Hardness of Optimal Spaced Seed Design ⋆

François Nicolas, Eric Rivals *,1

L.I.R.M.M., U.M.R. 5506
C.N.R.S. – Université de Montpellier II
161, rue Ada
34392 Montpellier Cedex 5
France

Correspondence address: Eric Rivals
L.I.R.M.M., U.M.R. 5506
C.N.R.S. – Université de Montpellier II
161, rue Ada
34392 Montpellier Cedex 5
France
Tel: (33) 4 67 41 86 64
Fax: (33) 4 67 41 85 00

Abstract

Speeding up approximate pattern matching is a line of research in stringology since the 80's. Practically fast approaches belong to the class of filtration algorithms, in which text regions dissimilar to the pattern are first excluded, and the remaining regions are then compared to the pattern by dynamic programming. Among the conditions used to test similarity between the regions and the pattern, many require a minimum number of common substrings between them. When only substitutions are taken into account for measuring dissimilarity, counting spaced subwords instead of substrings improves the filtration efficiency. However, a preprocessing step is required to design one or more patterns, called spaced seeds (or gapped seeds), for the subwords, depending on the search parameters. Two distinct lines of research appear the literature: one with probabilistic formulations of seed design problems, in which one wishes for instance to compute a seed with the highest probability to detect the desired similarities (lossy filtration), a second line with combinatorial formulations, where the goal is to find a seed that detects all or a maximum number of similarities (both lossless and lossy filtration). We concentrate on combinatorial seed design problems and consider formulations in which the set of sought similarities is either listed explicitly (RSOS), or characterised by their length and maximal number of mismatches (Non Detection). Several articles exhibit exponential algorithms for these problems. In this work, we provide hardness and inapproximability results for several seed design problems, thereby justifying the complexity of known algorithms. Moreover, we introduce a new formulation of seed design (MWLS), in which the weight of the seed has to be maximised, and show it is as difficult to approximate as Maximum Independent Set.

Key words: sequence comparison; alignment; string matching; filtration; spaced seed; gapped seed; maximum independent set; Golomb ruler; tiling; maximum coverage; approximability

⋆ An extended abstract containing some of the results presented in this paper, without all proofs, has been published in the proceedings of the 16th Annual Symposium on Combinatorial Pattern Matching (CPM'05) [1].

* Corresponding author.
Email address: rivals@lirmm.fr (Eric Rivals).
URL: http://www.lirmm.fr/~rivals (Eric Rivals).

[1] Both authors are supported by grants from the ACI IMPBio project "REPEVOL" http://www.lirmm.fr/~rivals/RESEARCH/REPEVOL.

1 Introduction

1.1 Context

A routine task in computational genomics is to search among all known sequences those being similar to a sequence of interest. "Similar" means "that can be aligned over reasonably long portions". The similarity in sequence helps in the annotation of the sequence of interest as it may reveal, *e.g.*, if it is a gene, a similarity in function, in regulation, in its interaction with other molecules, in three dimensional structure of the protein product, or a common evolutionary origin. This task is known as sequence similarity search.

1.1.1 Filtration

Since the 90's, heuristic algorithms [2] are preferred to the direct application of dynamic programming schemes, which require quadratic time. In practise, as the sizes of the sequence databases grow exponentially, efficiency is achieved by *filtration*. The underlying principle of filtration is to exclude in a first step regions of the sequence database that are not similar to the query sequence by testing a simple condition. The second step, or *verification* step, performs an alignment procedure by dynamic programming with the few remaining regions. Whenever the filter condition is a necessary condition, no potential match is excluded and one achieves *lossless filtration*. Otherwise some true matches may be missed and this is called *lossy filtration*. Note that in the latter, even an exact dynamic programming algorithm used for the verification step cannot recover the missed similarities.

Application of lossy filtration to sequence similarity searching occurs in software like BLAST [2], which originally requires one common substring of a fixed length between the two sequences, FLASH [3], which uses randomly chosen spaced seeds to index the database, or QUASAR [4], which eliminates candidate sequences sharing less than a threshold number of common substrings depending on the user-defined substring length.

This ubiquitous idea of filtration was developed for the problem of exact string matching by using hash tables (lossy filtration) [5] and for approximate pattern matching, using various necessary conditions, see for instance [6], to derive sub-linear expected time algorithms [7] (lossless filtration).

1.1.2 Seed design

Usual conditions rely on counting common contiguous subwords to the query and the database sequence. Recently, several authors have emphasised that the shape of the subwords plays a key role in filtration efficiency, and proposed to use "carefully chosen arbitrary shapes" for the subwords [8–12]. The shape of the subwords is given by a spaced seed, *e.g.* a pattern like `##-##--#` where the `#` symbol indicates which position should match between the query's and database sequence's subword, and the `-` are "don't care" positions. The *weight* of a seed is its number of `#`. The key question is to choose such a seed to optimise the filtration efficiency. In theory and in practise, the specificity increases with the weight, while the sensitivity varies greatly with the positions of the jokers. Given a set of similarities (alignments) of interest, the goal is to find the best seed or a family of seeds. (The latter is known as *multiple seed* design). Two axis of research have appeared in the literature. The first one promotes probabilistic formulations of seed design problems, in which the computed seed should maximise the probability of detecting or "hitting" a similarity (the so-called *hit probability*) without sacrificing specificity. The second axis explores combinatorial formulations of the problem, where one seeks a seed that detects all given similarities. Probabilistic formulations deal with lossy filtration.

The probabilistic approach. Following the line of [10], probabilistic formulation of seed design has been thoroughly investigated (among others [13–17]). Some approaches compute seeds specialised for coding regions detection [14], other compare spaced seeds with contiguous seeds, or propose methods to evaluate their sensitivity with different models of alignment [13,15–17]. For instance, if the matches occur independently and with the same probability, it was shown that computing the hit probability of a given spaced seed is NP-hard, but can be approximated (admits a PTAS) [10,18,19]. Recently, some authors seek to design seeds that accommodate not only mismatches, but also indels, to search for similarities in non-coding genomic regions [20].

Noteworthy is the existence of at least two heuristic or exact algorithms to determine a good or an optimal seed for a given search problem, see for instance the Mandala [16] and Hedera [21] programs.

The combinatorial approach. In this work we concentrate on combinatorial seed design, which has also been addressed by several groups [10,8,12,22,21,1]. We consider formulations in which the set of sought similarities is either listed explicitly (RSOS, MWLS), or characterised by their length and maximal number of mismatches (Non Detection). These problems are either decision or maximisation problems. In two formulations (Non

DETECTION, MWLS), one searches for a seed that detects all similarities, while in the third one (RSOS) one seeks a seed of maximal sensitivity for a given specificity. Several algorithms whose complexities depend exponentially on the length of the seed have been proposed to solve these combinatorial problems [8,12,21], but it is not known to which complexity class the simplest forms of the seed design problem belong. Our article answer these questions by showing these problems are NP-hard, or even worse, difficult to approximate.

Results. In this paper, we consider both lossy and lossless filtrations. In [10], the authors show the NP-hardness and the inapproximability of a region specific multiple seeds design problem, where both the set of similarities and the weight of the seeds are constrained (see problem RSOS below). We improve on these results by showing the inapproximability of RSOS even in the case of a single seed (Section 4). Moreover, we prove the hardness of a general seed design problem: NON DETECTION as defined in [12] (Section 2). In this problem, one considers the set of all similarities at a given Hamming distance from the query. As by-product of our proof, we introduce and classify a tiling problem (SSC). Several works [8,12,22] give empirical and theoretical evidences that support the correlation between the weight of the seed and filtration efficiency. Building on this idea, we propose an optimisation problem MWLS in which the weight of the designed seed has to be maximised. We provide a proof of NP-hardness and of inapproximability for MWLS (Section 3).

Organisation of the paper. In the remaining of this section, we introduce a notation, define the investigated problems, and survey known results. Sections 2, 3, and 4 are each dedicated to a problem listed above: NON DETECTION, MWLS, and RSOS, respectively. Section 2 is independent of Sections 3 and 4. In conclusion, we list open questions concerning combinatorial seed design, as well as tiling problems.

1.2 Notations

Throughout this paper, ϵ denotes an arbitrarily small positive real number, and $e \approx 2.71828183$ denotes the base of the natural logarithm. The set of all integers is denoted by $\mathbb{Z}$. For every a, $b \in \mathbb{Z}$, $[a, b]$ denotes the discrete line segment with endpoints a and b, that is the set of all $n \in \mathbb{Z}$ satisfying $a \leq n \leq b$. For every finite set X, $\#X$ denotes the *cardinality* of X. For every set Y and every integer $q \geq 0$, a *q-subset* of Y is any subset of Y with cardinality q. For every subsets X and Y of $\mathbb{Z}$, $X + Y$ denotes the set $\{x + y : (x, y) \in X \times Y\}$.

1.2.1 Words

In this section, basic notions from combinatorics on words are recalled.

An *alphabet* Σ is a finite set of *symbols*, also called its *letters*. A *word* or *string* over Σ is a finite sequence of elements of Σ. The set of all words over Σ is denoted by $\Sigma^\star$. Word *concatenation* is denoted multiplicatively. For any word w over Σ, the *length* of w is denoted by $|w|$; for every index $i \in [1, |w|]$, $w[i]$ denotes the i^{th} letter of w: $w = w[1]w[2]\cdots w[|w|]$.

Let t and w be two words over Σ. Given an index $i \in [0, |t| - |w|]$, we say that w *occurs in* t *at position* i if, for every index $j \in [1, |w|]$, $w[j] = t[i + j]$; w is called a *substring* of t whenever there exists $i \in [0, |t| - |w|]$ such that w occurs in t at position i. According to the previous definition, the letter $t[i]$ occurs in t at position $i - 1$ for every $i \in [1, |t|]$. For every $a \in \Sigma$, $|t|_a := \#\{i \in [1, |t|] : t[i] = a\}$ denotes the number of occurrences of letter a in t.

Note that all words involved in the sequel are elements of either $\{\mathtt{0}, \mathtt{1}\}^\star$ or $\{\mathtt{\#}, \mathtt{-}\}^\star$.

1.2.2 Seeds and similarities

Definition 1 (Weight, seed) *The* weight *of a word* $w \in \{\mathtt{\#}, \mathtt{-}\}^\star$*, denoted by* $\|w\|$*, is the number of occurrences of the letter* `#` *in* w*:* $\|w\| = |w|_{\mathtt{\#}}$*. A* seed *is a non-empty word over* $\{\mathtt{\#}, \mathtt{-}\}$ *whose first and last letters are* `#`*'s (*i.e.*, an element of* $\mathtt{\#}\{\mathtt{\#}, \mathtt{-}\}^\star\mathtt{\#} \cup \{\mathtt{\#}\}$*).*

Definition 2 (Similarity) *A* similarity *is a word over* $\{\mathtt{0}, \mathtt{1}\}$*. Let* m *and* k *be two integers such that* $0 \leq k \leq m$*. An* (m, k)-similarity *is a similarity of length* m *with* k *occurrences of the symbol* `0` *and* $m - k$ *occurrences of the symbol* `1` *(*i.e.*, an element of* $\{s \in \{\mathtt{0}, \mathtt{1}\}^m : |s|_{\mathtt{0}} = k\}$*).*

Definition 3 (Detection) *Let* w *be a word over* $\{\mathtt{\#}, \mathtt{-}\}$ *and let* s *be a similarity. Let* i *be an index in* $[0, |s| - |w|]$*. We say that* w detects s at position i *if, for every index* $j \in [1, |w|]$*,* $w[j] = \mathtt{\#}$ *implies* $s[i + j] = \mathtt{1}$*.*

We say that w detects s *whenever there exists* $i \in [0, |s| - |w|]$ *such that* w *detects* s *at position* i*.*

Note that the previous definition applies in particular if w is a seed.

Example 4 *The word* $g :=$ `#-##--#-##` *is a seed with weight* 6*, the word* $s :=$ `101101101101100` *is a* $(15, 6)$*-similarity, and* g *detects* s *at positions* 0 *and* 3*.*

Remark 5 summarises elementary properties related to similarity detection.

Remark 5 *Let w be a word over $\{\mathtt{\#}, \mathtt{-}\}$ and let s be a similarity.*

(*i*). *If w detects s then w is not longer than s.*
(*ii*). *If w and s have the same length then w may only detect s at position 0.*
(*iii*). *If w detects s then any substring of w detects s.*
(*iv*). *If w detects a substring of s then w detects s.*
(*v*). *For every integer $m \geq |w|$, w detects $\mathtt{1}^m$.*

1.3 Problems

Our aim is to study the computational complexity of three problems.

In Section 2, we consider the decision problem [12]:

Name: Non Detection
Instance: A seed g, two integers m and k satisfying $0 \leq k \leq m$.
Question: Does there exist an (m, k)-similarity that is not detected by g?

We show that Non Detection is NP-complete in Theorem 15. Note that we assume that any instance (g, m, k) has size $\Theta(|g| + m)$, *i.e.*, that the integers m and k are encoded in unary. If encoded in binary (as usual), (g, m, k) would have size only $\Theta(|g| + \log m)$. In other words, we demonstrate that Non Detection is *strongly* NP-complete.

In Section 3, we investigate the difficulty to approximate the maximisation problem:

Name: region specific Maximum Weight Lossless Seed (MWLS)
Instance: A finite set S of similarities.
Solution: A seed g that detects all similarities in S.
Measure: The weight of g.

Theorem 25 proves there does not exist any polynomial-time approximation algorithm for MWLS with bound $(\#S)^{0.5-\epsilon}$, unless P = NP.

In Section 4, we study the maximisation problem [10]:

Name: Region Specific Optimal Seed (RSOS)
Instance: An integer $\varpi \geq 1$ and a finite set S of similarities.
Solution: A seed g of weight ϖ.
Measure: The number of similarities in S detected by g.

Theorem 28 shows that RSOS is not approximable within bound $\frac{e}{e-1} - \epsilon$, unless P = NP.

1.4 Related works

1.4.1 Concerning Non Detection

Let m and k be two integers with $0 \leq k \leq m$, and let Γ be a set of seeds.

- Denote by $\mathrm{U}(\Gamma, m, k)$ the number of (m, k)-similarities left undetected by all seeds in Γ.
- Denote by $\mathrm{T}(\Gamma, m, k)$ the largest integer $t \geq 0$ satisfying: for every (m, k)-similarity s, there exist t distinct ordered pairs (g_1, i_1), (g_2, i_2), $\ldots$, (g_t, i_t) such that, for any $j \in [1, t]$, $g_j \in \Gamma$, $i_j \in [0, |s| - |g_j|]$, and g_j detects s at position i_j.

Informally, $\mathrm{T}(\Gamma, m, k)$ is the minimal number of positions at which any (m, k)-similarity is detected by the seeds in Γ.

In [12], one finds dynamic programming algorithms to compute $\mathrm{U}(\Gamma, m, k)$ and $\mathrm{T}(\Gamma, m, k)$ in time proportional to

$$m \times \sum_{j=0}^{k} \binom{\lambda}{j}(k - j + 1) + (\#\Gamma) \times \sum_{j=0}^{k} \binom{\lambda}{j} \quad \text{with} \quad \lambda := \max_{g \in \Gamma} |g| \ .$$

A simple bound [8] guarantees that these algorithms have complexities $O\Big(2^{\lambda} \times (mk + \#\Gamma)\Big)$, and are thus Fixed Parameter Tractable (FPT) for parameter λ (see [23] for details on parameterised complexity). The algorithm that computes $\mathrm{T}(\Gamma, m, k)$ described in [12] generalises to a family of seeds the one for the case of a single seed given in [8, Section 4].

Solving Non Detection on an instance (g, m, k) means to decide whether $\mathrm{U}(\{g\}, m, k)$ is non-zero, resp. whether $\mathrm{T}(\{g\}, m, k)$ equals zero. Hence, Theorem 15 implies that, even if we restrict ourselves to the case of a single seed ($\#\Gamma = 1$), any algorithm computing $\mathrm{U}(\Gamma, m, k)$, resp. $\mathrm{T}(\Gamma, m, k)$, requires in the worst case exponential time. Thus, the algorithms given in [8,12] have the best time complexities one can hope.

1.4.2 Concerning Region Specific Optimal Seed

The authors of [10] consider the following more general version of RSOS:

Name: Region Specific Optimal Seeds (RSOSs)

Instance: Three integers d, ϖ, and ℓ. A finite set S of similarities.
Solution: A set Γ of d seeds, each of weight ϖ and of length at most ℓ.
Measure: The number of similarities in S detected by at least one seed in Γ.

RSOS is the variation of RSOS$\underline{\text{S}}$ where one seeks a single (instead of several) seed and where the length of the seed is unconstrained. Formally, RSOS can be seen as the restriction of RSOS$\underline{\text{S}}$ to the instances (d, ϖ, ℓ, S) satisfying $d = 1$ and $\ell \geq \max_{s \in S} |s|$ (see point (i) of Remark 5). RSOS is thus a simpler version of RSOS$\underline{\text{S}}$. The following two results are shown in [10].

- RSOS$\underline{\text{S}}$ is not approximable within bound $\frac{e}{e-1} - \epsilon$, unless P = NP.
- The restriction of the decision version of RSOS$\underline{\text{S}}$ to instances (d, ϖ, ℓ, S) satisfying $\ell \geq \max_{s \in S} |s|$ is NP-hard.

Theorem 28 improves on both of these results. Indeed, it states for RSOS an inapproximation lower bound larger than 1 under the condition that P $\neq$ NP, which implies also its NP-hardness [24].

2 Hardness of Non Detection

To show the hardness of Non Detection, we introduce an auxiliary problem:

Name: Soapy Set Cover (SSC)
Instance: A finite subset $G \subseteq \mathbb{Z}$, two non-negative integers N and q.
Question: Does there exist a q-subset $T \subseteq \mathbb{Z}$ such that $G + T$ contains at least N consecutive integers?

It is related to tiling problems (see Section 5.1). We assume that any instance (G, N, q) of SSC has size $\Theta\left(\sum_{\gamma \in G} |\gamma| + N + q\right)$. In other words, we assume that all input integers, N, q, and the elements of G, are encoded in unary.

First, in the proof of Theorem 6, we reduce a well-known NP-complete problem, namely Exact cover by 3-sets (shortened into X3C hereafter) [25, Problem SP2], to SSC. Then, in the proof of Theorem 15, SSC is reduced to Non Detection.

Recall the definition of X3C. Given a set V, an *exact cover* (also called a *partition*) of V is a set $\mathcal{C}$ of non-empty subsets of V satisfying the following property: for every $v \in V$, there exists exactly one set $E \in \mathcal{C}$ such that $v \in E$.

Name: Exact cover by 3-sets (X3C)
Instance: A finite set V and a set $\mathcal{E}$ of 3-subsets of V.

Question: Is there a subset of $\mathcal{E}$ that is an exact cover of V?

Theorem 6 SSC *is* NP*-complete.*

PROOF. SSC is in NP, since for any yes-instance (G, N, q) of SSC, any subset $T \subseteq [1 - \max G, N - \min G]$ of cardinality at most q satisfying $[1, N] \subseteq G + T$ is a polynomial certificate for SSC on (G, N, q). Let us now reduce X3C to SSC.

Let $(V, \mathcal{E})$ be an instance of X3C. If 3 does not divide the cardinality of V then $(V, \mathcal{E})$ is a no-instance of X3C, which we transform into $(\emptyset, 1, 0)$, a no-instance of SSC. Without loss of generality, we can now assume that V has cardinality $3q$ for some integer q, and that

$$V = [q + 1, 4q]$$

after numbering the elements of V. Let us also number the sets in $\mathcal{E}$: denote by m the cardinality of $\mathcal{E}$ and write $\mathcal{E}$ as $\mathcal{E} = \{E_1, E_2, \ldots, E_m\}$.

Compute

$$N := 2q^2 + 4q$$

and set

$$G := \bigcup_{i=1}^{m} \bigcup_{j=1}^{q} (G_{i,j} + \tau_{i,j}) \,,$$

where for every $i \in [1, m]$ and every $j \in [1, q]$,

$$\begin{aligned} F_j &:= [(j-1)(2q-1) + 4q + 1, j(2q-1) + 4q] \,, \\ G_{i,j} &:= \{j\} \cup E_i \cup F_j \cup \{N - j + 1\} \,, \end{aligned}$$

and

$$\tau_{i,j} := 2N((i-1)q + j - 1) \,.$$

The transformation of $(V, \mathcal{E})$ into the instance (G, N, q) of SSC clearly takes polynomial time. It remains to prove the correctness of the reduction. In order to help understand the proof, we first give a brief overview of the gadgets formally defined above.

Overview of the gadgets. All $G_{i,j}$'s are subsets of $[1, N]$, and the $\tau_{i,j}$'s are the mq multiples of $2N$ comprised between 0 and $2N(mq - 1)$ inclusive. Thus, G is a subset of $[1, 2Nmq]$. Moreover, each of the mq segments of length $2N$ partitioning $[1, 2Nmq]$ (that is to say the segments of the form $[2N(k-1) + 1, 2Nk]$ with $k \in [1, mq]$) contains a unique $G_{i,j} + \tau_{i,j}$ in their left half and no elements of G in their right half.

	5	6	7	8	9	10	11	12	13	14	15	16
E_1	·	·	·	◯	◯	·	·	·	·	·	◯	·
E_2	·	◯	·	·	·	·	·	·	·	◯	·	◯
E_3	·	·	◯	·	·	◯	·	◯	·	·	·	·
E_4	◯	◯	·	·	·	·	·	·	·	◯	·	·
E_5	·	·	·	·	·	·	◯	·	◯	·	·	◯
E_6	◯	·	·	·	·	·	·	◯	·	·	◯	·
V	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯

Table 1
An illustration of an instance of X3C. A natural representation of the integer sets $V = [5, 16]$, $E_1 = \{8, 9, 15\}$, $E_2 = \{6, 14, 16\}$, $E_3 = \{7, 10, 12\}$, $E_4 = \{5, 6, 14\}$, $E_5 = \{11, 13, 16\}$ and $E_6 = \{5, 12, 15\}$. Since $\{E_1, E_3, E_4, E_5\}$ is an exact cover of V, the pair $(V, \{E_1, E_2, E_3, E_4, E_5, E_6\})$ is a yes-instance of X3C. It is also easy to see that $(V, \{E_1, E_2, E_3, E_5, E_6\})$ is a no-instance of X3C.

Description of the structure of a $G_{i,j}$. The segments $[1, q]$, V, F_1, F_2, ..., F_q, and $[N - q + 1, N]$ occur consecutively on the discrete line and they exactly cover $[1, N]$. Define F as $F := F_1 \cup F_2 \cup \cdots \cup F_q = [4q + 1, N - q]$. For any $(i, j) \in [1, m] \times [1, q]$, each of the four segments $[1, q]$, V, F, and $[N - q + 1, N]$ contributes to $G_{i,j}$:

- the singleton $\{j\}$ is included in $[1, q]$,
- the 3-set E_i is included in V,
- the segment F_j is included in F, and
- the singleton $\{N - j + 1\}$ in $[N - q + 1, N]$.

An example of an instance of X3C is shown in Table 1, and the reduction of this instance into an instance of SSC is illustrated in Table 2.

Now, let us prove:

Claim 7 *$(V, \mathcal{E})$ is a yes-instance of* X3C *if and only if (G, N, q) is a yes-instance of* SSC.

2.1 Proof of the "only if part" of Claim 7

Assume that $(V, \mathcal{E})$ is a yes-instance of X3C. Then, there exists a subset $\mathcal{C} \subseteq \mathcal{E}$ that is an exact cover of V: $\mathcal{C}$ has cardinality q, and thus there exist i_1, i_2, ..., $i_q \in [1, m]$ such that $\mathcal{C} = \{E_{i_1}, E_{i_2}, \ldots, E_{i_q}\}$.

	$[1,q]$	$V=[q+1,4q]$	F_1	F_2	F_3	F_4	$[N-q+1,N]$
$G_{1,1}$	○ · · ·	· · · ○○ · · · · ○ ·	○○○○○○○	· · · · · · ·	· · · · · · ·	· · · · · · ·	· · · ○
$G_{1,2}$	· ○ · ·	· · · ○○ · · · · ○ ·	· · · · · · ·	○○○○○○○	· · · · · · ·	· · · · · · ·	· · ○ ·
$G_{1,3}$	· · ○ ·	· · · ○○ · · · · ○ ·	· · · · · · ·	· · · · · · ·	○○○○○○○	· · · · · · ·	· ○ · ·
$G_{1,4}$	· · · ○	· · · ○○ · · · · ○ ·	· · · · · · ·	· · · · · · ·	· · · · · · ·	○○○○○○○	○ · · ·
$G_{2,1}$	○ · · ·	· ○ · · · · · ○ · ○	○○○○○○○	· · · · · · ·	· · · · · · ·	· · · · · · ·	· · · ○
$G_{2,2}$	· ○ · ·	· ○ · · · · · ○ · ○	· · · · · · ·	○○○○○○○	· · · · · · ·	· · · · · · ·	· · ○ ·
$G_{2,3}$	· · ○ ·	· ○ · · · · · ○ · ○	· · · · · · ·	· · · · · · ·	○○○○○○○	· · · · · · ·	· ○ · ·
$G_{2,4}$	· · · ○	· ○ · · · · · ○ · ○	· · · · · · ·	· · · · · · ·	· · · · · · ·	○○○○○○○	○ · · ·
$G_{3,1}$	○ · · ·	· · ○ · ○ · ○ · · ·	○○○○○○○	· · · · · · ·	· · · · · · ·	· · · · · · ·	· · · ○
$G_{3,2}$	· ○ · ·	· · ○ · ○ · ○ · · ·	· · · · · · ·	○○○○○○○	· · · · · · ·	· · · · · · ·	· · ○ ·
$G_{3,3}$	· · ○ ·	· · ○ · ○ · ○ · · ·	· · · · · · ·	· · · · · · ·	○○○○○○○	· · · · · · ·	· ○ · ·
$G_{3,4}$	· · · ○	· · ○ · ○ · ○ · · ·	· · · · · · ·	· · · · · · ·	· · · · · · ·	○○○○○○○	○ · · ·
$G_{4,1}$	○ · · ·	○○ · · · · · ○ · ·	○○○○○○○	· · · · · · ·	· · · · · · ·	· · · · · · ·	· · · ○
$G_{4,2}$	· ○ · ·	○○ · · · · · ○ · ·	· · · · · · ·	○○○○○○○	· · · · · · ·	· · · · · · ·	· · ○ ·
$G_{4,3}$	· · ○ ·	○○ · · · · · ○ · ·	· · · · · · ·	· · · · · · ·	○○○○○○○	· · · · · · ·	· ○ · ·
$G_{4,4}$	· · · ○	○○ · · · · · ○ · ·	· · · · · · ·	· · · · · · ·	· · · · · · ·	○○○○○○○	○ · · ·
$G_{5,1}$	○ · · ·	· · · · · ○ · ○ · · ○	○○○○○○○	· · · · · · ·	· · · · · · ·	· · · · · · ·	· · · ○
$G_{5,2}$	· ○ · ·	· · · · · ○ · ○ · · ○	· · · · · · ·	○○○○○○○	· · · · · · ·	· · · · · · ·	· · ○ ·
$G_{5,3}$	· · ○ ·	· · · · · ○ · ○ · · ○	· · · · · · ·	· · · · · · ·	○○○○○○○	· · · · · · ·	· ○ · ·
$G_{5,4}$	· · · ○	· · · · · ○ · ○ · · ○	· · · · · · ·	· · · · · · ·	· · · · · · ·	○○○○○○○	○ · · ·
$G_{6,1}$	○ · · ·	○ · · · · · ○ · · ○ ·	○○○○○○○	· · · · · · ·	· · · · · · ·	· · · · · · ·	· · · ○
$G_{6,2}$	· ○ · ·	○ · · · · · ○ · · ○ ·	· · · · · · ·	○○○○○○○	· · · · · · ·	· · · · · · ·	· · ○ ·
$G_{6,3}$	· · ○ ·	○ · · · · · ○ · · ○ ·	· · · · · · ·	· · · · · · ·	○○○○○○○	· · · · · · ·	· ○ · ·
$G_{6,4}$	· · · ○	○ · · · · · ○ · · ○ ·	· · · · · · ·	· · · · · · ·	· · · · · · ·	○○○○○○○	○ · · ·

Table 2. An illustration of the reduction from X3C to SSC for the X3C instance given in Table 1. The gadget sets $G_{i,j}$ corresponding to the instance $(V, \{E_1, E_2, E_3, E_4, E_5, E_6\})$ of X3C, where the integers sets V, E_1, E_2, E_4, E_5 and E_6 are as in Table 1: $q = 4$, $N = 48$, $F_1 = [17, 23]$, $F_2 = [24, 30]$, $F_3 = [31, 37]$, $F_4 = [38, 44]$ and $[N - q + 1, N] = [45, 48]$.

The integer set

$$T := \left\{-\tau_{i_1,1}, -\tau_{i_2,2}, \ldots, -\tau_{i_q,q}\right\}$$

has cardinality q, and for every $j \in [1,q]$, one has

$$G_{i_j,j} = (G_{i_j,j} + \tau_{i_j,j}) - \tau_{i_j,j} \subseteq G - \tau_{i_j,j} \subseteq G + T\,.$$

Therefore $G + T$ includes

$$\begin{aligned}\bigcup_{j=1}^{q} G_{i_j,j} &= \bigcup_{j=1}^{q} \{j\} \cup \bigcup_{j=1}^{q} E_{i_j} \cup \bigcup_{j=1}^{q} F_j \cup \bigcup_{j=1}^{q} \{N-j+1\} \\ &= [1,q] \quad \cup \quad V \quad \cup \quad F \quad \cup [N-q+1,N] \;= [1,N]\;.\end{aligned}$$

It follows that (G,N,q) is a yes-instance of SSC.

2.2 Proof of the "if part" of Claim 7

Lemma 8

$$\forall t \in \mathbb{Z} \quad \exists (i,j) \in [1,m] \times [1,q] \qquad (G+t) \cap [1,N] \subseteq G_{i,j} + \tau_{i,j} + t\;.$$

PROOF. Let t be an integer. The set $G + t$ is the union of the sets of the form $G_{i,j} + \tau_{i,j} + t$ with $(i,j) \in [1,m] \times [1,q]$. However, the $\tau_{i,j}$'s are chosen in such a way that the $G_{i,j} + \tau_{i,j}$'s, and thus, the $G_{i,j} + \tau_{i,j} + t$'s, are at least N positions apart from each other. It follows that $[1,N]$ cannot contain elements from two distinct $G_{i,j} + \tau_{i,j} + t$'s. This concludes the proof of Lemma 8. □

Assume that (G,N,q) is a yes-instance of SSC, and let us prove that $(V,\mathcal{E})$ is a yes-instance of X3C.

Lemma 9 *There exist* (i_1,j_1,u_1), (i_2,j_2,u_2), …, $(i_q,j_q,u_q) \in [1,m]\times[1,q]\times\mathbb{Z}$ *such that the sets* $G_{i_1,j_1} + u_1$, $G_{i_2,j_2} + u_2$, …, $G_{i_q,j_q} + u_q$ *are pairwise disjoint subsets of* $[1,N]$.

PROOF. Since (G,N,q) is a yes-instance of SSC, there exist q integers t_1, t_2, …, t_q satisfying

$$[1,N] \subseteq G + \{t_1,t_2,\ldots,t_q\} = \bigcup_{k=1}^{q} (G + t_k)\,.$$

It follows:

$$[1, N] = \bigcup_{k=1}^{q} (G + t_k) \cap [1, N] \; . \tag{1}$$

Moreover, Lemma 8 ensures that, for each $k \in [1, q]$, there exist $i_k \in [1, m]$ and $j_k \in [1, q]$ satisfying

$$(G + t_k) \cap [1, N] \subseteq G_{i_k, j_k} + u_k \tag{2}$$

where u_k is defined as $u_k := \tau_{i_k, j_k} + t_k$.

Combining Equations (1) and (2) yields

$$[1, N] \subseteq \bigcup_{k=1}^{q} (G_{i_k, j_k} + u_k) \, .$$

Besides, for every $(i, j) \in [1, m] \times [1, q]$, $G_{i,j}$ has cardinality $2q+4$. Indeed, $G_{i,j}$ is the disjoint union of the set E_i, whose cardinality is 3, of the segment F_j, whose cardinality equals $2q - 1$, and of two singletons. Hence, $[1, N]$, whose cardinality equals $N = q \times (2q+4)$, is covered by the $G_{i_k, j_k} + u_k$'s, which are at most q and have each cardinality $2q + 4$. The $G_{i_k, j_k} + u_k$'s are thus necessarily pairwise disjoint subsets of $[1, N]$. This concludes the proof of Lemma 9. □

Lemma 10

$$\forall k \in [1, q] \qquad -q < u_k < q \, .$$

PROOF. The integers j and $N - j + 1$ are respectively the smallest and largest elements of $G_{i,j}$. Therefore, for every $k \in [1, q]$, one has:

$$\min(G_{i_k, j_k} + u_k) = j_k + u_k \quad \text{and} \quad \max(G_{i_k, j_k} + u_k) = N - j_k + 1 + u_k \, .$$

As $G_{i_k, j_k} + u_k$ is included in $[1, N]$ by Lemma 9, we obtain

$$1 \leq j_k + u_k \quad \text{and} \quad N - j_k + 1 + u_k \leq N \, ,$$

which implies $1 - j_k \leq u_k \leq j_k - 1$. As j_k is at most q, one gets $1 - q \leq u_k \leq q - 1$. This concludes the proof of Lemma 10. □

Lemma 11

$$\{j_1, j_2, \ldots, j_q\} = [1, q] \; .$$

PROOF. By definition of the j_k's (Lemma 9), $\{j_1, j_2, \ldots, j_q\}$ is a subset of $[1, q]$. Thus, it suffices to show that the q indexes j_1, j_2, …, j_q are pairwise distinct. The proof relies on the following claim:

Claim 12 *If S is a segment of length $2q-1$ and if u is an integer satisfying $-q < u < q$ then the centre of S,* i.e., *$(\max S + \min S)/2$, belongs to $S + u$.*

By way of contradiction, assume there exist two distinct indexes k, $l \in [1, q]$ such that $j_k = j_l$. Lemma 10 ensures $-q < u_k, u_l < q$, and thus, by Claim 12, both $F_{j_k} + u_k$ and $F_{j_l} + u_l$ contain the centre, denote it by c, of $F_{j_k} = F_{j_l}$. Hence, c belongs to both sets $G_{i_k,j_k} + u_k$ and $G_{i_l,j_l} + u_l$: contradiction with Lemma 9. This concludes the proof of Lemma 11. □

Lemma 13

$$\forall k \in [1, q] \qquad u_k = 0\,.$$

PROOF. Lemma 11 allows to renumber the q triples (i_1, j_1, u_1), (i_2, j_2, u_2), $\ldots$, (i_q, j_q, u_q) in such a way that $j_k = k$ for every $k \in [1, q]$. Now, assume that the set $K := \{k \in [1, q] : u_k \neq 0\}$ is non-empty, and set $\kappa := \min K$. The following claim will lead to a contradiction.

Claim 14 *Let S and X be two subsets of $\mathbb{Z}$ with $X \subseteq S$ and such that $\min X = \min S$ and $\max X = \max S$. Then X is the unique translate of X included in S,* i.e., *for every $u \in \mathbb{Z}$, $X + u \subseteq S$ if and only if $u = 0$.*

For any $j \in [1, \kappa - 1]$, none of j and $N - j + 1$ belongs to $G_{i_\kappa,\kappa} + u_\kappa$. Indeed, j and $N - j + 1$ are elements of $G_{i_j,j}$, $G_{i_j,j}$ is equal to $G_{i_j,j} + u_j$ since $j \notin K$, and $G_{i_j,j} + u_j$ has an empty intersection with $G_{i_\kappa,\kappa} + u_\kappa$ by Lemma 9. Hence, $G_{i_\kappa,\kappa} + u_\kappa$, which is a subset of $[1, N]$ by Lemma 9, is in fact a subset of $[\kappa, N - \kappa + 1]$. Now, applying Claim 14 with $X := G_{i_\kappa,\kappa}$ and $S := [\kappa, N - \kappa + 1]$ yields $u_\kappa = 0$, which contradicts $\kappa \in K$. This concludes the proof of Lemma 13. □

Now, $u_1 = u_2 = \cdots = u_q = 0$ (Lemma 13), and thus the q sets $G_{i_1,j_1}, G_{i_2,j_2}, \ldots, G_{i_q,j_q}$ are pairwise disjoint (Lemma 9). It follows that their respective 3-subsets E_{i_1}, E_{i_2}, $\ldots$, E_{i_q} are also pairwise disjoints. Therefore, $\left\{E_{i_1}, E_{i_2}, \ldots, E_{i_q}\right\}$ is an exact cover of V, and $(V, \mathcal{E})$ is a yes-instance of X3C. This concludes the proof of Theorem 6. □

Theorem 15 Non Detection *is* NP*-complete.*

PROOF. Non Detection is in NP, since for any yes-instance (g, m, k) of Non Detection, an (m, k)-similarity left undetected by g is a polynomial certificate for Non Detection on (g, m, k).

Now, reduce SSC to Non Detection in order to apply Theorem 6.

Let (G, N, q) be an instance of SSC. If needed, we may translate G in such a way that $\min G = 0$; from now on we make this assumption: G is a subset of $[0, \max G]$. Compute the word g of length $\max G + 1$ over $\{\texttt{\#}, \texttt{-}\}$ defined by: for every $j \in [1, |g|]$, $g[j] = \texttt{\#}$ if and only if $|g| - j \in G$. As G contains both $|g|-1 = \max G$ and $|g|-|g| = 0 = \min G$, we have $g[1] = g[|g|] = \texttt{\#}$, or in other words, the first and last letters of g are \#'s. Compute $m := N - 1 + |g|$ and $k := \min\{m, q\}$. The transformation of (G, N, q) into the instance (g, m, k) of Non Detection clearly takes polynomial time. It remains to check the correctness of our reduction, that is:

Claim 16 *(G, N, q) is a yes-instance of* SSC *if and only if (g, m, k) is a yes-instance of* Non Detection.

First, remark that $N - 1 = m - |g|$ and thus,

$$[0, N - 1] = [0, m - |g|] \ . \tag{3}$$

2.3 Proof of the "if part" of Claim 16

Assume that (g, m, k) is a yes-instance of Non Detection. Then, there exists an (m, k)-similarity s that is not detected by g.

Let us set $T := \{j \in [1, m] : s[j] = \texttt{0}\} - |g|$. Clearly, T is an integer set with cardinality $|s|_{\texttt{0}} = k \leq q$. Moreover, let i be any element of $[0, N - 1]$. The index i is in $[0, m - |g|]$ by Equation (3), and g does not detect s at position i by hypothesis. Therefore, there exists $j \in [1, |g|]$ satisfying both $g[j] = \texttt{\#}$ and $s[i + j] = \texttt{0}$: one gets $|g| - j \in G$ and $i + j - |g| \in T$. Hence, $i = (|g| - j) + (i + j - |g|)$ is in $G + T$. We have thus shown $[0, N - 1] \subseteq G + T$.

It follows that (G, N, q) is a yes-instance of SSC.

2.4 Proof of the "only if part" of Claim 16

Assume that (G, N, q) is a yes-instance of SSC. Then, there exists a q-subset $T \subseteq \mathbb{Z}$ satisfying $[0, N - 1] \subseteq G + T$.

Let s be the similarity of length m defined by: for every $i \in [1, m]$, $s[i] = \texttt{0}$ if and only if $i \in T + |g|$. We check below that g does not detect s, but first, remark that $|s|_{\texttt{0}} \leq \min\{m, q\} = k$. Indeed, $|s|_{\texttt{0}}$ is bounded both by m, which is the total length of s, and by q, which equals the cardinality of $T + |g|$. However, $|s|_{\texttt{0}}$ may be less than k. (For instance, consider the case of $G = \{0, 2\}$, $N = 4$ and $q = 3$: $g = \texttt{\#-\#}$, $m = 6$, and $k = 3$. For every integer

$t \notin \{0,1\}$, $T := \{0,1,t\}$ is a q-subset of $\mathbb{Z}$ satisfying $[0, N-1] \subseteq G+T$. If t is less than -2 or if t is greater than 3 then $s = \texttt{110011}$, and thus $|s|_{\texttt{0}} = k-1$.) Anyway, s can be transformed into an (m,k)-similarity by replacing enough of its `1`'s by `0`'s.

Consider any index i in $[0, m-|g|]$ and let us show that g does not detect s at position i. According to Equation (3), i is in $[0, N-1]$, and thus there exist $\gamma \in G$ and $t \in T$ such that $i = \gamma + t$. Let us set $j := |g| - \gamma$. On the one hand, $|g| - j = \gamma$ is in G, and thus $g[j] = \texttt{\#}$. On the other hand, $i + j = (\gamma + t) + (|g| - \gamma) = t + |g|$ is in $T + |g|$, and thus $s[i+j] = \texttt{0}$. Hence, g does not detect s at position i.

It follows that (g, m, k) is a yes-instance of Non Detection and this concludes the proof of Theorem 15. □

3 Hardness and inapproximability of MWLS

In order to demonstrate the approximation hardness of MWLS, we reduce Maximum Independent Set to it.

Recall that a *(simple) graph* is an ordered pair $G = (V, \mathcal{E})$, where V is a finite set and where $\mathcal{E}$ is a set of 2-subsets of V. The elements of V are called the *vertices* of G, the ones of $\mathcal{E}$ are called the *edges* of G, and for any edge $E \in \mathcal{E}$, the two vertices belonging to E are called its *endpoints*. An *independent set* (also called a *stable set*) of G is a subset $I \subseteq V$ such that, for every edge $E \in \mathcal{E}$, E is not a subset of I. In other words, I satisfies that no pair of its vertices is joined by an edge of $\mathcal{E}$.

Name: Maximum Independent Set (MIS)
Instance: A graph $G = (V, \mathcal{E})$.
Solution: An independent set I of G.
Measure: The cardinality of I.

Many hardness results have been established so far on the approximability of MIS under various complexity assumptions. In particular, it is known that MIS is not approximable within bound $(\#V)^{1-\epsilon}$, unless P = NP [26].

Our reduction, as well as the reduction of Section 4, rely on Golomb rulers [27]. Formally, a *Golomb ruler* is a subset $R \subseteq \mathbb{Z}$ satisfying: for each integer $d \geq 1$, there exists at most one ordered pair $(\mu, \nu) \in R \times R$ such that $d = \nu - \mu$. The integers belonging to a Golomb ruler are called its *marks*. Informally, a Golomb ruler is a ruler such that no two pairs of distinct marks measure the

same distance.

We first define a specific class of Golomb rulers appropriate for our purposes.

Definition 17 *Let n be a positive integer. For each index $i \in [1, n]$, define $\mu_i^{(n)} := (i-1)n^2 + i^2$, and set $R_n := \left\{\mu_1^{(n)}, \mu_2^{(n)}, \ldots, \mu_n^{(n)}\right\}$.*

It is clear that $1 = \mu_1^{(n)} < \mu_2^{(n)} < \cdots < \mu_i^{(n)} < \cdots < \mu_n^{(n)} = n^3$, and that R_n is computable in a time polynomial in n. For instance, in the case of $n = 5$, we have: $\mu_1^{(5)} = 1$, $\mu_2^{(5)} = 29$, $\mu_3^{(5)} = 59$, $\mu_4^{(5)} = 91$, $\mu_5^{(5)} = 125$, and $R_5 = \{1, 29, 59, 91, 125\}$.

Lemma 18 *For every integer $n \geq 1$, the integer set R_n is a Golomb ruler.*

PROOF. It suffices to check that for any ordered pair (i, j) of indexes with $1 \leq i < j \leq n$, (i, j) can be written as a function of the difference $m := \mu_j^{(n)} - \mu_i^{(n)}$. More precisely, we show the following two equalities:

$$i = \frac{1}{2}\left(\frac{m \bmod n^2}{\lfloor m \,/\, n^2 \rfloor} - \left\lfloor m \,/\, n^2 \right\rfloor\right) \text{ and } j = \frac{1}{2}\left(\frac{m \bmod n^2}{\lfloor m \,/\, n^2 \rfloor} + \left\lfloor m \,/\, n^2 \right\rfloor\right). \quad (4)$$

Set $q := j - i$ and $r := j^2 - i^2$. It is clear that both i and j can be written as functions of q and r:

$$i = \frac{1}{2}\left(\frac{r}{q} - q\right) \text{ and } j = \frac{1}{2}\left(\frac{r}{q} + q\right). \quad (5)$$

Furthermore, q and r clearly satisfy $m = qn^2 + r$ and $0 \leq r \leq n^2 - 1$. Hence, q and r are respectively the quotient and the remainder of the division of m by n^2:

$$q = \left\lfloor m \,/\, n^2 \right\rfloor \text{ and } r = m \bmod n^2. \quad (6)$$

Combining Equations (5) and (6) yields Equation (4). This concludes the proof of Lemma 18. □

The sequel of the paper could be easily adapted to accommodate any choice of function $(n, i) \mapsto \mu_i^{(n)}$ provided that the following two requirements are met:

- the function is polynomial-time computable if the input and output integers are encoded in unary, and
- $\left\{\mu_1^{(n)}, \mu_2^{(n)}, \ldots, \mu_n^{(n)}\right\}$ is a Golomb ruler with an n marks, for every integer $n \geq 1$.

From now on until Theorem 25 on page 21, n denotes a given positive integer.

Gadget overview: The following gadget defines words over $\{\#, -\}$ that encodes marks of a Golomb ruler with symbols # and the remaining positions with symbols -. More precisely, for each subset $X \subseteq [1, n]$, it defines a word $\mathrm{w}_X^{(n)}$ of length n^3 that encodes the marks of R_n having an index in X. From each word $\mathrm{w}_X^{(n)}$, one can derive a unique longest seed, $\mathrm{g}_X^{(n)}$, by deleting the leading and trailing - symbols. Conversely, any subset $X \subseteq [1, n]$ with cardinality at least two is completely determined by any of the two words $\mathrm{w}_X^{(n)}$ and $\mathrm{g}_X^{(n)}$ (see Lemma 21).

Definition 19 (Gadgets) *Let X be a subset of $[1, n]$. Define $\mathrm{w}_X^{(n)}$ as the word over $\{\#, -\}$ satisfying:*

$$\left|\mathrm{w}_X^{(n)}\right| = n^3,\ \left\|\mathrm{w}_X^{(n)}\right\| = \#X,\ \textit{and}\ \mathrm{w}_X^{(n)}[\mu_x^{(n)}] = \#\ \textit{for every}\ x \in X.$$

Denote by $\mathrm{g}_X^{(n)}$ the seed obtained from $\mathrm{w}_X^{(n)}$ by deleting the leading and trailing - symbols.

For instance, in the case of $n = 5$ and $X = \{2, 3, 5\}$, we have: $\mathrm{w}_X^{(n)} = (-)^{28}\#(-)^{29}\#(-)^{65}\#$ and $\mathrm{g}_X^{(n)} = \#(-)^{29}\#(-)^{65}\#$.

Remark 20 *For every $X \subseteq [1, n]$, both $\mathrm{w}_X^{(n)}$ and $\mathrm{g}_X^{(n)}$ have weight $\#X$.*

Next lemma means that the $\mathrm{g}_X^{(n)}$'s with weight at least two and the subsets $X \subseteq [1, n]$ with cardinality at least two are in one-to-one correspondence. It builds on Lemma 18.

Lemma 21 *Let X_1 and X_2 be two subsets of $[1, n]$ with cardinalities at least two. Then, $\mathrm{g}_{X_1}^{(n)} = \mathrm{g}_{X_2}^{(n)}$ if and only if $X_1 = X_2$.*

PROOF. For each $\alpha \in \{1, 2\}$, set $g_\alpha := \mathrm{g}_{X_\alpha}^{(n)}$ and $w_\alpha := \mathrm{w}_{X_\alpha}^{(n)}$.

Note that X_α can be written as a function of w_α:

$$X_\alpha = \left\{x \in [1, n] : w_\alpha[\mu_x^{(n)}] = \#\right\}. \tag{7}$$

Let $p_\alpha \in [0, n^3 - |g_\alpha|]$ be such that

$$w_\alpha = (-)^{p_\alpha} g_\alpha (-)^{n^3 - |g_\alpha| - p_\alpha}. \tag{8}$$

As the first letter of g_α is a #, one has $w_\alpha[p_\alpha + 1] = \#$, and thus there exists $i_\alpha \in X_\alpha$ such that

$$p_\alpha + 1 = \mu_{i_\alpha}^{(n)}. \tag{9}$$

In the same way, $w_\alpha[p_\alpha + |g_\alpha|]$ is a #, as it is the last letter of g_α, and thus $p_\alpha + |g_\alpha| = \mu^{(n)}_{j_\alpha}$ for some $j_\alpha \in X_\alpha$. It follows that

$$\mu^{(n)}_{j_\alpha} - \mu^{(n)}_{i_\alpha} = (p_\alpha + |g_\alpha|) - (p_\alpha + 1) = |g_\alpha| - 1\,.$$

Moreover, g_α is of length at least $\|g_\alpha\| = \#X_\alpha \geq 2$, and thus $\mu^{(n)}_{j_\alpha} - \mu^{(n)}_{i_\alpha}$ is positive.

Assume that $g_1 = g_2$. Then, the two differences $\mu^{(n)}_{j_1} - \mu^{(n)}_{i_1}$ and $\mu^{(n)}_{j_2} - \mu^{(n)}_{i_2}$ are equal and positive. It follows successively: $\mu^{(n)}_{i_1} = \mu^{(n)}_{i_2}$ by Lemma 18, $p_1 = p_2$ by Equation (9), $w_1 = w_2$ by Equation (8), and eventually, $X_1 = X_2$ by Equation (7). This concludes the proof of Lemma 21. □

Definition 22 (Some more gadgets) *Let v be an element of $[1, n]$. Define $\mathrm{t}^{(n)}_v$ as the similarity satisfying:*

$$\left|\mathrm{t}^{(n)}_v\right| = n^3,\ \left|\mathrm{t}^{(n)}_v\right|_{\mathtt{1}} = n - 1,\ \textit{and}\ \mathrm{t}^{(n)}_v[\mu^{(n)}_x] = \mathtt{1}\ \textit{for every}\ x \in [1, n]\ \textit{such that}\ x \neq v.$$

For instance, in the case of $n = 5$ and $v = 2$, we have $\mathrm{t}^{(n)}_v = \mathtt{10^{57}10^{31}10^{33}1}$.

Gadget overview: The gadget above defines the similarities needed for the reduction from MIS to MWLS. Like the words $\mathrm{w}^{(n)}_X$, they also have length n^3 and encode marks of the Golomb ruler R_n (by a symbol 1 instead of a #). For any element v of $[1, n]$, we define a similarity $\mathrm{t}^{(n)}_v$ that encodes over $\{0, 1\}$ all the marks of R_n except the one of index v. In general, $\mathrm{t}^{(n)}_v$ can be obtained from $\mathrm{w}^{(n)}_{[1,n]\setminus\{v\}}$ in the following way: replace each letter # in $\mathrm{w}^{(n)}_{[1,n]\setminus\{v\}}$ by a 1, and each letter - by a 0.

Next claim links the $\mathrm{t}^{(n)}_v$'s and the $\mathrm{w}^{(n)}_X$'s: it means that the words of length n^3 over $\{\#, -\}$ detecting a fixed $\mathrm{t}^{(n)}_v$ are exactly the $\mathrm{w}^{(n)}_X$'s with $v \notin X$.

Claim 23 *For any $v \in [1, n]$ and any word w of length n^3 over $\{\#, -\}$, the following three statements are equivalent:*

(i) w detects $\mathrm{t}^{(n)}_v$,
(ii) w detects $\mathrm{t}^{(n)}_v$ at position 0, and
(iii) $w = \mathrm{w}^{(n)}_X$ for some subset $X \subseteq [1, n]$ satisfying $v \notin X$.

PROOF. Since w and $\mathrm{t}^{(n)}_v$ have the same length, points *(i)* and *(ii)* are equivalent (point *(ii)* of Remark 5). Moreover, it is easy to see that point *(iii)* implies point *(ii)*. Eventually, if w detects $\mathrm{t}^{(n)}_v$ at position 0, then

$X := \left\{x \in [1,n] : w[\mu_x^{(n)}] = \texttt{\#}\right\}$ is such that $w = \mathrm{w}_X^{(n)}$ and $v \notin X$. Indeed, $\mathrm{t}_v^{(n)}[\mu_v^{(n)}] = \texttt{0}$ requires $w[\mu_v^{(n)}] = \texttt{-}$. Hence, point (*ii*) implies point (*iii*). □

Next lemma means that the seeds detecting a fixed $\mathrm{t}_v^{(n)}$ are exactly the $\mathrm{g}_X^{(n)}$'s with $v \notin X$. It is a corollary of the previous claim.

Lemma 24 *Let v be an element of $[1,n]$ and let g be a seed. Then, g detects $\mathrm{t}_v^{(n)}$ if and only if there exists a subset $X \subseteq [1,n]$ such that $v \notin X$ and $g = \mathrm{g}_X^{(n)}$.*

PROOF. Assume that there exists $X \subseteq [1,n]$ such that $v \notin X$ and $g = \mathrm{g}_X^{(n)}$. Then, g is a substring of $\mathrm{w}_X^{(n)}$, and according to Claim 23, $\mathrm{w}_X^{(n)}$ detects $\mathrm{t}_v^{(n)}$. Hence, g also detects $\mathrm{t}_v^{(n)}$ (point (*iii*) of Remark 5).

Conversely, suppose that g detects $\mathrm{t}_v^{(n)}$: there exists $p \in \left[0, \left|\mathrm{t}_v^{(n)}\right| - |g|\right]$ such that g detects $\mathrm{t}_v^{(n)}$ at position p. Then, $w := (\texttt{-})^p g (\texttt{-})^{n^3-|g|-p}$ detects $\mathrm{t}_v^{(n)}$ at position 0. According to Claim 23, w can be written as $w = \mathrm{w}_X^{(n)}$ for some $X \subseteq [1,n]$ satisfying $v \notin X$, and thus $g = \mathrm{g}_X^{(n)}$. □

Theorem 25 *The* MWLS *problem has no polynomial-time approximation algorithm with bound $(\#S)^{0.5-\epsilon}$, unless* P = NP*.*

PROOF. The MIS problem is reduced to MWLS in such a way that suitable approximation properties are preserved.

Let $G = (V, \mathcal{E})$ be a graph. Denote by n the cardinality of V. After numbering the vertices of G, we can assume that $V = [1,n]$. For each edge $E \in \mathcal{E}$, compute the similarity $s_E := \mathrm{t}_{\min E}^{(n)} \texttt{0}^{n^3} \mathrm{t}_{\max E}^{(n)}$ (note that, according to our notation, $\min E$ and $\max E$ are the two endpoints of E). Then, build the set of similarities $S := \{\texttt{1}^{n^3}\} \cup \{s_E : E \in \mathcal{E}\}$. The transformation of the instance G of MIS into the instance S of MWLS clearly takes a time polynomial in n.

Overview of the reduction. The set of similarities includes $\texttt{1}^{n^3}$. Any seed not longer than n^3 detects this similarity (Remark 5), but no seed longer than n^3 can detect it. By this mean, we limit the length of feasible seeds. For each edge $E \in \mathcal{E}$, the similarity s_E is the concatenation of three strings each of length n^3: $\mathrm{t}_{\min E}^{(n)}$, $\texttt{0}^{n^3}$, and $\mathrm{t}_{\max E}^{(n)}$. No seed can detect $\texttt{0}^{n^3}$. Hence, a (non trivial) seed that detects both $\texttt{1}^{n^3}$ and s_E, is at most n^3 long, and must therefore detect either $\mathrm{t}_{\min E}^{(n)}$ or $\mathrm{t}_{\max E}^{(n)}$, since it cannot overlap both of them. It follows that a seed detects both $\texttt{1}^{n^3}$ and s_E if and only if it is of the form $\mathrm{g}_I^{(n)}$, where I is a subset of $[1,n]$ such that E is not fully contained in I (Lemma 24).

Lemma 26 *For any non-empty independent set I of G, there exists a seed of weight $\#I$ that detects all similarities in S.*

PROOF. Clearly, $\mathrm{g}_I^{(n)}$ is a seed of weight $\#I$ detecting $\mathtt{1}^{n^3}$ (point (v) of Remark 5). Moreover, let $E \in \mathcal{E}$ be any edge of G. As I is an independent set of G, some endpoint $v \in E$ is such that $v \notin I$. Hence, according to Lemma 24, $\mathrm{g}_I^{(n)}$ detects $\mathrm{t}_v^{(n)}$, and all the more reason for $\mathrm{g}_I^{(n)}$ to detect its superstring s_E (point (iv) of Remark 5). □

Lemma 27 *For any seed g detecting all similarities in S, there is an independent set I of G with cardinality $\|g\|$. Moreover, I is computable from g and G in polynomial time.*

PROOF. If g has weight 1 then it suffices to choose $I := \{1\}$. Hence, we may assume $\|g\| \geq 2$ throughout the remaining of the proof.

Let $E \in \mathcal{E}$ be an edge of G. Let f_E be a substring of s_E detected by g and with the same length as g. Since g starts and ends with a $\mathtt{\#}$, f_E starts and ends with a $\mathtt{1}$. Moreover, the presence of $\mathtt{1}^{n^3}$ in S implies $|f_E| = |g| \leq n^3$ (point (i) of Remark 5). Hence, the block $\mathtt{0}^{n^3}$ that lies between $\mathrm{t}_{\min E}^{(n)}$ and $\mathrm{t}_{\max E}^{(n)}$ in s_E is at least as long as f_E. This requires f_E to fully occur either in $\mathrm{t}_{\min E}^{(n)}$ or in $\mathrm{t}_{\max E}^{(n)}$: there exists an endpoint $v_E \in E$, such that g detects $\mathrm{t}_{v_E}^{(n)}$. According to Lemma 24, it follows that there exists $X_E \subseteq [1, n]$ such that $g = \mathrm{g}_{X_E}^{(n)}$ and $v_E \notin X_E$.

The vertex set $J := [1, n] \setminus \{v_E : E \in \mathcal{E}\}$ is clearly an independent set of G. Moreover, for every edge $E \in \mathcal{E}$, X_E has cardinality $\|g\|$ (Remark 20) and the weight of g is at least two. Hence, Lemma 21 implies that all sets of the form X_E with $E \in \mathcal{E}$ are equal to a same subset of J. Therefore, J has cardinality at least $\|g\|$, and removing $\#J - \|g\|$ elements from J yields an independent set I whose cardinality is exactly $\|g\|$.

We now check that I is computable from g and G in polynomial time. For each $E \in \mathcal{E}$, compute $\mathrm{t}_{\min E}^{(n)}$ and check whether g detects $\mathrm{t}_{\min E}^{(n)}$ to obtain v_E: if g detects $\mathrm{t}_{\min E}^{(n)}$ then $v_E := \min E$, otherwise $v_E := \max E$. The previous procedure computes the v_E's from g and G in polynomial time. With the v_E's in hand the computations of J and then I easily follow. □

One has $\#S = \#\mathcal{E} + 1 \leq (\#V)^2$; so, if there exists an approximation algorithm for MWLS with bound $(\#S)^{0.5-\epsilon}$ then Lemmas 26 and 27 ensure that it can be transformed into an approximation algorithm for MIS with bound $(\#S)^{0.5-\epsilon} \leq$

$\left((\#V)^2\right)^{0.5-\epsilon} = (\#V)^{1-2\epsilon}$. But, this is possible only if P = NP [26]. This concludes the proof of Theorem 25. □

4 Hardness and inapproximability of RSOS

As in [10], we prove the approximation-hardness result for RSOS by reduction from the MAXIMUM k-COVER problem [28]. However, our reduction is much more complicated than the one in [10] since it works even for a single seed. The MAXIMUM k-COVER problem (also termed MAXIMUM COVERAGE [24]) is:

Name: MAXIMUM k-COVER (MAX k-COV)
Instance: A tuple $(E_1, E_2, \ldots, E_m)$ of finite sets, an integer $k \in [1, m]$.
Solution: Any k-subset $I \subseteq [1, m]$.
Measure: The cardinality of the set union $\bigcup_{i \in I} E_i$.

Whereas a simple greedy algorithm achieves an approximation ratio of $\frac{e}{e-1}$ [24,28], Feige showed that MAX k-COV is NP-hard to approximate within ratio $\frac{e}{e-1} - \epsilon$ [28]. The result is even slightly stronger: there is no polynomial-time approximation algorithm with bound $\frac{e}{e-1} - \epsilon$ for the *evaluation problem* associated with MAX k-COV, unless P = NP [28]. The approximation lower bound for (the evaluation problem associated with) RSOS is derived from the latter result.

Let us clarify the terminology. Let Π be a maximisation problem and let ρ be a real number greater than or equal to one. For each instance X of Π, let $\mathrm{opt}(X)$ denote the maximum measure, over all solutions of Π on X. We say that a polynomial-time algorithm A approximates within ratio ρ the evaluation problem associated with Π, if on any instance X of Π taken as input, A outputs a number comprised between $\frac{1}{\rho}\mathrm{opt}(X)$ and $\mathrm{opt}(X)$ inclusive.

Theorem 28 *The (evaluation problem associated with the)* RSOS *problem has no polynomial-time approximation algorithm with bound* $\frac{e}{e-1} - \epsilon$*, unless* P = NP*.*

PROOF. According to the above-mentioned approximation-hardness result from Feige [28], it suffices to exhibit an optimum-preserving reduction from MAX k-COV to RSOS.

Claim 29 *The* Max k-Cov *problem can be exactly solved in polynomial time on instances* $((E_1, E_2, \ldots, E_m), k)$ *with bounded* $m - k$.

PROOF. The number of k-subsets of $[1, m]$ equals $\binom{m}{k} = \binom{m}{m-k} = O(m^{m-k})$. Hence, if $m - k$ is bounded then the set of all solutions of Max k-Cov on $((E_1, E_2, \ldots, E_m), k)$ can be enumerated in polynomial time. □

Let $((E_1, E_2, \ldots, E_m), k)$ be an instance of Max k-Cov. According to Claim 29, we may assume

$$k \leq m - 2 \tag{10}$$

without loss of generality (the role played by this assumption is clarified below). Let n denote the cardinality of the ground set $E_1 \cup E_2 \cup \cdots \cup E_m$, and let v_1, v_2, …, v_n be an enumeration of its elements:

$$E_1 \cup E_2 \cup \cdots \cup E_m = \{v_1, v_2, \ldots, v_n\}\,.$$

For each $j \in [1, n]$, let F_j denote the set of all indexes $i \in [1, m]$ such that $v_j \in E_i$:

$$\forall i \in [1, m] \quad \forall j \in [1, n] \qquad v_j \in E_i \iff i \in F_j\,. \tag{11}$$

Compute a Golomb ruler R with cardinality $\#F_1 + \#F_2 + \cdots + \#F_n$ such that all marks in R are positive odd numbers. (According to Lemma 18, $2R_{\#F_1+\#F_2+\cdots+\#F_n} - 1$ is a suitable choice for R.) Then, partition R in the following way: compute n pairwise disjoint subsets Q_1, Q_2, …, $Q_n \subseteq R$ such that Q_j has the same cardinality as F_j for each $j \in [1, n]$.

For each $j \in [1, n]$, compute a similarity $s_j := s_{j,1}s_{j,2}\cdots s_{j,l_j}$, where l_j denotes the greatest element of Q_j and where the substrings $s_{j,1}$, $s_{j,2}$, …, s_{j,l_j} satisfy:

- for every $h \in [1, l_j] \setminus Q_j$, $s_{j,h} = \mathtt{0}^{m^3}$, and
- as h takes all values in Q_j, the set of $s_{j,h}$ is the set of all similarities of the form $\mathrm{t}_i^{(m)}$ with $i \in F_j$. (Remind that $\mathrm{t}_i^{(m)}$ is defined in Definition 22 on page 20.)

Compute $\varpi := m - k$ and $S := \{s_1, s_2, \ldots, s_n\}$. It is easy to see that (ϖ, S) is computable from $((E_1, E_2, \ldots, E_m), k)$ in polynomial time. Moreover, according to Equation (10), ϖ is a positive integer, and thus (ϖ, S) is an instance of RSOS.

Table 3 summarises the reduction. Note that the last two lines of the table sketch the correspondence between the solutions of RSOS and those of Max

MAX k-COV	RSOS
instance $((F_1, F_2, \ldots, F_n), k)$	instance (ϖ, S)
element v_j	similarity s_j
v_j is an element of E_i	$\mathrm{t}_i^{(m)}$ is a substring of s_j
v_j is an element of $\bigcup_{i \in I} E_i$	the seed $\mathrm{g}_{[1,m]\setminus I}^{(m)}$ detects s_j
I is a k-subset of $[1, m]$	$\mathrm{g}_{[1,m]\setminus I}^{(m)}$ is seed with weight ϖ

Table 3
Informal presentation of the reduction from MAX k-COV to RSOS.

k-COV. They are clarified in the next section.

4.2 Correctness of our reduction

We have to prove that the measure of an optimum solution of MAX k-COV on $((E_1, E_2, \ldots, E_m), k)$ equals the measure of an optimum solution of RSOS on (ϖ, S).

Let I be a k-subset of $[1, m]$. The measure of I as a solution of MAX k-COV on $((E_1, E_2, \ldots, E_m), k)$ equals the number of indexes $j \in [1, n]$ such that $v_j \in \bigcup_{i \in I} E_i$. Let g be a seed with weight ϖ. The measure of g as a solution of RSOS on (ϖ, S) equals the number of indexes $j \in [1, n]$ such that g detects s_j. Indeed:

Claim 30 *The n similarities s_1, s_2, …, s_n are pairwise distinct.*

PROOF. For any $j \in [1, n]$, $|s_{j,1}| = |s_{j,2}| = \cdots = \left|s_{j,l_j}\right| = m^3$, and thus s_j has length $m^3 l_j$. Besides, the integers l_1, l_2, …, l_n are pairwise distinct since they belong to the pairwise disjoints sets Q_1, Q_2, …, Q_n, respectively. Therefore, the s_j's have pairwise distinct lengths. □

Now, to prove Theorem 28, it suffices to check that the following two assertions are equivalent for any non-empty subset $J \subseteq [1, n]$:

(A) there exists a k-subset $I \subseteq [1, m]$ such that $v_j \in \bigcup_{i \in I} E_i$ for every $j \in J$, and
(B) there exists a seed g with weight ϖ such that g detects similarity s_j for every $j \in J$.

4.2.1 Proof of (A) $\Rightarrow$ (B)

Assume that assertion (A) holds. Let I be k-subset of $[1, m]$ such that $v_j \in \bigcup_{i \in I} E_i$ for every $j \in J$.

The seed $g := \mathrm{g}^{(m)}_{[1,m]\setminus I}$ has weight $\|g\| = \#([1, m]\setminus I) = m-k = \varpi$ (Remark 20). Moreover, let j be an element of J. Then, there exists $i \in I$ such that $v_j \in E_i$. It follows from Equation (11) that i is an element of F_j. Therefore, $\mathrm{t}^{(m)}_i$ is a substring of s_j. Since g detects $\mathrm{t}^{(m)}_i$ (Lemma 24), g detects also its superstring s_j (point (iv) of Remark 5).

Hence, assertion (B) holds.

4.2.2 Proof of (B) $\Rightarrow$ (A)

Since s_j is obtained as the concatenation of the m^3 symbols long strings $s_{j,1}$, $s_{j,2}$, ..., s_{j,l_j} in this order, we may state:

Remark 31 *For every $j \in [1, n]$ and every $\eta \in [1, |s_j|]$, $s_j[\eta]$ occurs in $s_{j,h}$ where $h := \lceil \eta \,/\, m^3 \rceil$, and more precisely $s_j[\eta] = s_{j,h}[\eta - m^3(h-1)]$.*

Assume that assertion (B) holds. Let g be a seed with weight ϖ such that, for every $j \in J$, g detects similarity s_j at some position p_j.

For every $j \in J$, define $h_j := \lceil (p_j + 1) \,/\, m^3 \rceil$ and $h'_j := \lceil (p_j + |g|) \,/\, m^3 \rceil$.

Claim 32 *For every $j \in J$, h_j and h'_j are elements of Q_j.*

PROOF. According to Remark 31, $s_j[p_j + 1]$ and $s_j[p_j + |g|]$ occur in s_{j,h_j} and s_{j,h'_j}, respectively. Besides, the first letter and the last letter of g are `#`'s, requiring $s_j[p_j + 1] = \mathtt{1}$ and $s_j[p_j + |g|] = \mathtt{1}$, respectively. It follows that both s_{j,h_j} and s_{j,h'_j} are distinct from $\mathtt{0}^{m^3}$, requiring $h_j \in Q_j$ and $h'_j \in Q_j$, respectively. $\square$

Claim 33 *The differences of the form $h'_j - h_j$ with $j \in J$ are all equal.*

PROOF. Recall that for any two real numbers x and y, $\lceil x \rceil - \lceil y \rceil$ equals either $\lfloor x - y \rfloor$ or $\lceil x - y \rceil$. Hence, for any $j \in J$, $h'_j - h_j$ equals either $\lfloor \rho \rfloor$ or $\lceil \rho \rceil$, where $\rho := (|g|-1) \,/\, m^3$. However, h'_j and h_j are odd numbers as they are elements of R (Claim 32), and thus $h'_j - h_j$ is even. Since the two integers $\lfloor \rho \rfloor$ and $\lceil \rho \rceil$ are either equal or consecutive, exactly one of them, say d, is even: $h'_j - h_j = d$ for every $j \in J$. $\square$

Note that if J is reduced to a singleton then assertion (A) clearly holds.

Claim 34 *If J is not reduced to a singleton then $h_j = h'_j$ for every $j \in J$.*

PROOF. By way of contradiction assume both that there exists $j_0 \in J$ such that $h_{j_0} \neq h'_{j_0}$, and that J is not reduced to a singleton. Then, there exists an element $j_1 \in J$ with $j_0 \neq j_1$. Consider the two ordered pairs (h'_{j_0}, h_{j_0}) and (h'_{j_1}, h_{j_1}): according to Claim 32, both are elements of $R \times R$, and in addition, (h'_{j_1}, h_{j_1}) is distinct from (h'_{j_0}, h_{j_0}) since Q_{j_0} and Q_{j_1} are disjoint. However, according to Claim 33, $h'_{j_1} - h_{j_1}$ equals $h'_{j_0} - h_{j_0}$ and the latter difference is non-zero by hypothesis. Hence, R cannot be a Golomb ruler: contradiction. □

Claim 35 *If J is not reduced to a singleton then for every $j \in J$, g detects s_{j,h_j}.*

PROOF. Let γ be any index in $[1, |g|]$. It is clear that $\lceil (p_j + \gamma) \,/\, m^3 \rceil$ lies between h_j and h'_j inclusive. Now, assume that J is not reduced to a singleton. It follows from Claim 34, that the three integers $\lceil (p_j + \gamma) \,/\, m^3 \rceil$, h_j, and h'_j are equal; thus Remark 31 yields: $s_{j,h_j}[p_j - m^3(h_j - 1) + \gamma] = s_j[p_j + \gamma]$. Thence we deduce that g detects s_{j,h_j} at position $p_j - m^3(h_j - 1)$. □

For each $j \in J$, h_j is an element of Q_j (Claim 32), and thus there exists $i_j \in F_j$ such that $s_{j,h_j} = \mathrm{t}_{i_j}^{(m)}$. Since we may assume that J is not reduced to a singleton, g detects $\mathrm{t}_{i_j}^{(m)}$ (Claim 35), and thus g is of the form $g = \mathrm{g}_{X_j}^{(m)}$ where X_j is a subset of $[1, m]$ such that $i_j \notin X_j$ (Lemma 24). Besides, all X_j's are equal to a same set, denoted by X below. Indeed, the cardinality of X_j equals the weight ϖ of the seed g (Remark 20), and ϖ is at least two (Equation (10)). Hence, Lemma 21 applies: $g = \mathrm{g}_X^{(m)}$ for some ϖ-subset $X \subseteq [1, m] \backslash \{i_j : j \in J\}$. Now, $I := \{i_j : j \in J\}$ has cardinality $m - \varpi = k$ as the complement of X in $[1, m]$. Moreover, for any $j \in J$, $i_j \in F_j$ requires $v_j \in E_{i_j}$ (Equation (11)) and $i_j \in I$ ensures $E_{i_j} \subseteq \bigcup_{i \in I} E_i$: $v_j \in \bigcup_{i \in I} E_i$.

Hence assertion (A) holds.

This concludes the proof of Theorem 28. □

5 Conclusion

In this work, we have demonstrated the hardness of a tiling problem and of three combinatorial problems related to spaced seed design: the decision

problem Non Detection, and the two optimisation problems Maximum Weight Lossless Seed and Region Specific Optimal Seed. However, some questions remain open about spaced seed design since up to now, little is known about the structure of optimal seeds. Below, we list some directions for future work.

5.1 Tiling problems

In Section 2, we introduced the problem SSC and showed in Theorem 6 that SSC is NP-complete. This result raises several questions about the tiling of a finite one-dimensional figure.

Let d be a positive integer and $\mathbb{Z}^d$ denote the d-dimensional integer lattice. Let F be a subset of $\mathbb{Z}^d$, and let $\mathcal{P}$ be a set of subsets of $\mathbb{Z}^d$. Let us call *figure* the set F, and *tiles* the elements of $\mathcal{P}$. We say that $\mathcal{P}$ *tiles* F *(by translation)* if there exists a subset of $\left\{P + t : (P, t) \in \mathcal{P} \times \mathbb{Z}^d\right\}$ that is an exact cover of F.

Name: d-dimensional Finite Region Tiling (d-DFRT)
Instance: A finite subset $F \subseteq \mathbb{Z}^d$ and a finite set $\mathcal{P}$ of finite subsets of $\mathbb{Z}^d$.
Question: Does $\mathcal{P}$ tile F by translation?

This problem clearly belongs to NP. Moreover, when restricted to tiles of cardinality 2, d-DFRT reduces to the *perfect matching* problem which can be solved in polynomial time [29]. However, consider $H_2 := \{(0,0),(1,0)\}$, $V_3 := \{(0,0),(0,1),(0,2)\}$, and $\mathcal{P}_{23} := \{H_2, V_3\}$: the tiles H_2 and V_3 are called *horizontal domino* and *vertical triomino*, respectively. The restriction of 2-DFRT to instances $(F, \mathcal{P})$ satisfying $\mathcal{P} = \mathcal{P}_{23}$ has been shown NP-hard [30]. Thus, deciding whether two sets of cardinalities at most 3 tile a finite two-dimensional figure is NP-complete.

Now focus on the one-dimensional case. The proof of Theorem 6 can be easily adapted to settle the complexity of 1-DFRT. Consider the transformation that maps each instance $(V, \mathcal{E})$ of X3C to $([1, N], \{G_{i,j} : (i,j) \in [1,m] \times [1,q]\})$, where N and the $G_{i,j}$'s are as in the proof of Theorem 6: it induces a Karp-reduction from X3C to 1-DFRT. Hence, 1-DFRT is NP-complete, even if the figure is constrained to be a segment. Nevertheless, this reduction does not enable one to answer the following two open questions. Is 1-DFRT still NP-hard on a bounded number of input tiles? Does 1-DFRT remain NP-complete when the input tiles have bounded cardinalities?

5.2 Concerning the maximisation of the seed weight

Theorem 25 shows that MWLS is hard to approximate within bound $(\#S)^{0.5-\epsilon}$. We conjecture that, for any real $\delta \geq 0$, MWLS is not approximable within bound $(\#S)^{\delta}$.

Moreover, it is difficult to make any hypothesis on the form of the similarities without restraining the generality of the approach. So, we propose another formulation of lossless seed design with weight optimisation that seems interesting: "*Let m and k be integers satisfying $0 \leq k \leq m$. Find a seed of maximal weight that detects* all *(m,k)-similarities*". This problem has been partially addressed in the literature [22,12].

5.3 Concerning the approximation of Region Specific Optimal Seed

On the one hand, RSOS admits a trivial approximation algorithm with bound $\#S$. Indeed, let (ϖ, S) be an instance of RSOS.

- If there exists a similarity $s \in S$ with $|s|_1 \geq \varpi$, then return a seed of weight ϖ that detects s.
- Otherwise, return any seed of weight ϖ.

On the other hand, Theorem 28 guarantees that RSOS is NP-hard to approximate within ratio $\frac{e}{e-1} - \epsilon$. The existence of a constant-ratio approximation algorithm for RSOS is still open.

Acknowledgements

This work is supported by the CNRS STIC Specific Action #185 and the ACI IMPBio "REPEVOL". We thank G. Kucherov and L. Noé for introducing us to the problem Non Detection, and S. Guillemot for reference [26].

References

[1] F. Nicolas, E. Rivals, Hardness of optimal spaced seed design, in: A. Apostolico, M. Crochemore, K. Park (Eds.), Proceedings of the 16th Annual Symposium on Combinatorial Pattern Matching (CPM'05), Vol. 3537 of Lecture Notes in Computer Science, Springer-Verlag, 2005, pp. 144–155.

[2] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, D. J. Lipman, Basic Local Alignment Search Tool, Journal of Molecular Biology 215 (3) (1990) 403–410.

[3] A. Califano, I. Rigoutsos, FLASH: a fast look-up algorithm for string homology, in: L. Hunter, D. Searls, J. Shavlik (Eds.), Proceedings of the 1st International Conference on Intelligent Systems for Molecular Biology (ISMB'93), AAAI Press, 1993, pp. 56–64.

[4] S. Burkhardt, A. Crauser, P. Ferragina, H.-P. Lenhof, E. Rivals, M. Vingron, q-gram Based Database Searching Using a Suffix Array (QUASAR), in: Proceedings of the 3rd Annual International Conference on Computational Molecular Biology (RECOMB'99), ACM Press, 1999, pp. 77–83.

[5] R. M. Karp, M. O. Rabin, Efficient randomized pattern-matching algorithms, IBM Journal of Research and Development 31 (2) (1987) 249–260.

[6] E. Ukkonen, Approximate string-matching with q-grams and maximal matches, Theoretical Computer Science 92 (1) (1992) 191–211.

[7] W. I. Chang, E. L. Lawler, Sublinear approximate string matching and biological applications, Algorithmica 12 (4/5) (1994) 327–344.

[8] S. Burkhardt, J. Kärkkäinen, Better filtering with gapped q-grams, Fundamenta Informaticae 56 (1–2) (2003) 51–70.

[9] B. Ma, J. Tromp, M. Li, Patternhunter: faster and more sensitive homology search, Bioinformatics 18 (3) (2002) 440–445.

[10] M. Li, B. Ma, D. Kisman, J. Tromp, PatternHunter II: Highly sensitive and fast homology search, Journal of Bioinformatics and Computational Biology 2 (3) (2004) 417–439.

[11] L. Noé, G. Kucherov, Improved hit criteria for DNA local alignment, BMC Bioinformatics 5 (1) (2004) 149.

[12] G. Kucherov, L. Noé, M. Roytberg, Multiseed lossless filtration, IEEE/ACM Transactions on Computational Biology and Bioinformatics 2 (1) (2005) 51–61.

[13] K. P. Choi, L. Zhang, Sensitivity analysis and efficient method for identifying optimal spaced seeds, Journal of Computer and System Sciences 68 (1) (2004) 22–40.

[14] B. Brejová, D. G. Brown, T. Vinar, Optimal spaced seeds for homologous coding regions, Journal of Bioinformatics and Computational Biology 1 (4) (2004) 595–610.

[15] G. Kucherov, L. Noé, Y. Ponty, Estimating seed sensitivity on homogeneous alignments, in: Proceedings of the 4th IEEE Symposium on Bioinformatics and Bioengineering (BIBE'04), IEEE Computer Society Press, 2004, pp. 387–394.

[16] Y. Sun, J. Buhler, Designing multiple simultaneous seeds for DNA similarity search, Journal of Computational Biology 12 (6) (2005) 847–861.

[17] J. Buhler, U. Keich, Y. Sun, Designing seeds for similarity search in genomic DNA, Journal of Computer and System Sciences 70 (3) (2005) 342–363.

[18] M. Li, B. Ma, L. Zhang, Superiority and complexity of the spaced seeds, in: Proceedings of the 17th Annual ACM-SIAM Symposium On Discrete Algorithms (SODA'06), ACM Press, 2006, pp. 444–453.

[19] B. Ma, M. Li, On the complexity of the spaced seeds, Journal of Computer and System Sciences 73 (7) (2007) 1024–1034.

[20] D. Mak, Y. Gelfand, G. Benson, Indel seeds for homology search, Bioinformatics 22 (14) (2006) e341–349.

[21] G. Kucherov, L. Noé, M. Roytberg, A unifying framework for seed sensitivity and its application to subset seeds, Journal of Bioinformatics and Computational Biology 4 (2) (2006) 553–570.

[22] M. Farach-Colton, G. M. Landau, S. Cenk Sahinalp, D. Tsur, Optimal spaced seeds for faster approximate string matching, Journal of Computer and System Sciences 73 (7) (2007) 1035–1044.

[23] R. G. Downey, M. R. Fellows, Parameterized Complexity, Monographs in Computer Science, Springer, 1999.

[24] D. S. Hochbaum (Ed.), Approximation Algorithms for NP-Hard Problems, PWS Publishing Company, 1996.

[25] M. R. Garey, D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman and Co., 1979.

[26] D. Zuckerman, Linear degree extractors and the inapproximability of max clique and chromatic number, in: J. M. Kleinberg (Ed.), Proceedings of The 38th ACM Symposium on Theory of Computing (STOC'06), 2006, pp. 681–690.

[27] W. C. Babcock, Intermodulation interference in radio systems, Bell System Technical Journal 32 (1) (1953) 63–73.

[28] U. Feige, A threshold of $\ln n$ for approximating set cover, Journal of the Association for Computing Machinery 45 (4) (1998) 634–652.

[29] J. Edmonds, Paths, trees, and flowers, Canadian Journal of Mathematics 17 (3) (1965) 449–467.

[30] D. Beauquier, M. Nivat, E. Rémila, M. Robson, Tiling figures of the plane with two bars, Computational Geometry 5 (1995) 1–25.