 : u,v \in U \,\}$. For a non-empty string R we define the width and height as the width and height of $\mathrm{dom}(R)$.

Theorem 8. *Given two non-empty strings* P *and* T *of widths* w_P, w_T *and heights* h_P, h_T*, we can calculate* $\mathrm{Ham}(P+q,T)$ *for every* $q \in \mathbb{Z}^2$*, for which the result is non-zero, in total time* $\tilde{\mathcal{O}}((|\Sigma|+1)(w_P+w_T)(h_P+h_T))$*, where* $|\Sigma|$ *denotes the number of different characters present in both* P *and* T*.*

Proof. We can prove it by slightly generalizing Theorem 3, although following the same method, and utilizing Observation 1. □

From now we will assume that $d \le m/4$, since for $d > m/4$ we can, by Theorem 8, calculate the results in time $\tilde{\mathcal{O}}(m^2 + m^2|\Sigma|)$, which is sufficient.

Lemma 6. $\mathrm{dom}(P_0+q) \cap \mathrm{dom}(S_1) = \emptyset$ *for every* $q \in Q$*.*

Proof. Let us assume the contrary. Select any $q \in Q$ such that $\mathrm{dom}(P_0+q) \cap \mathrm{dom}(S_1)$ contains some point u and consider the point $v = (u.x+d, u.y+d)$. Since $u \in [m-d]^2 + q$, we have $v \in [m]^2 + q \subseteq \mathrm{dom}(T_{\mathbf{a}})$, thus the points $u \in S_1$ and $v \in T_{\mathbf{a}}$ contradict Lemma 5. □

Observation 3. P_1 *can be partitioned into two strings* P_2 *and* P_3 *such that the width of* P_2 *and the height of* P_3 *are equal to* d*.*

By Lemma 6, $\mathrm{Ham}(P_0+q,S_1) = 0$ for every $q \in Q$ and by Observation 3 we have

$$\mathrm{Ham}(P+q,S_1) = \mathrm{Ham}(P_1+q,S_1) = \mathrm{Ham}(P_2+q,S_1) + \mathrm{Ham}(P_3+q,S_1)$$

for some strings P_2 and P_3 partitioning P_1, such that the width of P_2 and the height of P_3 are equal to d. We calculate $\mathrm{Ham}(P_2+q,S_1)$ and $\mathrm{Ham}(P_3+q,S_1)$ for every Q independently and sum the results. We only show how to calculate $\mathrm{Ham}(P_2+q,S_1)$, since the other case is symmetric.

We will now partition S_1. Consider an array of strings $U_0,\ldots,U_{\lceil n/d\rceil-1}$, where U_i is the restriction of S_1 to $\{\, id,\ldots,id+d-1 \,\} \times [n] \cap \mathrm{dom}(S_1)$. For the sake of formality (since the maximum/minimum of an empty set is undefined), let $V_0,\ldots,V_{\ell-1}$ consist of all non-empty strings U_i, given in the increasing order of i. Observe that $V_0,\ldots,V_{\ell-1}$ partition S_1 and their width is not greater than d.

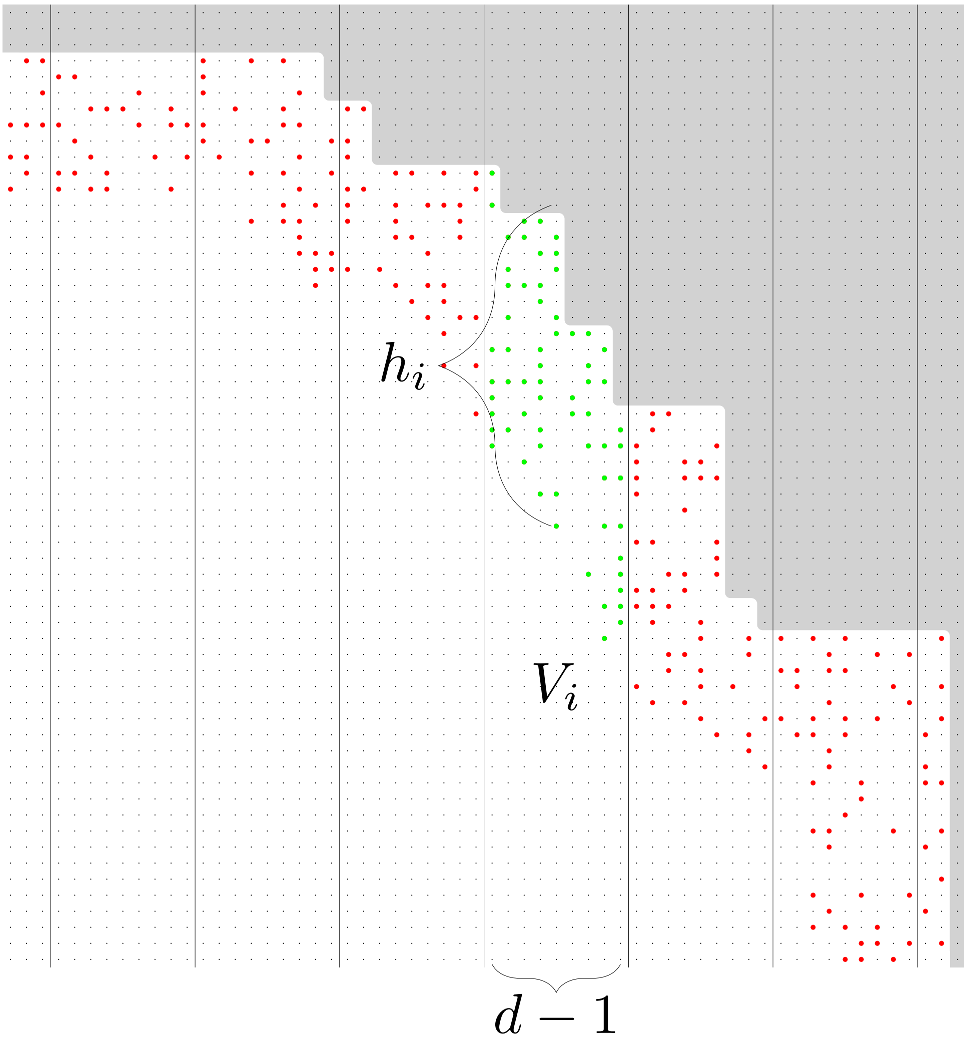

Figure 2: The decomposition of S_1.

For each $i \in [\ell]$ we find $h_i \in \mathbb{Z}^+$, which we define as the minimal number such that $(u.x, u.y + h_i) \notin T_{\mathbf{a}}$ for every $u \in V_i$.

The construction is illustrated in Figure 2. The points in the gray area are outside of the active text. The remaining ones are in the active text, where the red and green represent $\mathrm{dom}(S_1)$, and the green belong to some fixed V_i.

Lemma 7. *The sum of all h_i is $\mathcal{O}(m)$.*

Proof. Since for $\ell < 2$, the proof is trivial, we assume $\ell \geq 2$. For every i (since h_i is minimal) there exists a point $u_i \in V_i$, such that $(u_i.x, u_i.y + h_i - 1) \in T_{\mathbf{a}}$. It can be shown that for all $i \geq 2$ we have

$$h_i \leq u_{i-2}.y - u_i.y + d,$$

since if that was not the case for some i, then the points u_{i-2} and $v = (u_i.x, u_i.y + h_i - 1)$ would contradict Lemma 5. We can conclude that

$$\sum_{i=0}^{\ell-1} h_i \leq h_0+h_1+\sum_{i=2}^{\ell-1}(u_{i-2}.y-u_i.y+d) = h_0+h_1+u_0.y+u_1.y-u_{\ell-2}.y-u_{\ell-1}.y+(\ell-2)d = \mathcal{O}(m).$$

$\square$

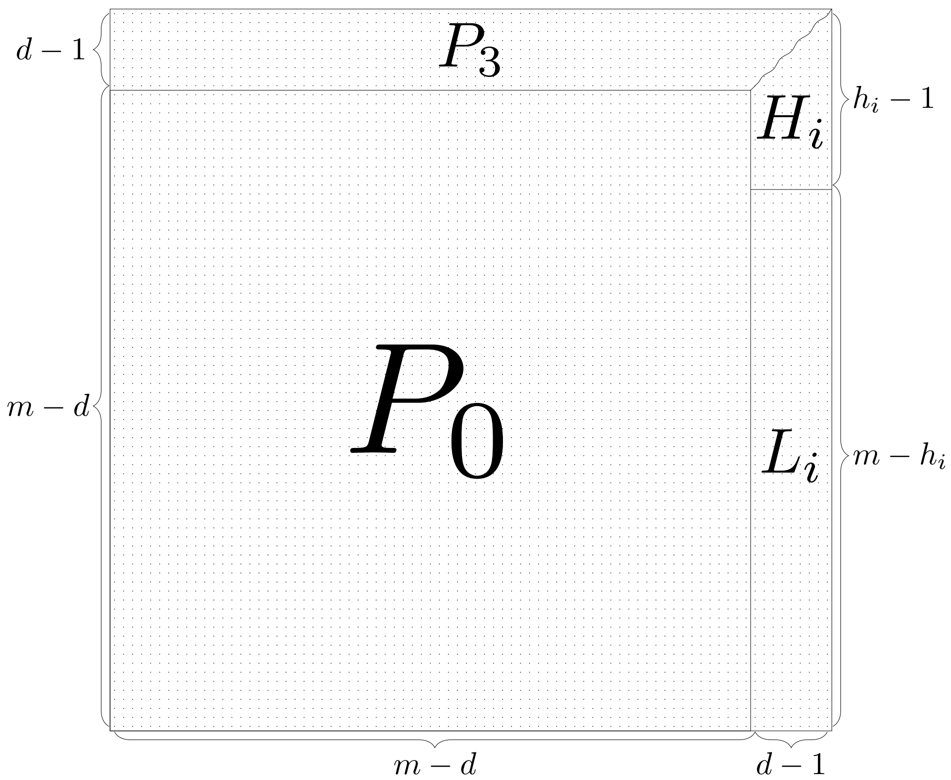

Figure 3: Pattern partitioning.

Observation 4. *For every i, the height of V_i is not greater than h_i. By Lemma 7 we have $|\operatorname{dom}(S_1)| = \mathcal{O}(md)$ and by extension $|\operatorname{dom}(S)| = \mathcal{O}(md)$.*

For every $i \in [l]$ we construct the string L_i as the restriction of P_2 to $[m] \times [m-h_i] \cap \operatorname{dom}(P_2)$ and the string H_i as the restriction of P_2 to $\operatorname{dom}(P_2) \setminus \operatorname{dom}(L_i)$. The construction is illustrated in Figure 3. Since L_i and H_i partition V_i, we have

$$\operatorname{Ham}(P_2+q, S_1) = \sum_{i=0}^{\ell-1} \operatorname{Ham}(P_2+q, V_i) = \sum_{i=0}^{\ell-1} \operatorname{Ham}(L_i+q, V_i) + \sum_{i=0}^{\ell-1} \operatorname{Ham}(H_i+q, V_i).$$

Lemma 8. $\operatorname{dom}(L_i+q) \cap \operatorname{dom}(V_i) = \emptyset$ *for every $q \in Q$ and $i \in [\ell]$.*

Proof. Let us assume the contrary. Select any $q \in Q$ and $i \in [\ell]$, such that $\operatorname{dom}(L_i+q) \cap \operatorname{dom}(V_i)$ contains some point u and consider the point $v = (u.x, u.y + h_i)$. Since $u \in [m] \times [m-h_i] + q$, we have $v \in [m]^2 + q \subseteq \operatorname{dom}(T_{\mathbf{a}})$, thus $v \in T_{\mathbf{a}}$, which contradicts the definition of h_i. $\square$

By Lemma 8, for every $q \in Q$ we have $\sum_{i=0}^{\ell-1} \operatorname{Ham}(L_i+q, V_i) = 0$, thus our result is equal to $\sum_{i=0}^{\ell-1} \operatorname{Ham}(H_i+q, V_i)$. We run the algorithm from Theorem 8 for every pair of H_i and V_i and, since both H_i and V_i have widths not greater than d and heights not greater than h_i, we obtain the total complexity of $\tilde{\mathcal{O}}(\sum_{i=0}^{\ell-1}(|\Sigma|+1)dh_i)$, which, by Lemma 7, is $\tilde{\mathcal{O}}(m^2 + md|\Sigma|)$.

5.4 Period acquisition

This section serves as the proof of the theorem, which we used to obtain the periods φ and ψ:

Theorem 9. *For a given $\ell \in \mathbb{Z}^+$ and a set of points $U \subseteq [\ell+1]^2$, such that $|U| > 12\ell$, there exist $s, t, s', t' \in U$, such that the following conditions hold for $w = t - s$ and $w' = t' - s'$:*

- $0 < |w||w'| = \mathcal{O}(\ell^2/|U|)$,
- $|\sin\alpha| \geq \frac{1}{2}$ *where* α *is the angle between* w *and* w',
- $w, w', -w, -w'$ *are all contained in different* ***quadrants****, defined as*

$$\begin{aligned}
\mathcal{Q}_1 &= (0, +\infty) \times [0, +\infty),\\
\mathcal{Q}_2 &= (-\infty, 0] \times (0, +\infty),\\
\mathcal{Q}_3 &= (-\infty, 0) \times (-\infty, 0],\\
\mathcal{Q}_4 &= [0, +\infty) \times (-\infty, 0).
\end{aligned}$$

Such w, w' *can be found in* $\tilde{\mathcal{O}}(|U|)$ *operations.*

We start by finding the closest pair of points in U. Specifically, we select any pair of different points $s, t \in U$, which minimizes $|t - s|$. Such pair can be obtained in $\tilde{\mathcal{O}}(|U|)$ operations, for example with a sweep line method. We construct $w = t - s$.

We define a partial order $\leq_w$ on $\mathbb{Z}^2$, where we have $u \leq_w u$ for every $u \in \mathbb{Z}^2$ and $v \leq_w u$ for some pair of different points $u, v \in \mathbb{Z}^2$, when at least one condition holds for $\delta = u - v$:

a) w and δ belong to the same quadrant,

b) $\alpha \in (-\pi/6, \pi/6)$, where α is the angle between w and δ.

Consider a vector ρ, where

1° if $w \in \mathcal{Q}_1$, then $\rho = (+\sqrt{2}/2, +\sqrt{2}/2)$,

2° if $w \in \mathcal{Q}_2$, then $\rho = (-\sqrt{2}/2, +\sqrt{2}/2)$,

3° if $w \in \mathcal{Q}_3$, then $\rho = (-\sqrt{2}/2, -\sqrt{2}/2)$,

4° if $w \in \mathcal{Q}_4$, then $\rho = (+\sqrt{2}/2, -\sqrt{2}/2)$.

Observe that the condition (a) is equivalent to

a') $\alpha \in [-\pi/4, \pi/4)$, where α is the angle between ρ and δ.

Let β be the angle between k and w. Similarly, the condition (b) is equivalent to

b') $\alpha \in (\beta - \pi/6, \beta + \pi/6)$, where α is the angle between ρ and δ.

Let $r = [-\pi/4, \pi/4) \cup (\beta - \pi/6, \beta + \pi/6)$. We can see that the conditions (a) and (b) are thus equivalent to a single condition:

A) $\alpha \in r$, where α is the angle between ρ and δ.

Observe that $r \subseteq (-5\pi/12, 5\pi/12)$. Thus, the vectors δ, which hold (A), belong to a single half-plane and they satisfy $\delta \cdot \rho > \cos(5\pi/12)|\delta||\rho| > |\delta|/4$. Also, since r is a continuous range of angles, for every δ_1 and δ_2 satisfying the condition, $\delta_1 + \delta_2$ also satisfies it. Thus, we can prove that for every $u_1, u_2, u_3 \in \mathbb{Z}^2$, such that $u_1 \leq_w u_2$ and $u_2 \leq_w u_3$, we have $u_1 \leq_w u_3$ (meaning the relation is transitive). If $u_1 = u_2$ or $u_2 = u_3$, the proof is trivial. If not, observe that $\delta_1 = u_2 - u_1$ and $\delta_2 = u_3 - u_1$ hold the condition, thus it also holds for $u_3 - u_1 = \delta_1 + \delta_2$. It is also easy to prove that $\leq_w$ is acyclic.

Under the partial order $\leq_w$, we find the longest chain C and the longest antichain A using dynamic programming in $\tilde{\mathcal{O}}(|U|)$ operations.

Lemma 9. $(|C| - 1)|w| < 6\ell$.

Proof. Let $f = |C| - 1$ and let $c_0, \ldots, c_f$ denote the consecutive points in C, such that we have $c_i \leq_w c_{i+1}$ for every $i \in [f]$. Consider the array $\delta_0, \ldots, \delta_{f-1}$, where $\delta_i = c_{i+1} - c_i$ for every $i \in [f]$. By definition of w, we have $|\delta_i| \geq |w|$, and since $\delta_i \cdot \rho > |\delta_i|/4$, we get $\delta_i \cdot \rho > |w|/4$ for every $i \in [f]$. We have

$$\sum_{i=0}^{f-1} \delta_i = c_f - c_0,$$

and thus

$$f|w|/4 < \sum_{i=0}^{f-1} \delta_i \cdot \rho = (c_f - c_0) \cdot \rho \leq \ell\sqrt{2},$$

which gives us $(|C| - 1)|w| < 4\ell\sqrt{2} < 6\ell$. □

Lemma 10. $|U| \leq |C||A|$.

Proof. It follows from Dilworth's theorem. □

We know that $|C| \geq 2$, since there exists a chain containing s and t. By Lemma 9, we have

$$|C||w|/2 \leq (|C| - 1)|w| < 6\ell,$$

and thus

$$|C| \leq |C||w| < 12\ell.$$

By the assumption $|U| \geq 12\ell$ and Lemma 10

$$12\ell < |U| \leq |C||A| < 12\ell|A|,$$

thus $|A| > 1$, which means $|A| \geq 2$. We select any pair of different vectors $s', t' \in A$, which minimizes $|t' - s'|$ and construct $w' = t' - s'$. We will now show that $|w||w'| = \mathcal{O}(\ell^2/|U|)$.

Lemma 11. $(|A| - 1)|w'| \leq 2\ell$.

Proof. Recall that $(-\pi/4, \pi/4) \subseteq r$. Define a range of angles $r' = [-\pi/4, 3\pi/4]$. Consider any $u, v \in \mathbb{Z}^2$, such that $u \not\leq_w v$ and $v \not\leq_w u$. It can be shown that the angle between ρ and δ is in r' for some $\delta \in \{u - v, v - u\}$. Thus $|(u - v) \times \rho| \geq \sin(\pi/4)|u - v||\rho| = |u - v|\sqrt{2}/2$.

Let $f = |A| - 1$ and let $a_0, \ldots, a_f$ be the points in A ordered such that $a_i \times \rho \leq a_{i+1} \times \rho$ for every $i \in [f]$. Consider the array $\delta_0, \ldots, \delta_{f-1}$, where $\delta_i = a_{i+1} - a_i$ for every $i \in [f]$. By definition of w', we have $|\delta_i| \leq |w'|$ and since $|\delta_i \times \rho| \geq |\delta_i|\sqrt{2}/2$, we get $\delta_i \times \rho = |\delta_i \times \rho| \geq |w|\sqrt{2}/2$ for every $i \in [f]$. We have

$$\sum_{i=0}^{f-1} \delta_i = a_f - a_0,$$

and thus

$$f|w'|\sqrt{2}/2 \leq \sum_{i=0}^{f-1} \delta_i \times \rho = (c_f - c_0) \times \rho \leq \ell\sqrt{2},$$

which gives us $(|A| - 1)|w| \leq 2\ell$. □

By Lemma 11, and since $|A| \geq 2$, we have

$$|A||w'| \leq 2(|A| - 1)|w'| \leq 4\ell.$$

Recall that $|C||w| < 12\ell$ and $|U| \leq |C||A|$. We can multiply the inequalities and obtain

$$|U||w||w'| \leq |C||A||w||w'| < 48\ell^2,$$

which finally gives us

$$|w||w'| < \frac{48\ell^2}{|U|} = \mathcal{O}(\ell^2/|U|).$$

It can be easily shown that w, w' hold the remaining conditions by the definition of $\leq_w$.

5.5 Subparquet convolution

Throughout this section we will denote $D = \{ u : u \in \mathcal{L}, \varphi \times u \geq 0, \psi \times u \geq 0 \}$, where $\mathcal{L}$ is the set defined in Definition 7. We start by introducing some auxiliary tools, which we later use in the proof of Theorem 11.

Lemma 12. *Given a set of subparquets $\mathcal{V}$ and a set of points Q, we can calculate*

$$\sum_{V \in \mathcal{V}} |(D + q) \cap V|$$

for every $q \in Q$ in total time $\tilde{\mathcal{O}}(n^2 + |Q| + |\mathcal{V}|)$, assuming that every $V \in \mathcal{V}$ consists of vectors of length $\mathcal{O}(n)$.

Proof. For every $u \in \mathbb{Z}^2$ let us define $\mathrm{score}(u) = |\{ V : V \in \mathcal{V}, u \in V \}|$. Observe that

$$\sum_{V \in \mathcal{V}} |(D + q) \cap V| = \sum_{u \in D + q} \mathrm{score}(u).$$

We start by explicitly calculating the scores. We find the maximum length of a vector that some $V \in \mathcal{V}$ is defined for, which we denote ℓ. We construct the set $U \subseteq \mathbb{Z}^2$ of all vectors of length at most ℓ. By the assumption, we have $\ell = \mathcal{O}(n)$, and thus $|U| = \mathcal{O}(l^2) = \mathcal{O}(n^2)$. We observe that since all the scores are zero for points outside of U, we can only calculate them for $u \in U$.

We find the set Γ introduced in Lemma 4 and for every $\gamma \in \Gamma$ we construct $U_\gamma = U \cap (\mathcal{L} + \gamma)$. Consider any $u \in U_\gamma$ for some fixed $\gamma \in \Gamma$ and any $V \in \mathcal{V}$. We observe that if $V \not\equiv \gamma$, then $u \notin V$ and thus V does not contribute to $\mathrm{score}(u)$. If $V \equiv \gamma$, then we can find a parquet W such that $V = W \cap (\mathcal{L} + \gamma)$ and we have $u \in V \Leftrightarrow u \in W \cap (\mathcal{L} + \gamma) \Leftrightarrow u \in W$. Thus, if we denote $\mathcal{W}_\gamma$ as the set of parquets W obtained for every $V \in \mathcal{V}$ such that $V \equiv \gamma$, then $\mathrm{score}(u)$ for $u \in U_\gamma$ is the number of parquets $W \in \mathcal{W}_\gamma$ such that $u \in W$. We calculate $\mathrm{score}(u)$ for every $u \in U_\gamma$ by sweeping U_γ and $\mathcal{W}_\gamma$ in time $\tilde{\mathcal{O}}(|U_\gamma| + |\mathcal{W}_\gamma|)$. We do it independently for every $\gamma \in \Gamma$, performing $\tilde{\mathcal{O}}(|U| + |\mathcal{V}|) = \tilde{\mathcal{O}}(n^2 + |\mathcal{V}|)$ operations in total.

Now consider a query vector $q \in Q$. Let $\gamma \in \Gamma$ be such that $q \equiv \gamma$. We have already shown that the sum of scores for $u \in D + q$ is equal to the sum of scores for $u \in (D + q) \cap U$. Since $(D + q) \cap U = (D + q) \cap U_\gamma$, we see that the result is the sum of scores for such $u \in U_\gamma$, for which $\varphi \times u \geq \varphi \times q$ and $\psi \times u \geq \psi \times q$. If we denote $Q_\gamma = Q \cap (\mathcal{L} + \gamma)$, we see that we can calculate the results for all $q \in Q_\gamma$ by sweeping Q_γ and U_γ in time $\tilde{\mathcal{O}}(|Q_\gamma| + |U_\gamma|)$. We do it independently for every $\gamma \in \Gamma$, performing $\tilde{\mathcal{O}}(|Q| + |U|) = \tilde{\mathcal{O}}(n^2 + |Q|)$ operations in total. □

Lemma 13. *For any simple subparquet U we can find $w_0, \ldots, w_3 \in \mathbb{Z}^2$, such that*

$$|U \cap X| = \sum_{j=0}^{3} (-1)^j |(D + w_j) \cap X|$$

for every $X \subseteq \mathbb{Z}^2$. If U consists of vectors of length $\mathcal{O}(n)$, then $w_0, \ldots, w_3$ are of length $\mathcal{O}(n)$.

Proof. Let

$$\varphi_0 = \min \{ \varphi \times u : u \in U \}, \quad \varphi_1 = \max \{ \varphi \times u : u \in U \},$$
$$\psi_0 = \min \{ \psi \times u : u \in U \}, \quad \psi_1 = \max \{ \psi \times u : u \in U \}.$$

Note that these values can be extracted from the signature. Since U is a parquet, there exist unique points $u_0, \ldots, u_3 \in U$, such that

- $\varphi \times u_0 = \varphi_0$ and $\psi \times u_0 = \psi_0$,

- $\varphi \times u_1 = \varphi_1$ and $\psi \times u_1 = \psi_0$,
- $\varphi \times u_2 = \varphi_1$ and $\psi \times u_2 = \psi_1$,
- $\varphi \times u_3 = \varphi_0$ and $\psi \times u_3 = \psi_1$.

We construct

$$w_0 = u_0, \quad w_1 = u_1 + \psi, \quad w_2 = u_2 + \varphi + \psi, \quad w_3 = u_3 + \varphi.$$

It can be proven that the condition is satisfied. □

Theorem 10. *For a given list of signatures of simple subparquets $U_0, \dots, U_{\ell-1}$, list of signatures of subparquets $V_0, \dots, V_{\ell-1}$ and a set of vectors Q we can calculate*

$$\sum_{i=0}^{\ell-1} |(U_i + q) \cap V_i|$$

for every $q \in Q$ in total time $\tilde{\mathcal{O}}(m^2 + \ell + |Q|)$, assuming that the subparquets only contain vectors of length $\mathcal{O}(m)$.

Proof. We apply Lemma 13 to every U_i and find $w_{i,0}, \dots, w_{i,3}$, so that we have

$$\sum_{i=0}^{\ell-1} |(U_i + q) \cap V_i| = \sum_{i=0}^{\ell-1} |U_i \cap (V_i - q)| = \sum_{i=0}^{\ell-1} \sum_{j=0}^{3} (-1)^j |(D + w_{i,j}) \cap (V_i - q)| =$$
$$= \sum_{j=0}^{3} (-1)^j \sum_{i=0}^{\ell-1} |(D + q) \cap (V_i - w_{i,j})|.$$

By Lemma 12 we can independently calculate the values $\sum_{i=0}^{\ell-1} |(D + q) \cap (V_i - w_{i,j})|$ for every j by running the algorithm for $\mathcal{V}_j = \{ V_i - w_{i,j} : i \in [\ell] \}$ and Q. □

Theorem 11. *For a given set of monochromatic simple subparquet strings $\mathcal{S}$ we can calculate*

$$\sum_{S \in \mathcal{S}} \mathrm{Ham}(P + q, S)$$

for every $q \in Q$ in total time $\tilde{\mathcal{O}}(m^2 + \sum_{S \in \mathcal{S}} |\mathcal{V}_{\mathrm{C}(S)}|)$, assuming that the sets $\mathrm{dom}(S)$ for $S \in \mathcal{S}$ are some pairwise disjoint subsets of $\mathrm{dom}(T)$.

Proof. Let $U = \bigcup_{S \in \mathcal{S}} \mathrm{dom}(S)$. Observe that

$$\sum_{S \in \mathcal{S}} \mathrm{Ham}(P + q, S) = |(P + q) \cap U| - \sum_{S \in \mathcal{S}} \sum_{V \in \mathcal{V}_{\mathrm{C}(S)}} |\operatorname{dom}(V + q) \cap \operatorname{dom}(S)|.$$

We can calculate $|(P + q) \cap U|$ for every $q \in Q$ with a single instance of FFT (see Theorem 3) or by using prefix sums in time $\tilde{\mathcal{O}}(m^2)$. To calculate the values

$$\sum_{S \in \mathcal{S}} \sum_{V \in \mathcal{V}_{\mathrm{C}(S)}} |\operatorname{dom}(V + q) \cap \operatorname{dom}(S)|$$

we use the algorithm from Theorem 10 (where $\ell = \sum_{S \in \mathcal{S}} |\mathcal{V}_{\mathrm{C}(S)}|$). □

5.6 Periodic parquet partitioning

In this section we explore the properties of periodic (sub-)parquet strings (recall Definitions 8, 9, 10). Specifically, we introduce some methods of partitioning them into monochromatic strings, which we utilize when decomposing both the pattern and the active text.

Definition 14 (Lattice graph)**.** For a set $U \subseteq \mathbb{Z}^2$ we define its **lattice graph** $(U, \mathrm{E}(U))$, where

$$\mathrm{E}(U) = \big\{ \{ u, u+\delta \} : \delta \in \{ \varphi, \psi \}, u \in U, u+\delta \in U \big\},$$

so every vector is connected with its translations by $\varphi, \psi, -\varphi, -\psi$, which are contained in U.

Lemma 14. *If U is a spacious subparquet, then $(U, \mathrm{E}(U))$ is connected.*

Proof. Assume the contrary. Consider any pair of points $u, v \in U$, such that

- u and v belong to different connected components,
- if we let $s, t \in \mathbb{Z}$ be such that $v = u + s\varphi + t\psi$, then $|s| + |t|$ is minimized.

Let us assume that for such u and $v = u + s\varphi + t\psi$ we have $s \geq 0$, since in the other case they can be swapped. We now show that there exists a point $w \in U$, such that $\{ u, w \} \in \mathrm{E}(U)$ and if we let $s', t' \in \mathbb{Z}$ be such that $v = w + s'\varphi + t'\psi$, then $|s'| + |t'| < |s| + |t|$, which contradicts the minimality of $|s| + |t|$.

Let $x_0, x_1, y_0, y_1, \varphi_0, \varphi_1, \psi_0, \psi_1 \in \mathbb{Z}$ be such that

- $U = [x_0, x_1] \times [y_0, y_1] \cap \{ w : w \in \mathbb{Z}^2, \varphi \times w \in [\varphi_0, \varphi_1], \psi \times w \in [\psi_0, \psi_1], w \equiv u \}$,
- $x_1 - x_0 + 1 \geq |\varphi.x| + |\psi.x|$ and $y_1 - y_0 + 1 \geq |\varphi.y| + |\psi.y|$.

They exist by definition of a spacious subparquet. Recall that $\varphi.x \geq 0$, $\varphi.y \leq 0$, $\psi.x \geq 0$, $\psi.y \geq 0$. We have the following cases:

1° If $s = 0$ and $t > 0$, then $w = u + \psi$. Observe that

$$u.x \leq w.x \leq v.x, \quad u.y \leq w.y \leq v.y, \quad \varphi \times u \leq \varphi \times w \leq \varphi \times v, \quad \psi \times w = \psi \times u,$$

and since $w \equiv u$, we get $w \in U$.

2° If $s = 0$ and $t < 0$, then $w = u - \psi$ and we can similarly show that $w \in U$, since

$$v.x \leq w.x \leq u.x, \quad v.y \leq w.y \leq u.y, \quad \varphi \times v \leq \varphi \times w \leq \varphi \times u, \quad \psi \times w = \psi \times u.$$

3° If $s > 0$ and $t = 0$, then $w = u + \varphi$ and we get $w \in U$, since

$$u.x \leq w.x \leq v.x, \quad v.y \leq w.y \leq u.y, \quad \varphi \times v = \varphi \times u, \quad \psi \times v \leq \psi \times w \leq \psi \times u.$$

4° If $s > 0$ and $t > 0$, consider the point $w' = u + \varphi$. We have

$$u.x \leq w'.x \leq v.x, \quad \varphi \times w' = \varphi \times u, \quad \psi \times v \leq \psi \times w' \leq \psi \times u.$$

If $w' \in U$, then $w = w'$. If $w' \notin U$, then since all other requirements are satisfied, we must have $w'.y \notin [y_0, y_1]$. Since $w'.y = u.y + \varphi.y \leq u.y$, we have $u.y + \varphi.y \leq y_0 - 1$, and considering $y_1 - y_0 + 1 \geq |\varphi.y| + |\psi.y|$, we get $y_1 \geq u.y + \psi.y$. Now let $w = u + \psi$. We have

$$u.x \leq w.x \leq v.x, \quad u.y \leq w.y = u.y + \psi.y \leq y_1, \quad \varphi \times u \leq \varphi \times w \leq \varphi \times v, \quad \psi \times w = \psi \times u,$$

thus $w \in U$.

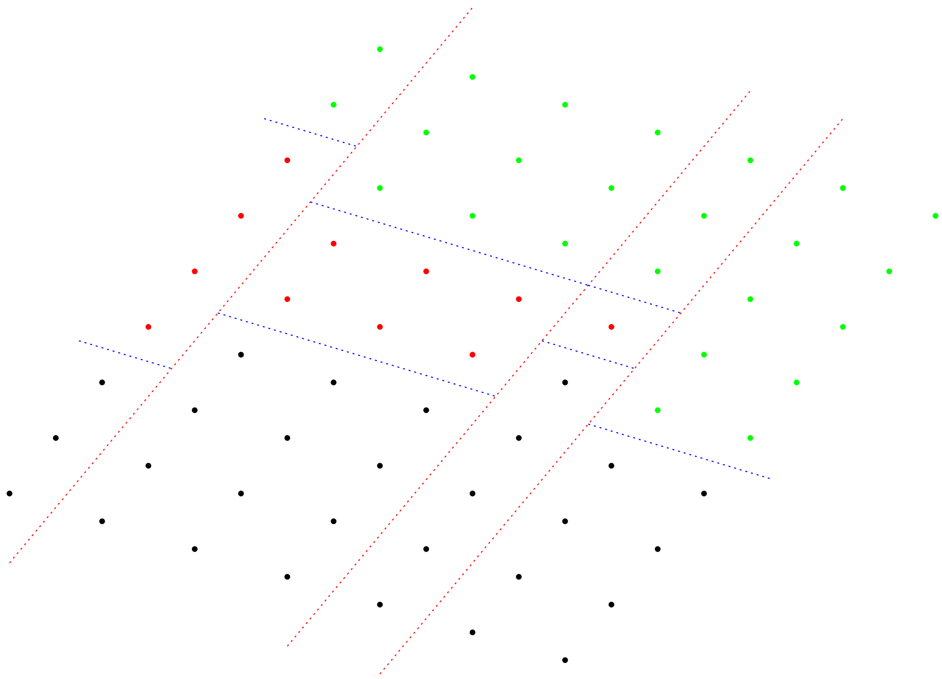

Figure 4: The partitioning of a simple subparquet string into monochromatic simple subparquet strings. Different colors represent different characters assigned to a point.

5° If $s > 0$ and $t < 0$, consider $w' = u + \varphi$. We have

$$v.y \leq w'.y \leq u.y, \quad \varphi \times w' = \varphi \times u, \quad \psi \times v \leq \psi \times w' \leq \psi \times u.$$

If $w' \in U$, then $w = w'$. Otherwise we can (similarly to 4°) show that $w = u - \psi \in U$, by the fact that $x_1 - x_0 + 1 \geq |\varphi.x| + |\psi.x|$. □

Lemma 15. *A spacious subparquet string S is monochromatic if and only if*

$$\mathrm{Ham}(S + \varphi, S) + \mathrm{Ham}(S + \psi) = 0.$$

Proof. If S is monochromatic, then clearly $\mathrm{Ham}(S + \varphi, S) + \mathrm{Ham}(S + \psi, S) = 0$. Assume the contrary for the other implication. Let $u, v \in S$ be such that $S(u) \neq S(v)$. Since $\mathrm{dom}(S)$ is a spacious subparquet, the graph $(\mathrm{dom}(S), \mathrm{E}(\mathrm{dom}(S))$ is connected (by Lemma 14) and there must exist a path between u and v. On that path there must exist a pair of neighbors w, w', such that $S(w) \neq S(w')$ and $w' = w + \delta$ for some $\delta \in \{\varphi, \psi\}$. If $\delta = \varphi$, then $\mathrm{Ham}(S + \varphi, S) \geq 1$ and if $\delta = \psi$, then $\mathrm{Ham}(S + \psi, S) \geq 1$ and we get a contradiction. □

Lemma 16. *A spacious subparquet string S can be partitioned in time $\tilde{\mathcal{O}}(|\mathrm{dom}(S)| + 1)$ into both the following sets of strings (we have two options):*

a) a set of $\mathcal{O}(\mathrm{Ham}(S + \varphi, S) + 1)$ strings $\mathcal{U}$, such that $\mathrm{Ham}(U + \varphi, U) = 0$ for each $U \in \mathcal{U}$ and

b) a set of $\mathcal{O}(\mathrm{Ham}(S + \psi, S) + 1)$ strings $\mathcal{V}$, such that $\mathrm{Ham}(V + \psi, V) = 0$ for each $V \in \mathcal{V}$.

All the obtained strings are spacious and if S is simple, they are simple.

Proof. Let us consider option (a). We construct the set

$$A = \{\, \psi \times u : u \in S, u + \varphi \in S, S(u) \neq S(u + \varphi) \,\} \cup \{\, -\infty, +\infty \,\}$$

and then sort its elements increasingly, creating an array $a_0, \dots, a_\ell$. Note that $\ell \le \mathrm{Ham}(S+\varphi, S) + 2$. We then construct the strings $S_0, \dots, S_{\ell-1}$, where S_i is the restriction of S to $\{ u : u \in S, \psi \times u \in [a_i, a_{i+1}) \}$ for every $i \in [\ell]$. Observe that $S_0, \dots, S_{\ell-1}$ partition S and that $\mathrm{Ham}(S_i + \varphi, S_i) = 0$ for every $i \in [\ell]$. Also, is S is spacious, then they are spacious and if S is simple, then they are simple.

In the case of option (b), we similarly construct

$$A = \{ \varphi \times u : u \in S, u + \psi \in S, S(u) \neq S(u + \psi) \} \cup \{ -\infty, +\infty \}$$

and then sort it increasingly, creating $a_0, \dots, a_\ell$, where $\ell \le \mathrm{Ham}(S+\psi, S)+2$. We then construct the strings $S_0, \dots, S_{\ell-1}$, where S_i is the restriction of S to $\{ u : u \in S, \varphi \times u \in (a_i, a_{i+1}] \}$. $\square$

Theorem 12. *A given spacious/simple parquet string R with $\mathcal{O}(k)$-periods φ and ψ can be partitioned in time $\tilde{\mathcal{O}}(|\operatorname{dom}(R)| + k)$ into $\mathcal{O}(k)$ monochromatic spacious/simple subparquet strings, correspondingly.*

Proof. We partition R into a set of subparquet strings $\mathcal{S}$, such that $|\mathcal{S}| = \mathcal{O}(\min\{ m, k \})$. Specifically, for each $\gamma \in \Gamma$ (see Lemma 4), we construct a restriction of R to $\operatorname{dom}(R) \cap (\mathcal{L} + \gamma)$. Observe that if R is spacious, then all $S \in \mathcal{S}$ are spacious and if R is simple, then all $S \in \mathcal{S}$ are simple. We now partition each $S \in \mathcal{S}$ independently by using Lemma 16 (a) and construct a set of subparquet strings $\mathcal{S}'$, such that $\mathcal{S}'$ partitions R and $\mathrm{Ham}(S' + \varphi, S') = 0$ for every $S' \in \mathcal{S}'$. Note that

$$|S'| = \sum_{S \in \mathcal{S}} \mathcal{O}(\mathrm{Ham}(S + \varphi, S) + 1) = \mathcal{O}(\mathrm{Ham}(R + \varphi, R) + |\mathcal{S}|) = \mathcal{O}(k),$$

since R has an $\mathcal{O}(k)$-period φ. We now partition each $S' \in \mathcal{S}'$ by using Lemma 16 (b) and construct a set of subparquet strings $\mathcal{S}''$, such that $\mathcal{S}''$ partitions R and $\mathrm{Ham}(S'' + \psi, S'') = 0$ for every $S'' \in \mathcal{S}''$. Again we have

$$|S''| = \sum_{S' \in \mathcal{S}'} \mathcal{O}(\mathrm{Ham}(S' + \psi, S') + 1) = \mathcal{O}(\mathrm{Ham}(R + \psi, R) + |\mathcal{S}'|) = \mathcal{O}(k),$$

since R has an $\mathcal{O}(k)$-period ψ. The process is illustrated in Figure 4. The red lines represent the partitioning done in the first phase, when constructing $\mathcal{S}'$, and blue in the second, when constructing $\mathcal{S}''$. By Lemma 15, the strings $S'' \in \mathcal{S}''$ are monochromatic. The total number of operations is $\mathcal{O}(|\operatorname{dom}(R)| + k)$. $\square$

5.7 Active text decomposition

This section serves as the proof of the following major theorem:

Theorem 13. *For a given positive integer $\ell \le m$, we can partition the active text in time $\tilde{\mathcal{O}}(m^2 + \ell k)$ into a set of $\mathcal{O}(\ell k)$ monochromatic simple subparquet strings and an $\mathcal{O}(m/\ell)$-peripheral string.*

We will use a more geometrical approach and construct some lines and parallelograms. For the sake of simplicity, we will consider an empty set to be a valid parallelogram. Also, we assume that a parallelogram contains the points laying on its border and its vertices.

Definition 15. For a set of points $U \subseteq \mathbb{R}^2$ we will denote

$$\mathrm{X}(U) = \{ u.x : u \in U \}, \quad \mathrm{Y}(U) = \{ u.y : u \in U \}.$$

Observation 5. *For any given $\ell \in \mathbb{Z}^+$ and $v \in \mathbb{Z}^2$ we can find an array of parallel lines $f_0, f_1, \dots, f_\ell$, where $f_i = \{ u : u \in \mathbb{R}^2, v \times u = c_i \}$ for some $c_i \in \mathbb{R} \setminus \mathbb{Q}$, such that*

- *$c_0 < v \times u < c_\ell$ for every $u \in [n]^2$, or namely, the set $[n]^2$ is between f_0 and f_ℓ,*
- *$0 < c_{i+1} - c_i = \mathcal{O}(n|v|/\ell)$ for every $i \in [\ell]$, or namely, the distance between every two consecutive lines is $\mathcal{O}(n/\ell)$.*

We use Observation 5 with $v = \varphi$ to construct the lines $h_0, \ldots, h_\ell$ and with $v = \psi$ to construct the lines $s_0, \ldots, s_\ell$. For every $i, j \in [\ell + 1]$ we construct a point $w_{i,j}$ as an intersection of h_i and s_j (it is easy to see that since φ and ψ are not colinear, h_i and s_j are not parallel). For every $i, j \in [\ell]$ we construct a parallelogram $p_{i,j}$ defined as the area between s_i and s_{i+1} intersected with the area between h_j and h_{j+1}. Specifically,

$$p_{i,j} = \{\, u : u \in \mathbb{R}^2, \varphi \times u \in [\varphi \times w_{i,j}, \varphi \times w_{i+1,j+1}], \psi \times u \in [\psi \times w_{i,j}, \psi \times w_{i+1,j+1}] \,\}\,.$$

For better reference, the vertices of $p_{i,j}$ are $w_{i,j}, w_{i+1,j}, w_{i+1,j+1}, w_{i,j+1}$. Observe that every $u \in [n]^2$ is contained strictly inside exactly one parallelogram $p_{i,j}$.

Lemma 17. *For every $i \in [\ell - 1]$ and $j \in [\ell]$ we have*

$$\begin{aligned} \min \mathrm{X}(p_{i,j}) &< \min \mathrm{X}(p_{i+1,j}), & \min \mathrm{Y}(p_{i,j}) &\le \min \mathrm{Y}(p_{i+1,j}), \\ \max \mathrm{X}(p_{i,j}) &< \max \mathrm{X}(p_{i+1,j}), & \max \mathrm{Y}(p_{i,j}) &\le \max \mathrm{Y}(p_{i+1,j}) \end{aligned}$$

and for every $i \in [\ell]$ and $j \in [\ell - 1]$ we have

$$\begin{aligned} \min \mathrm{X}(p_{i,j}) &\ge \min \mathrm{X}(p_{i,j+1}), & \min \mathrm{Y}(p_{i,j}) &< \min \mathrm{Y}(p_{i,j+1}), \\ \max \mathrm{X}(p_{i,j}) &\ge \max \mathrm{X}(p_{i,j+1}), & \max \mathrm{Y}(p_{i,j}) &< \max \mathrm{Y}(p_{i,j+1}). \end{aligned}$$

Proof. It follows from the fact that we selected $\varphi \in [0, +\infty) \times (-\infty, 0)$ and $\psi \in (0, +\infty) \times [0, +\infty)$. For example, to prove the first inequality, we can consider a point $u \in p_{i+1,j}$, such that $u.x = \min \mathrm{X}(p_{i+1,j})$ and then construct a point $v \in p_{i,j}$, such that $v = u - t\psi$ for some $t > 0$, and thus $\min \mathrm{X}(p_{i,j}) \le v.x \le u.x = \min \mathrm{X}(p_{i+1,j})$. The other inequalities can be proven analogously. □

Lemma 25. *For every $i, j \in [\ell]$ and every $u, v \in \mathrm{X}(p_{i,j}) \times \mathrm{Y}(p_{i,j})$, we have $|u - v| = \mathcal{O}(n/\ell)$.*

Proof. See Section 5.7.2. □

Consider the case when $\max \mathrm{X}(p_{i,j}) - \min \mathrm{X}(p_{i,j}) \ge m/4$ for some $i, j \in [\ell]$. By Lemma 25, we would have $m/4 \le \max \mathrm{X}(p_{i,j}) - \min \mathrm{X}(p_{i,j}) = \mathcal{O}(n/\ell)$, and thus $\ell = \mathcal{O}(1)$. In that case we can return a trivial partitioning where $F = T_{\mathbf{a}}$ and the set of monochromatic strings is empty, since $T_{\mathbf{a}}$ is $\mathcal{O}(m)$-peripheral. We can use the same argument if we have $\max \mathrm{Y}(p_{i,j}) - \min \mathrm{Y}(p_{i,j}) \ge m/4$ for some $i, j \in [\ell]$. Thus, from now on we will assume that $\max X_{i,j} - \min X_{i,j} < m/4$ and $\max Y_{i,j} - \min Y_{i,j} < m/4$ for every $i, j \in [\ell]$.

Let $z = \frac{n-1}{2}$. We split the plane with two lines $x = z$ and $y = z$ into four quarters:

1) $K_1 = (z, +\infty) \times (z, +\infty)$,
2) $K_2 = (-\infty, z) \times (z, +\infty)$,
3) $K_3 = (-\infty, z) \times (-\infty, z)$,
4) $K_4 = (z, +\infty) \times (-\infty, z)$.

Let us denote by $\mathcal{I}$ the set of all parallelograms $p_{i,j}$, such that they intersect with the line $x = z$ or with the line $y = z$ (or both). Observe that every parallelogram $p_{i,j} \notin \mathcal{I}$ must be fully contained in one of the quarters, meaning $p_{i,j} \subseteq K_d$ for some $d \in \{\, 1, \ldots, 4 \,\}$.

Lemma 18. $|\mathcal{I}| = \mathcal{O}(\ell)$.

Proof. Consider the line $x = z$, denoted f. It intersects with every line $h_0, \ldots, h_\ell$ at most once (and does not overlap with any of them). Similarly, it intersects with every line $s_0, \ldots, s_\ell$ at most once. Denote the set of such intersections as U. For every parallelogram $p \in \mathcal{I}$, there must exist $u \in U$, such that $u \in p$. For every $u \in U$, there are at most four parallelograms $p \in \mathcal{I}$, such that $u \in p$, thus the number of parallelograms intersecting with f is at most $4|U| = \mathcal{O}(\ell)$. We can identically bound the number of parallelograms intersecting with the line $y = z$, and thus get $|\mathcal{I}| = \mathcal{O}(\ell)$. $\square$

Now consider any $j \in [\ell]$. By Lemma 17, we can find $s, t \in [\ell + 1]$, such that the array $p_{0,j}, \ldots, p_{\ell-1,j}$ is split into three groups:

a) $p_{0,j}, \ldots, p_{s-1,j}$, which includes only parallelograms fully contained in K_3,

b) $p_{s,j}, \ldots, p_{t-1,j}$, which does not include any parallelogram fully contained in K_1 or K_3,

c) $p_{t,j}, \ldots, p_{\ell-1,j}$, which includes only parallelograms fully contained in K_1.

We now "merge together" the parallelograms from group (a) and from group (c). Specifically, we construct

$$g_j^3 = \bigcup_{i=0}^{s-1} p_{i,j}, \quad g_j^1 = \bigcup_{i=t}^{\ell-1} p_{i,j}.$$

We do it for every $j \in [\ell]$. Observe that the sets $g_0^1, \ldots, g_{\ell-1}^1$ are are parallelograms (possibly empty) and that they cover the same area as all the fully contained in K_1 parallelograms $p_{i,j}$. The same is true for $g_0^3, \ldots, g_{\ell-1}^3$ and the parallelograms in K_3.

Now for every $i \in [\ell]$ we similarly find $s, t \in [\ell + 1]$, such that the array $p_{i,0}, \ldots, p_{i,\ell-1}$ is split into three groups:

a) $p_{i,0}, \ldots, p_{i,s-1}$, which includes only parallelograms fully contained in K_4,

b) $p_{i,s}, \ldots, p_{i,t-1}$, which does not include any parallelogram fully contained in K_2 or K_4,

c) $p_{i,t}, \ldots, p_{i,\ell-1}$, which includes only parallelograms fully contained in K_2,

and then construct

$$g_i^4 = \bigcup_{j=0}^{s-1} p_{i,j}, \quad g_i^2 = \bigcup_{j=t}^{\ell-1} p_{i,j}.$$

We denote

$$\mathcal{G} = \{ g_0^1, \ldots, g_{\ell-1}^1 \} \cup \{ g_0^2, \ldots, g_{\ell-1}^2 \} \cup \{ g_0^3, \ldots, g_{\ell-1}^3 \} \cup \{ g_0^4, \ldots, g_{\ell-1}^4 \}.$$

Again observe that for every $u \in [n]^2$ there exists exactly one parallelogram $p \in \mathcal{G} \cup \mathcal{I}$, such that $u \in p$, and since the sides of p do not contain integer points, u lays strictly inside p.

Definition 16 (Coverability)**.** We say that a set $U \subseteq \mathbb{Z}^2$ is **coverable** if $U \subseteq \mathrm{dom}(P + q)$ for some $q \in Q$.

Lemma 19. *For every $p \in \mathcal{I}$, the set $p \cap \mathbb{Z}^2$ is either coverable or $\mathcal{O}(n/\ell)$-peripheral.*

Proof. Consider any $p \in \mathcal{I}$. Since the other cases are rotationally symmetric, assume that it intersects with some point $s \in \mathbb{R}^2$, such that $s.x = z$ and $s.y \geq z$. Let $v = (\lfloor \max \mathrm{X}(p), \max \mathrm{Y}(p) \rfloor)$. We have $v.x \geq \lfloor z \rfloor = n/2 - 1$ and $v.y \geq \lfloor z \rfloor = n/2 - 1$. If $v \notin T_{\mathbf{a}}$, we can see that by Lemma 25, $|u - v| = \mathcal{O}(n/\ell)$ for every $u \in p \cap \mathbb{Z}^2$, thus p is $\mathcal{O}(n/\ell)$-peripheral. If $v \in T_{\mathbf{a}}$,

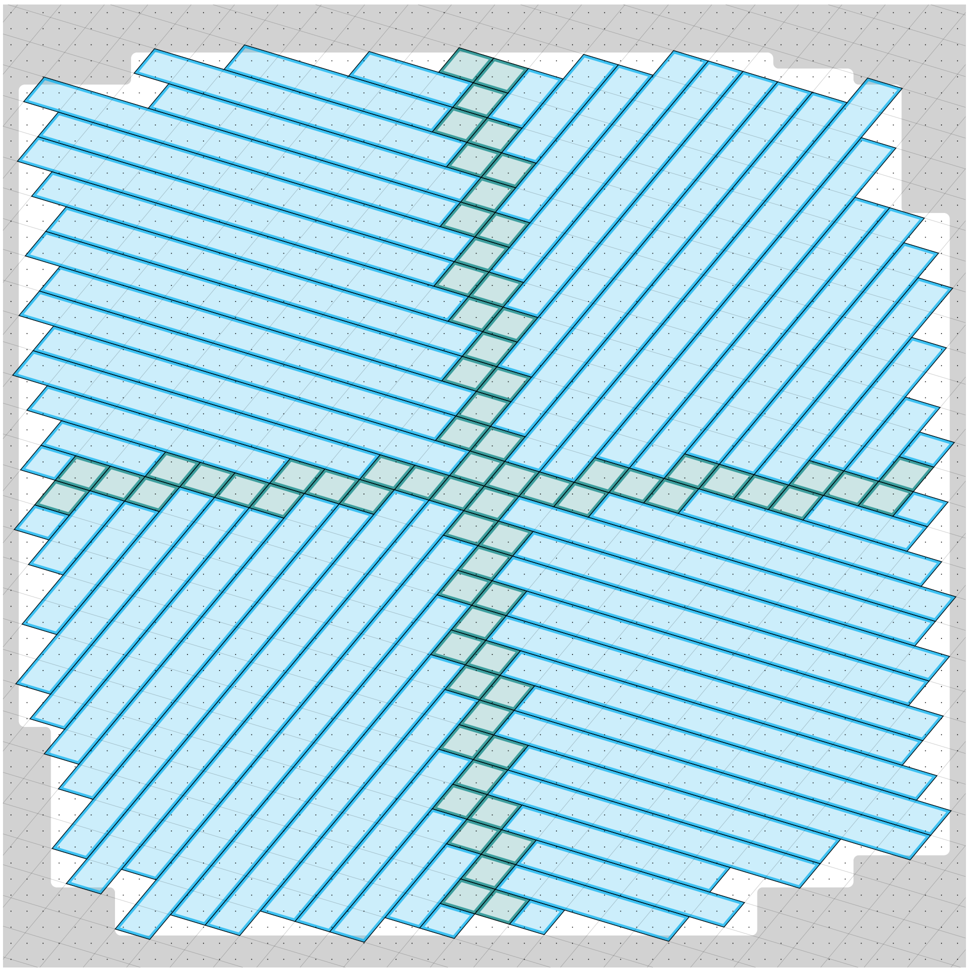

Figure 5: The parallelograms from $\mathcal{C}$ (blue) and $\mathcal{C}'$ (green).

there exists $q \in Q$, such that $v \in [m]^2 + q$. Consider any $u \in p \cap \mathbb{Z}^2$. By the assumption that $\max \mathrm{X}(p) - \min \mathrm{X}(p) < m/4$ and $\max \mathrm{Y}(p) - \min \mathrm{Y}(p) < m/4$ we get

$$u.x \geq n/2 - m/4 \geq n - m \geq q.x,$$
$$u.y \geq n/2 - m/4 \geq n - m \geq q.y,$$

and since $u.x \leq v.x \leq q.x + m - 1$ and $u.y \leq v.y \leq q.y + m - 1$, we get $u \in [m]^2 + q$, thus $U \subseteq [m]^2 + q$. □

Lemma 20. *The restriction of T to a coverable set has $\mathcal{O}(k)$-periods φ and ψ.*

Proof. Let R denote the restriction. For $q \in Q$, such that $\mathrm{dom}(R) \subseteq \mathrm{dom}(P + q)$ we have

$$\begin{aligned}\mathrm{Ham}(R + \varphi, R) &\leq \mathrm{Ham}(R + \varphi, P + q + \varphi) + \mathrm{Ham}(P + q + \varphi, P + q) + \mathrm{Ham}(P + q, R) \leq \\ &\leq \mathrm{Ham}(T, P + q) + \mathrm{Ham}(P + \varphi, P) + \mathrm{Ham}(P + q, T) = \mathcal{O}(k)\end{aligned}$$

and identically $\mathrm{Ham}(R + \psi, R) = \mathcal{O}(k)$. □

Lemma 21. *For every $g \in \mathcal{G}$ we can construct two parallelograms c and b, such that*

- *$c \cap \mathbb{Z}^2$ is coverable,*
- *$b \cap \mathbb{Z}^2$ is $\mathcal{O}(n/\ell)$-peripheral,*

- $g \cap \mathbb{Z}^2$ *is partitioned into* $b \cap \mathbb{Z}^2$ *and* $c \cap \mathbb{Z}^2$.

Proof. See the next section (5.7.1). $\square$

We split every non-empty parallelogram $g \in \mathcal{G}$ (by Lemma 21) into parallelograms c and b. We construct the set $\mathcal{C}$ consisting of all the obtained parallelograms c and a set $\mathcal{B}$ consisting of all the obtained parallelograms b.

We similarly divide the parallelograms in $\mathcal{I}$ (by Lemma 19) and construct the sets $\mathcal{C}' = \{\, p : p \in \mathcal{I}, p \cap \mathbb{Z}^2 \text{ is coverable} \,\}$ and $\mathcal{B}' = \mathcal{I} \setminus \mathcal{C}'$.

Now construct $\mathcal{U} = \{\, c \cap \mathbb{Z}^2 : c \in \mathcal{C} \cup \mathcal{C}' \,\}$ and $V = \bigcup_{b \in \mathcal{B} \cup \mathcal{B}'} b \cap \operatorname{dom}(T_{\mathbf{a}})$. Observe that all sets $U \in \mathcal{U}$ are coverable simple parquets, the set V is $\mathcal{O}(n/\ell)$-peripheral, and $\operatorname{dom}(T_{\mathbf{a}})$ is partitioned into sets $\mathcal{U} \cup \{\, V \,\}$.

The decomposition is illustrated in Figure 5. The points in the gray area are outside of the active text. The parallelograms from $\mathcal{C}$ and $\mathcal{C}'$ are colored blue and green, respectively. The points outside of them (in the white area) form the peripheral set V.

For each $U \in \mathcal{U}$ we construct the restriction of T to U. By Lemma 20, it has $\mathcal{O}(k)$-periods φ and ψ, thus, by Theorem 12, it can be partitioned into $\mathcal{O}(k)$ monochromatic simple subparquet strings. Since $|\mathcal{C}'| \le |\mathcal{I}| = \mathcal{O}(\ell)$ (by Lemma 18) and $|\mathcal{C}| \le |\mathcal{G}| = \mathcal{O}(\ell)$, we have $|\mathcal{U}| = |\mathcal{C}| + |\mathcal{C}'| = \mathcal{O}(\ell)$, thus the total number of constructed strings is $\mathcal{O}(\ell k)$.

Finally, we construct the restriction of T to V, which is a $\mathcal{O}(n/\ell)$-peripheral string.

5.7.1 Parallelogram splitting

This section serves as the proof of Lemma 21, introduced at the end of the previous section (5.7). Since for an empty parallelogram the proof is trivial, consider a non-empty set g^1_j for some $j \in [\ell]$. We will explore some properties of the part of the text contained in K_1 specifically, which can be generalized to other quarters by symmetry.

Lemma 22. *Every set* $U \subseteq K_1 \cap \mathbb{Z}^2$*, such that* $(\max \mathrm{X}(U), \max \mathrm{Y}(U)) \in T_{\mathbf{a}}$ *is coverable.*

Proof. Let $v = (\max \mathrm{X}(U), \max \mathrm{Y}(U))$. By assumption, there exists $q \in Q$, such that $v \in [m]^2 + q$. For every $u \in U$ we have $q.x \le n - m \le n/2 \le u.x \le v.x < q.x + m$ and $q.y \le n - m \le n/2 \le u.y \le v.y < q.y + m$, thus $u \in [m]^2 + q$. $\square$

Observation 6. *By Lemma 22, there does not exist a pair of points* $u \in \mathbb{Z}^2 \cap K_1 \setminus \operatorname{dom}(T_{\mathbf{a}})$ *and* $v \in T_{\mathbf{a}}$*, such that* $u.x \le v.x$ *and* $u.y \le v.y$*.*

Recall that there exists $t \in [\ell + 1]$, such that $g^1_j = \bigcup_{i=t}^{\ell-1} p_{i,j}$. We find

$$f = \min \{\, i : i \in \{\, t, \ldots, \ell - 1 \,\}, (\lfloor \max \mathrm{X}(p_{i,j}) \rfloor, \lfloor \max \mathrm{Y}(p_{i,j}) \rfloor) \in \mathbb{Z}^2 \setminus \operatorname{dom}(T_{\mathbf{a}}) \,\}.$$

If the minimum does not exist, we consider $f = \ell$. We then construct the parallelograms

$$c = \bigcup_{i=0}^{f-1} p_{i,j}, \quad b = \bigcup_{i=f}^{\ell-1} p_{i,j}.$$

We now show that the set $c \cap \mathbb{Z}^2$ is coverable. If $c \cap \mathbb{Z}^2$ is empty, then it is coverable, so let us assume it is not. In that case $f > 0$. It is clear that $c \cap \mathbb{Z}^2 \subseteq K_1$. By Lemma 17, we have

$$\max \mathrm{X}(c) = \max \mathrm{X}(p_{f-1,j}),$$
$$\max \mathrm{X}(c) = \max \mathrm{Y}(p_{f-1,j}),$$

and thus

$$\max \mathrm{X}(c \cap \mathbb{Z}^2) \le \lfloor \max \mathrm{X}(c) \rfloor = \lfloor \max \mathrm{X}(p_{f-1,j}) \rfloor,$$
$$\max \mathrm{Y}(c \cap \mathbb{Z}^2) \le \lfloor \max \mathrm{Y}(c) \rfloor = \lfloor \max \mathrm{Y}(p_{f-1,j}) \rfloor.$$

We see that $(\max \mathrm{X}(c \cap \mathbb{Z}^2), \max \mathrm{Y}(c \cap \mathbb{Z}^2)) \in T_{\mathbf{a}}$, since it would otherwise contradict Observation 6, considering that $(\lfloor \max \mathrm{X}(p_{f-1,j}) \rfloor, \lfloor \max \mathrm{Y}(p_{f-1,j}) \rfloor) \in T_{\mathbf{a}}$. We see that $c \cap \mathbb{Z}^2$ satisfies the conditions of Lemma 22, thus $c \cap \mathbb{Z}^2$ is coverable.

We now show that the set $b \cap \mathbb{Z}^2$ is $\mathcal{O}(n/\ell)$-peripheral. If it is empty, then the proof is trivial, so let us assume it is not. In that case $f < \ell$. Denote $v = (\lfloor \max \mathrm{X}(p_{f,j}) \rfloor, \lfloor \max \mathrm{Y}(p_{f,j}) \rfloor)$. By definition, $v \in \mathbb{Z}^2 \setminus \mathrm{dom}(T_{\mathbf{a}})$. Consider any point $u \in b \cap \mathbb{Z}^2$. There exists exactly one $i \in \{ f, \ldots, \ell - 1 \}$, such that u lays strictly inside $p_{i,j}$. Let $w = (\lfloor \max \mathrm{X}(p_{i,j}) \rfloor, \lfloor \max \mathrm{Y}(p_{i,j}) \rfloor)$. By Lemma 17, we have $w.x \geq v.x$ and $w.y \geq v.y$, and by considering Observation 6, we get $w \in \mathbb{Z}^2 \setminus \mathrm{dom}(T_{\mathbf{a}})$. Finally, by Lemma 25, we get $|u - w| = \mathcal{O}(n/\ell)$.

The constructions for g_i^2, g_j^3, g_i^4 are rotationally symmetric.

5.7.2 Parallelogram span bounds

In this section we will establish a distance bound for points laying inside or in the proximity of the constructed parallelograms. Consider any fixed $p_{i,j}$ for some $i, j \in [\ell]$. We first show some auxiliary (weaker) lemmas, which we then use to prove Lemma 25.

Lemma 23. *For every $u, v \in p_{i,j}$, we have $|u - v| = \mathcal{O}(n/\ell)$.*

Proof. Consider any $u, v \in p_{i,j}$ and denote $w = u - v$. By definition of $p_{i,j}$, we have

$$|\varphi \times w| = |\varphi \times (u - v)| = |\varphi \times u - \varphi \times v| = \mathcal{O}(n|\varphi|/\ell)$$

and similarly $|\psi \times w| = \mathcal{O}(n|\psi|/\ell)$. Since φ and ψ are not colinear, there exist $s, t \in \mathbb{R}$, such that $w = s\varphi + t\psi$. Recall that by Theorem 9 we have $|\varphi \times \psi| \geq \frac{1}{2}|\varphi||\psi|$ (since $|\sin \alpha| \geq 1/2$), thus

$$\frac{1}{2}|t||\varphi||\psi| \leq |t||\varphi \times \psi| = |\varphi \times (s\varphi + t\psi)| = |\varphi \times w| = \mathcal{O}(n|\varphi|/\ell),$$

which gives us $|t\varphi| = \mathcal{O}(n/\ell)$. We can similarly prove that $|s\psi| = \mathcal{O}(n/\ell)$ and finally

$$|w| = |s\varphi + t\psi| \leq |s\varphi| + |t\psi| = \mathcal{O}(n/\ell).$$

□

Lemma 24. *For every point $u \in \mathrm{X}(p_{i,j}) \times \mathrm{Y}(p_{i,j})$ there exists a point $v \in p_{i,j}$, such that $|u - v| = \mathcal{O}(n/\ell)$.*

Proof. There exists a point $w \in p_{i,j}$, such that $u.x = w.x$, and a point $v \in p_{i,j}$, such that $u.y = v.y$. By Lemma 23, we have

$$|u - v| = |u.x - v.x| = |w.x - v.x| \leq |w - v| = \mathcal{O}(n/\ell).$$

□

Lemma 25. *For every $i, j \in [\ell]$ and every $u, v \in \mathrm{X}(p_{i,j}) \times \mathrm{Y}(p_{i,j})$, we have $|u - v| = \mathcal{O}(n/\ell)$.*

Proof. Consider any $u, v \in p_{i,j}$. By Lemma 24, there exist $u', v' \in p_{i,j}$ such that $|u - u'| = \mathcal{O}(n/\ell)$ and $|v - v'| = \mathcal{O}(n/\ell)$. By Lemma 23, we have

$$|u - v| \leq |u - u'| + |u' - v'| + |v' - v| = \mathcal{O}(n/\ell).$$

□

References

[1] Karl R. Abrahamson. Generalized string matching. *SIAM J. Comput.*, 16(6):1039–1051, 1987. doi:10.1137/0216067.

[2] Amihood Amir, Yonatan Aumann, Moshe Lewenstein, and Ely Porat. Function matching. *SIAM J. Comput.*, 35(5):1007–1022, 2006. doi:10.1137/S0097539702424496.

[3] Amihood Amir and Gary Benson. Two-dimensional periodicity and its applications. In Greg N. Frederickson, editor, *Proceedings of the Third Annual ACM/SIGACT-SIAM Symposium on Discrete Algorithms, 27-29 January 1992, Orlando, Florida, USA*, pages 440–452. ACM/SIAM, 1992. URL: http://dl.acm.org/citation.cfm?id=139404.139489.

[4] Amihood Amir and Gary Benson. Two-dimensional periodicity in rectangular arrays. *SIAM J. Comput.*, 27(1):90–106, 1998. doi:10.1137/S0097539795298321.

[5] Amihood Amir, Gary Benson, and Martin Farach. An alphabet independent approach to two-dimensional pattern matching. *SIAM J. Comput.*, 23(2):313–323, 1994. doi:10.1137/S0097539792226321.

[6] Amihood Amir, Gary Benson, and Martin Farach. Optimal two-dimensional compressed matching. *J. Algorithms*, 24(2):354–379, 1997. URL: https://doi.org/10.1006/jagm.1997.0860, doi:10.1006/JAGM.1997.0860.

[7] Amihood Amir and Martin Farach. Efficient 2-dimensional approximate matching of half-rectangular figures. *Inf. Comput.*, 118(1):1–11, 1995. URL: https://doi.org/10.1006/inco.1995.1047, doi:10.1006/INCO.1995.1047.

[8] Amihood Amir and Gad M. Landau. Fast parallel and serial multidimensional approximate array matching. *Theor. Comput. Sci.*, 81(1):97–115, 1991. doi:10.1016/0304-3975(91)90318-V.

[9] Amihood Amir, Gad M. Landau, Shoshana Marcus, and Dina Sokol. Two-dimensional maximal repetitions. *Theor. Comput. Sci.*, 812:49–61, 2020. URL: https://doi.org/10.1016/j.tcs.2019.07.006, doi:10.1016/J.TCS.2019.07.006.

[10] Amihood Amir, Gad M. Landau, and Dina Sokol. Inplace 2d matching in compressed images. *J. Algorithms*, 49(2):240–261, 2003. doi:10.1016/S0196-6774(03)00088-9.

[11] Amihood Amir, Gad M. Landau, and Dina Sokol. Inplace run-length 2d compressed search. *Theor. Comput. Sci.*, 290(3):1361–1383, 2003. doi:10.1016/S0304-3975(02)00041-5.

[12] Amihood Amir, Moshe Lewenstein, and Ely Porat. Faster algorithms for string matching with k mismatches. *J. Algorithms*, 50(2):257–275, 2004. doi:10.1016/S0196-6774(03)00097-X.

[13] Alberto Apostolico and Valentin E. Brimkov. Fibonacci arrays and their two-dimensional repetitions. *Theor. Comput. Sci.*, 237(1-2):263–273, 2000. doi:10.1016/S0304-3975(98)00182-0.

[14] Alberto Apostolico and Valentin E. Brimkov. Optimal discovery of repetitions in 2d. *Discret. Appl. Math.*, 151(1-3):5–20, 2005. URL: https://doi.org/10.1016/j.dam.2005.02.019, doi:10.1016/J.DAM.2005.02.019.

[15] Alberto Apostolico, Laxmi Parida, and Simona E. Rombo. Motif patterns in 2d. *Theor. Comput. Sci.*, 390(1):40–55, 2008. URL: https://doi.org/10.1016/j.tcs.2007.10.019, doi:10.1016/J.TCS.2007.10.019.

[16] Ricardo A. Baeza-Yates. Similarity in two-dimensional strings. In Wen-Lian Hsu and Ming-Yang Kao, editors, *Computing and Combinatorics, 4th Annual International Conference, COCOON '98, Taipei, Taiwan, R.o.C., August 12-14, 1998, Proceedings*, volume 1449 of *Lecture Notes in Computer Science*, pages 319–328. Springer, 1998. doi:10.1007/3-540-68535-9_36.

[17] Ricardo A. Baeza-Yates and Gonzalo Navarro. Fast two-dimensional approximate pattern matching. In Claudio L. Lucchesi and Arnaldo V. Moura, editors, *LATIN '98: Theoretical Informatics, Third Latin American Symposium, Campinas, Brazil, April, 20-24, 1998, Proceedings*, volume 1380 of *Lecture Notes in Computer Science*, pages 341–351. Springer, 1998. URL: https://doi.org/10.1007/BFb0054334, doi:10.1007/BFB0054334.

[18] Ricardo A. Baeza-Yates and Mireille Régnier. Fast two-dimensional pattern matching. *Inf. Process. Lett.*, 45(1):51–57, 1993. doi:10.1016/0020-0190(93)90250-D.

[19] Theodore P. Baker. A technique for extending rapid exact-match string matching to arrays of more than one dimension. *SIAM J. Comput.*, 7(4):533–541, 1978. doi:10.1137/0207043.

[20] Piotr Berman, Marek Karpinski, Lawrence L. Larmore, Wojciech Plandowski, and Wojciech Rytter. On the complexity of pattern matching for highly compressed two-dimensional texts. *J. Comput. Syst. Sci.*, 65(2):332–350, 2002. URL: https://doi.org/10.1006/jcss.2002.1852, doi:10.1006/JCSS.2002.1852.

[21] Richard S. Bird. Two dimensional pattern matching. *Inf. Process. Lett.*, 6(5):168–170, 1977. doi:10.1016/0020-0190(77)90017-5.

[22] Itai Boneh, Dvir Fried, Shay Golan, Matan Kraus, Adrian Miclaus, and Arseny Shur. Searching 2d-strings for matching frames. *CoRR*, abs/2310.02670, 2023. URL: https://doi.org/10.48550/arXiv.2310.02670, arXiv:2310.02670, doi:10.48550/ARXIV.2310.02670.

[23] Karl Bringmann, Philip Wellnitz, and Marvin Künnemann. Few matches or almost periodicity: Faster pattern matching with mismatches in compressed texts. In Timothy M. Chan, editor, *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 1126–1145. SIAM, 2019. doi:10.1137/1.9781611975482.69.

[24] Timothy M. Chan, Shay Golan, Tomasz Kociumaka, Tsvi Kopelowitz, and Ely Porat. Approximating text-to-pattern hamming distances. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pages 643–656. ACM, 2020. doi:10.1145/3357713.3384266.

[25] Panagiotis Charalampopoulos, Tomasz Kociumaka, and Philip Wellnitz. Faster approximate pattern matching: A unified approach. In Sandy Irani, editor, *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020*, pages 978–989. IEEE, 2020. doi:10.1109/FOCS46700.2020.00095.

[26] Panagiotis Charalampopoulos, Jakub Radoszewski, Wojciech Rytter, Tomasz Walen, and Wiktor Zuba. The number of repetitions in 2d-strings. In Fabrizio Grandoni, Grzegorz Herman, and Peter Sanders, editors, *28th Annual European Symposium on Algorithms, ESA 2020, September 7-9, 2020, Pisa, Italy (Virtual Conference)*, volume 173 of *LIPIcs*, pages 32:1–32:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. URL: https://doi.org/10.4230/LIPIcs.ESA.2020.32, doi:10.4230/LIPICS.ESA.2020.32.

[27] Panagiotis Charalampopoulos, Jakub Radoszewski, Wojciech Rytter, Tomasz Walen, and Wiktor Zuba. Computing covers of 2d-strings. In Pawel Gawrychowski and Tatiana Starikovskaya, editors, *32nd Annual Symposium on Combinatorial Pattern Matching, CPM 2021, July 5-7, 2021, Wrocław, Poland*, volume 191 of *LIPIcs*, pages 12:1–12:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. URL: https://doi.org/10.4230/LIPIcs.CPM.2021.12, doi:10.4230/LIPICS.CPM.2021.12.

[28] Raphaël Clifford, Allyx Fontaine, Ely Porat, Benjamin Sach, and Tatiana Starikovskaya. The k-mismatch problem revisited. *CoRR*, abs/1508.00731, 2015. URL: http://arxiv.org/abs/1508.00731, arXiv:1508.00731, doi:10.48550/arxiv.1508.00731.

[29] Raphaël Clifford, Allyx Fontaine, Ely Porat, Benjamin Sach, and Tatiana Starikovskaya. The k-mismatch problem revisited. In Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 2039–2052. SIAM, 2016. URL: https://doi.org/10.1137/1.9781611974331.ch142, doi:10.1137/1.9781611974331.CH142.

[30] Raphaël Clifford, Allyx Fontaine, Tatiana Starikovskaya, and Hjalte Wedel Vildhøj. Dynamic and approximate pattern matching in 2d. In Shunsuke Inenaga, Kunihiko Sadakane, and Tetsuya Sakai, editors, *String Processing and Information Retrieval - 23rd International Symposium, SPIRE 2016, Beppu, Japan, October 18-20, 2016, Proceedings*, volume 9954 of *Lecture Notes in Computer Science*, pages 133–144, 2016. doi:10.1007/978-3-319-46049-9_13.

[31] Richard Cole, Maxime Crochemore, Zvi Galil, Leszek Gasieniec, Ramesh Hariharan, S. Muthukrishnan, Kunsoo Park, and Wojciech Rytter. Optimally fast parallel algorithms for preprocessing and pattern matching in one and two dimensions. In *34th Annual Symposium on Foundations of Computer Science, Palo Alto, California, USA, 3-5 November 1993*, pages 248–258. IEEE Computer Society, 1993. doi:10.1109/SFCS.1993.366862.

[32] Richard Cole, Zvi Galil, Ramesh Hariharan, S. Muthukrishnan, and Kunsoo Park. Parallel two dimensional witness computation. *Inf. Comput.*, 188(1):20–67, 2004. doi:10.1016/S0890-5401(03)00162-7.

[33] Richard Cole, Carmit Hazay, Moshe Lewenstein, and Dekel Tsur. Two-dimensional parameterized matching. *ACM Trans. Algorithms*, 11(2):12:1–12:30, 2014. doi:10.1145/2650220.

[34] Maxime Crochemore, Leszek Gasieniec, Ramesh Hariharan, S. Muthukrishnan, and Wojciech Rytter. A constant time optimal parallel algorithm for two-dimensional pattern matching. *SIAM J. Comput.*, 27(3):668–681, 1998. doi:10.1137/S0097539795280068.

[35] Maxime Crochemore, Leszek Gasieniec, Wojciech Plandowski, and Wojciech Rytter. Two-dimensional pattern matching in linear time and small space. In Ernst W. Mayr and Claude Puech, editors, *STACS 95, 12th Annual Symposium on Theoretical Aspects of Computer Science, Munich, Germany, March 2-4, 1995, Proceedings*, volume 900 of *Lecture Notes in Computer Science*, pages 181–192. Springer, 1995. doi:10.1007/3-540-59042-0_72.

[36] Maxime Crochemore, Costas S. Iliopoulos, and Maureen Korda. Two-dimensional prefix string matching and covering on square matrices. *Algorithmica*, 20(4):353–373, 1998. doi:10.1007/PL00009200.

[37] N. J. Fine and H. S. Wilf. Uniqueness theorems for periodic functions. *Proceedings of the American Mathematical Society*, 16(1):109–114, 1965. doi:10.1090/s0002-9939-1965-0174934-9.

[38] Kimmo Fredriksson, Gonzalo Navarro, and Esko Ukkonen. Optimal exact and fast approximate two dimensional pattern matching allowing rotations. In Alberto Apostolico and Masayuki Takeda, editors, *Combinatorial Pattern Matching, 13th Annual Symposium, CPM 2002, Fukuoka, Japan, July 3-5, 2002, Proceedings*, volume 2373 of *Lecture Notes in Computer Science*, pages 235–248. Springer, 2002. doi:10.1007/3-540-45452-7_20.

[39] Zvi Galil and Raffaele Giancarlo. Improved string matching with k mismatches. *SIGACT News*, 17(4):52–54, 1986. doi:10.1145/8307.8309.

[40] Zvi Galil and Kunsoo Park. Alphabet-independent two-dimensional witness computation. *SIAM J. Comput.*, 25(5):907–935, 1996. doi:10.1137/S0097539792241941.

[41] Guilhem Gamard and Gwénaël Richomme. Coverability and multi-scale coverability on infinite pictures. *J. Comput. Syst. Sci.*, 104:258–277, 2019. URL: https://doi.org/10.1016/j.jcss.2017.05.001, doi:10.1016/J.JCSS.2017.05.001.

[42] Guilhem Gamard, Gwénaël Richomme, Jeffrey O. Shallit, and Taylor J. Smith. Periodicity in rectangular arrays. *Inf. Process. Lett.*, 118:58–63, 2017. URL: https://doi.org/10.1016/j.ipl.2016.09.011, doi:10.1016/J.IPL.2016.09.011.

[43] Pawel Gawrychowski, Samah Ghazawi, and Gad M. Landau. Lower bounds for the number of repetitions in 2d strings. In Thierry Lecroq and Hélène Touzet, editors, *String Processing and Information Retrieval - 28th International Symposium, SPIRE 2021, Lille, France, October 4-6, 2021, Proceedings*, volume 12944 of *Lecture Notes in Computer Science*, pages 179–192. Springer, 2021. doi:10.1007/978-3-030-86692-1_15.

[44] Pawel Gawrychowski and Przemyslaw Uznanski. Optimal trade-offs for pattern matching with k mismatches. *CoRR*, abs/1704.01311, 2017. URL: http://arxiv.org/abs/1704.01311, arXiv:1704.01311, doi:10.48550/arxiv.1704.01311.

[45] Pawel Gawrychowski and Przemyslaw Uznanski. Towards unified approximate pattern matching for hamming and l_1 distance. In Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella, editors, *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*, volume 107 of *LIPIcs*, pages 62:1–62:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. URL: https://doi.org/10.4230/LIPIcs.ICALP.2018.62, doi:10.4230/LIPICS.ICALP.2018.62.

[46] Juha Kärkkäinen and Esko Ukkonen. Two- and higher-dimensional pattern matching in optimal expected time. *SIAM J. Comput.*, 29(2):571–589, 1999. doi:10.1137/S0097539794275872.

[47] Howard J. Karloff. Fast algorithms for approximately counting mismatches. *Inf. Process. Lett.*, 48(2):53–60, 1993. doi:10.1016/0020-0190(93)90177-B.

[48] Richard M. Karp, Raymond E. Miller, and Arnold L. Rosenberg. Rapid identification of repeated patterns in strings, trees and arrays. In Patrick C. Fischer, H. Paul Zeiger, Jeffrey D. Ullman, and Arnold L. Rosenberg, editors, *Proceedings of the 4th Annual ACM Symposium on Theory of Computing, May 1-3, 1972, Denver, Colorado, USA*, pages 125–136. ACM, 1972. doi:10.1145/800152.804905.

[49] Richard M. Karp and Michael O. Rabin. Efficient randomized pattern-matching algorithms. *IBM J. Res. Dev.*, 31(2):249–260, 1987. URL: https://doi.org/10.1147/rd.312.0249, doi:10.1147/RD.312.0249.

[50] Donald E. Knuth, James H. Morris Jr., and Vaughan R. Pratt. Fast pattern matching in strings. *SIAM J. Comput.*, 6(2):323–350, 1977. doi:10.1137/0206024.

[51] Kamala Krithivasan and R. Sitalakshmi. Efficient two-dimensional pattern matching in the presence of errors. *Inf. Sci.*, 43(3):169–184, 1987. doi:10.1016/0020-0255(87)90037-5.

[52] Gad M. Landau and Uzi Vishkin. Efficient string matching with k mismatches. *Theor. Comput. Sci.*, 43:239–249, 1986. doi:10.1016/0304-3975(86)90178-7.

[53] R. C. Lyndon and M. P. Schützenberger. The equation $a^m = b^n c^p$ in a free group. *Michigan Mathematical Journal*, 9(4), December 1962. doi:10.1307/mmj/1028998766.

[54] Kalpana Mahalingam and Palak Pandoh. On the maximum number of distinct palindromic sub-arrays. In Carlos Martín-Vide, Alexander Okhotin, and Dana Shapira, editors, *Language and Automata Theory and Applications - 13th International Conference, LATA 2019, St. Petersburg, Russia, March 26-29, 2019, Proceedings*, volume 11417 of *Lecture Notes in Computer Science*, pages 434–446. Springer, 2019. doi:10.1007/978-3-030-13435-8_32.

[55] Shoshana Marcus and Dina Sokol. 2d lyndon words and applications. *Algorithmica*, 77(1):116–133, 2017. URL: https://doi.org/10.1007/s00453-015-0065-z, doi:10.1007/S00453-015-0065-Z.

[56] Filippo Mignosi, Antonio Restivo, and Pedro V. Silva. On fine and wilf's theorem for bidimensional words. *Theor. Comput. Sci.*, 292(1):245–262, 2003. doi:10.1016/S0304-3975(01)00226-2.

[57] Shoshana Neuburger and Dina Sokol. Succinct 2d dictionary matching. *Algorithmica*, 65(3):662–684, 2013. URL: https://doi.org/10.1007/s00453-012-9615-9, doi:10.1007/S00453-012-9615-9.

[58] Kunsoo Park. Analysis of two-dimensional approximate pattern matching algorithms. *Theor. Comput. Sci.*, 201(1-2):263–273, 1998. doi:10.1016/S0304-3975(97)00277-6.

[59] Jakub Radoszewski, Wojciech Rytter, Juliusz Straszynski, Tomasz Walen, and Wiktor Zuba. Rectangular tile covers of 2d-strings. In Hideo Bannai and Jan Holub, editors, *33rd Annual Symposium on Combinatorial Pattern Matching, CPM 2022, June 27-29, 2022, Prague, Czech Republic*, volume 223 of *LIPIcs*, pages 23:1–23:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. URL: https://doi.org/10.4230/LIPIcs.CPM.2022.23, doi:10.4230/LIPICS.CPM.2022.23.

[60] Sanjay Ranka and Todd Heywood. Two-dimensional pattern matching with k mismatches. *Pattern Recognit.*, 24(1):31–40, 1991. doi:10.1016/0031-3203(91)90114-K.

[61] Wojciech Rytter. Compressed and fully compressed pattern matching in one and two dimensions. *Proc. IEEE*, 88(11):1769–1778, 2000. doi:10.1109/5.892712.

[62] Jorma Tarhio. A sublinear algorithm for two-dimensional string matching. *Pattern Recognit. Lett.*, 17(8):833–838, 1996. doi:10.1016/0167-8655(96)00055-4.

[63] Rui Feng Zhu, Masayuki Nakajima, and Takeshi Agui. The extension of the aho-corasick algorithm to multiple rectangular patterns of different sizes and n-dimensional patterns and text. In *Proceedings of IAPR Workshop on Computer Vision - Special Hardware and Industrial Applications, MVA 1988, Tokyo, Japan, October 12-14, 1988*, pages 185–188, 1988. URL: http://www.mva-org.jp/Proceedings/CommemorativeDVD/1988/papers/1988185.pdf.

[64] Rui Feng Zhu and Tadao Takaoka. A technique for two-dimensional pattern matching. *Commun. ACM*, 32(9):1110–1120, 1989. doi:10.1145/66451.66459.