View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Dagstuhl Research Online Publication Server

Analysis of the Period Recovery Error Bound

Amihood Amir
Department of Computer Science, Bar-Ilan University, Ramat Gan, Israel
https://u.cs.biu.ac.il/~amir
amir@esc.biu.ac.il

Itai Boneh
Department of Computer Science, Bar-Ilan University, Ramat Gan, Israel
barbunyaboy2@gmail.com

Michael Itzhaki
Department of Computer Science, Bar-Ilan University, Ramat Gan, Israel
michaelitzhaki@gmail.com

Eitan Kondratovsky
Department of Computer Science, Bar-Ilan University, Ramat Gan, Israel
https://u.cs.biu.ac.il/~kondrae
kondrae@cs.biu.ac.il

Abstract

The *recovery problem* is the problem whose input is a *corrupted* text T that was originally *periodic*, and where one wishes to recover its original period. The algorithm's *input* is T without any information about either the period's length or the period itself. An algorithm that solves this problem is called a *recovery algorithm*. In order to make recovery possible, there must be some assumption that not "too many" errors corrupted the initial periodic string. This is called the *error bound*. In previous recovery algorithms, it was shown that a given error bound of $\frac{n}{(2+\epsilon)p}$ can lead to $O(\log_{1+\epsilon} n)$ period candidates, that are *guaranteed* to include the original period, where p is the length of the original period (unknown by the algorithm) and $\epsilon > 0$ is an arbitrary constant.

This paper provides the first analysis of the relationship between the error bound and the number of candidates, as well as identification of the error parameters that still guarantee recovery. We improve the previously known upper error bound on the number of corruptions, $\frac{n}{(2+\epsilon)p}$, that outputs $O(\log_{1+\epsilon} n)$ period candidates. We show how to (1) remove ϵ from the bound, (2) relax the error bound to allow more errors while keeping the candidates set of size $O(\log n)$. It turns out that this relaxation on the previously known upper bound is quite challenging.

To achieve this result we provide what, to our knowledge, is the first known non-trivial lower bound on the *Hamming* distance between two periodic strings. This proof leads to an error bound, that produces a family of period candidates of size $2\log_3 n$. We show that this result is tight and further provide a compact representation of the period candidates. We call this representation the *canonic period seed*.

In addition to providing less restrictive error bounds that guarantee a smaller candidate set, we also provide a *hierarchy* of more restrictive upper error bounds that asymptotically reduces the size of the potential period candidate set.

2012 ACM Subject Classification Theory of computation → Pattern matching; Theory of computation → Sorting and searching

Keywords and phrases Period Recovery, Period Recovery Hierarchy, Hamming Distance

Digital Object Identifier 10.4230/LIPIcs.ESA.2020.5

Funding *Amihood Amir*: Partly supported by ISF grant 1475/18 and BSF grant 2018141.

© Amihood Amir, Itai Boneh, Michael Itzhaki, and Eitan Kondratovsky;
licensed under Creative Commons License CC-BY
28th Annual European Symposium on Algorithms (ESA 2020).
Editors: Fabrizio Grandoni, Grzegorz Herman, and Peter Sanders; Article No. 5; pp. 5:1–5:21
LIPICS Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Deterministic algorithms live in a “sterile” world: The problem is combinatorially clean, the environment is exact and unchanging, and thus the result is well defined. Reality is seldom so accommodating. Therefore, when applying algorithms to real world problems, one is generally required to approximate a solution.

Such approximations are derived from two sources: (1) Problems that can’t be solved efficiently, due to their inherent complexity, and (2) input that has been corrupted by various error-inducing sources. Theoretical Computer Science, in the field of Algorithms Development and Analysis, solves the above first problem by optimization algorithms (see e.g. [11–13, 22]). These are algorithms that come provably close to the optimal solution. The second problem is generally solved using the assumption that the input incurred the *smallest* number of possible corruptions (see e.g. [1, 15–17]). Two examples are the following:

The first solution to the Human Genome Sequencing project [21] used “shotgun sequencing”. Since the genome can not be read in its entirety, we are really presented with a “soup” of small subsequences of the genome. These subsequences need to be combined to produce the full sequence. The assumption was that the *shortest common superstring* [20] is the solution, i.e. the shortest string that can be cut into the input subsequences. Of course, there is no absolute guarantee that the original input was, indeed, the shortest. But this is the assumption that was made. Whenever there is statistical support for an assumption on the nature of the output, this support strengthens the result, but one can never be 100% sure that the produced output is indeed the “real” one.

The second example deals with Evolutionary Biology. By 1987, 145 races of humans were identified. The question was, how did the different races evolved? Cann, Stoneking, and Wilson [7] wrote their paper on human evolution, based on mitochondrial DNA. The idea is to fix a gene, and by its differences in the different races, construct a tree depicting the evolution, with races having smaller differences in the gene being closer to each other in the tree than races with greater differences. These evolutionary trees are constructed with the idea of parsimony in mind [8]. Clearly, though, the resulting tree is not the initial one, since different genes cause different trees, which then need to be reconciled [4, 9, 18, 19].

The above two are just examples of our shortcoming in approximating scientific phenomena. The scientific paradigm for reconstructing a phenomenon, at best can produce a measure of confidence, but never guarantee that the result of the algorithm indeed recovers the initial object.

The reconstruction task is the problem in which one has sampled a corrupted text T that was originally periodic, and wishes to recover its original period phenomenon. The input is T without any information about either the period’s length nor the period itself. This problem seems doomed since in most cases, even one error can lead to an ambiguity. However, in 2012, Amir et. al [3] introduced the *recovery model.* They have achieved some surprising results. They identified a phenomenon - *periodicity* - where one can get a corrupted input and produce a very small set of solutions (logarithmic in the input size), one of which is **guaranteed** to be the initial uncorrupted source, provided that the number of errors is reasonably bounded. This result was succeeded by additional papers, dealing with various types of error measures [2, 14], and recovering different phenomena [5].

In the *recovery model* for periodic strings, the output is a set of period candidates that must include the original period. Algorithms in such a model assume that the number of errors introduced into the text is limited. The error bound ensures that the recovery algorithm outputs a set of at most $o(n)$ candidates. However, there has not been a systematic study of what types of errors are to be bound, or whether the bound is tight. Nor has the relation between the error bound and the number of potential candidates ever been studied.

One of the topics that this work investigates is different types of error bounds. An error bound *type* is defined by the variables that it limits. We may consider a *universal error bound*, i.e. that in the entire string T there do not appear more than a constant number of errors. We may say that the number of errors is a function of the length of T. Another possibility is that the number of errors is a function of cycles of the period. In other words, how many copies of the period have to pass by without corruption. This last is the type of error considered historically. Some of those variables allow a degree of freedom, some are known while others might be unknown, and even without the ability to be estimated. For example, the historic error bound of [3] has a known value n, which is the length of input T, an unknown value p, the length of the original period (which is not part of the input), a set degree of freedom ϵ, and a parameter c determined by the metric. We can say in an abbreviated manner that previously known error bounds are of (n, p, ϵ, c)-type.

In this paper, we make the first attempt at a systematic analysis of the required error bound for recovery of a periodic string under mismatch errors. The only currently known relations between error bounds and number of candidates are: (1) the trivial bound of $|\Sigma|^{n/2}$ candidates for n possible errors, where Σ is the alphabet, since all possible periodic strings are candidates, and (2) The bound shown in Amir et al. [3]: Let T be an n-length periodic string with period P of length p. For $\epsilon > 0$, if we are guaranteed that there are no more than $\frac{n}{(2+\epsilon)p}$ mismatch errors, then a set of $\log_{1+\epsilon} n$ candidates can be constructed in $O(n \log n)$ time, that is guaranteed to include the original period P.

Amir et al. [3] proved their error bound result for pseudo-local metrics. For simplicity, we consider the *Hamming distance* as the distance metric, i.e. errors counted as the number of replacements. All our results can be easily extended to c-pseudo local metrics.

We desire answers to the following questions:
Past upper bounds depended on p. The first question to ask is whether this dependency is essential. In this paper, we study the case where the upper bound on errors is the number k. We are departing from the historical (n, p, ϵ, c)-type upper bounds, and instead consider k-type bounds, where k is either known or could be estimated. In this case, when one wants to reproduce the original periodic phenomena, the period length is unknown. On the other hand, the estimation of the number of corrupt copies of the period, as high as k, is assumed to be known.

In previous results, there is a "degree of freedom" variable ϵ. it seems that ϵ is not a natural variable to have in the upper bound formula. Its only use was to support the analysis. Indeed, one might want ϵ to be as close to zero as possible to relax upper bound on errors. However, this aim increases the number of candidates. This observation leads to our second question, whether it is possible to improve such analysis and get rid of ϵ without having too many candidates.

The second question leads to a greater task of how significant the upper bound of (n, p, c)-type can be relaxed while ensuring $o(n)$ candidates. This paper uses algebraic techniques to improve the upper bound on errors while preserving the property of $o(n)$ candidates. We give a partial answer, but believe that the question of how can be upper bound be further relaxed, and yet offer $o(n)$ periodic candidates is difficult and open to future research.

This paper successfully answers the above two questions.

However, on the path toward that goal, our first non-trivial insight is that even a single error (when $k = 1$) in text T might result in $\Theta(n)$ indistinguishable candidates. That is, without any additional knowledge on the problem, having a universal upper bound with $o(n)$ candidates is impossible. We further show that if the original period is repeated in the text $k+1$ times, i.e. there exists a *single* complete uncorrupted occurrence of the original period, then it is still possible to construct an example with $\Theta(\frac{n}{k(k+1)}) = \Theta(n)$ candidates.

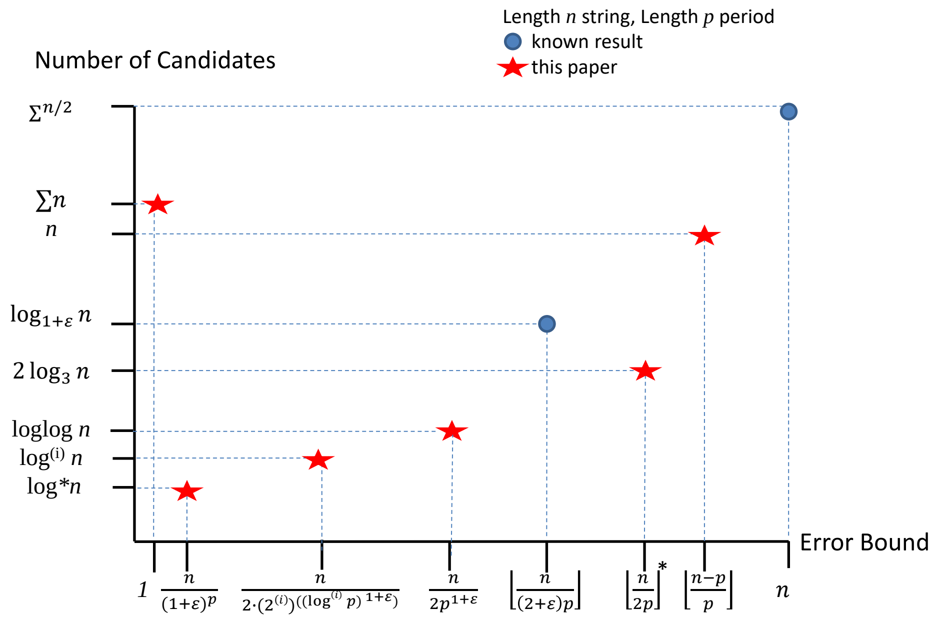
The novel combinatorial results are that (1) if the number of original period repetitions is $2k+1$ or higher then there is at most one candidate, i.e. *the original period can be recovered*, and (2) if the period originally repeated $2k$ times or higher ($k \geq 2$) then there are at most 2 indistinguishable candidates. We conclude that when k is known, the only required additional knowledge to find a constant number of period candidates is whether $p \leq \frac{n}{2k+1}$ or $p \leq \frac{n}{2k}$, respectively. Analysing the candidate set size when the number of repetitions is in the range between $k+1$ and $2k$ requires future research.

We highlight a connection between (n,p)-type and k-type upper bounds. Assume the case when k is unknown, but the above assumption about the repetitions holds. Then we should look at all possible k values and add their number of candidates. This leads us to the bounds $\lfloor \frac{n-p}{2p} \rfloor$ and $\lfloor \frac{n}{2p} \rfloor$, respectively. However, there are $\Theta(n)$ different values for k, and we require that the number of candidates must be $o(n)$, therefore we provide a new proof methodology.

In this paper, we analyze the $\lfloor \frac{n}{2p} \rfloor$ upper bound on errors, and prove that the number of candidates is $2\log_3 n$ and that this bound is tight by providing a family of examples. Moreover, we show that these candidates can be represented by a single *canonic period seed*.

Finally, we show a hierarchy of more restricted upper bounds of (n,p,ϵ,c)-type that yield $\Theta(\log^{(i)} n)$ candidates, where $\log^{(i)} n$ is $\log\log\cdots\log n$ i times.

In Fig. 1 we show the known bounds and the results of this paper. The formulae on the horizontal axis are error bounds, we show a hierarchy of candidate set size upper bounds. In various cases we show examples where this upper bound is indeed tight.



* $p \leq n/3$. Two canonic period constructed.

Figure 1 The results of this paper.

This paper is organized as follows: In Section 5, we discuss the k-type upper bounds. In Section 6, we tighten the logarithmic bound on the number of candidates. While in [3] the log base was dependent on ϵ of the allowed error bound, we show that for all ϵ the bound is $2\log_3 n$. In this section, we introduce a new algebraic method to analyze the distance between periodic strings derived by different seeds. This enables us to further tighten the upper error bounds to $\lfloor \frac{n}{2p} \rfloor$, and construct a **single** *canonic period seed*, where no more than $2\log_3 n$ candidates derived from it, among which the initial period is guaranteed to exist. Finally, in Section 7, we give a hierarchy of upper bounds that produce decreasing number of candidates that include the original period. The hierarchy decreases from $\log n$ via $\log\log n$, $\log^{(i)} n$, to $\log^* n$.

2 Preliminaries

Let Σ be an alphabet. A *string* T over Σ is a finite sequence of letters from Σ. By $T[i]$, for $0 \leq i \leq t-1$, we denote the i^{th} letter of T. The *empty string* is denoted by λ. By $T[i..j]$ we denote the string $T[i]\dots T[j]$ called a *substring* of T (if $i > j$, then the substring is the empty string). A substring is called a *prefix* if $i = 0$ and a *suffix* if $j = t-1$. The prefix of length $j+1$ is denoted by $T[..j]$. While by $T[i..]$ we denote the suffix which starts from index i in T. We will follow the convention of using capital letters for string names, and the same small letter for the length of the string. For example: the length of string T is t. Other notations we use is $T[-i] = T[t-i]$.

An n-length string T is periodic if $T = P^k P'$, where $k \in \mathbb{N}$, $k \geq 2$, and P' is a prefix of P (the prefix might be empty). P^k is the concatenation of P to itself k times. It is clear that P is a substring of T and $p \leq t/2$.

If T is periodic, the shortest P, such that $T = P^k P'$ is called *the period of* T. There is a unique such period by the *periodicity lemma* [10]. The periodicity lemma states that for two different periods of lengths p and q, where $n \geq p + q - gcd(p,q)$, there exists a period of length $gcd(p,q)$, where $gcd(a,b)$ is the greatest common divisor of a and b. A string P is *primitive* if there is no string S such that $P = S^k$ and $k > 1$.

The recovery problem seeks the original period of a corrupted text T. The corruption may have caused T to be non periodic. Thus there may be a number of indistinguishable periods that generate strings of length t. We are seeking an error bound on the distance between these strings and T that forces only a small set of such periods.

▶ **Example 1.** $T = abaabaab$, then $P = aba$ is the period. In our exposition, for the sake of brevity, we may denote this string by $T = P^{2\frac{2}{3}}$. We allow ourselves to use fractions $\frac{x}{y}$ where $\frac{x}{y} \times t$ is an integer. In this example, we could not use $\frac{1}{4}$ because $\frac{1}{4} \times 3$ is not an integer.

Let T and S be two n-length strings, their *Hamming distance*, denoted by $Ham(T,S)$, is the number of mismatches between these strings. The Hamming distance represents the number of substitutions required to convert one string to the other.

Let P be a primitive string of length p. Let T be a n-length string, P^∞ denotes the periodic string of length ℓ, where the value of ℓ is clear from the context. For example, in the expression $Ham(T, P^\infty)$, both operands T and P^∞ should be of equal lengths, $\ell = n$. P is called a *period candidate* with bound e if $Ham(T, P^\infty) \leq e$. If $\ell = n$, we denote $T_P = P^\infty$. This notation simplifies the expressions when dealing with two different periodic candidates of T. For example, when observing $Ham(T_P, T_Q)$, where P and Q are two different periods.

Let P be a periodic candidate of T. The substring ranges of the form $[i, i+p-1]$, where $i = 1, p+1, 2p+1, \dots, (\lfloor \frac{n}{p} \rfloor - 1)p + 1$ denote *full occurrences* of P in T. A *full occurrence* $[i, i+p-1]$ is called an *exact occurrence* of P if $P = T[i..i+p-1]$, otherwise it is a *corrupted occurrence*.

3 A universal Error Bound Does not Allow Recovery

We begin by proving that a universal error bound, even if it is a single error in the string, does not suffice for $o(n)$ candidates. We describe an example of $\Omega(n)$ indistinguishable candidates when there is only a single corruption in some periodic text. Then we generalize it to any number of corruptions $k \in \mathbb{N}$.

▶ **Example 2.** Let $T = a^{2\ell}ba^{4\ell+1}$, where $n = 6\ell + 2$. We show that there are $\frac{n}{6}$ indistinguishable period candidates.

The original periodic source of the text T is one of the following.
$C = \{a^{2\ell}b\ a^{2\ell}b\ a^{2\ell}, a^{2\ell}ba\ a^{2\ell}ba\ a^{2\ell-2}, a^{2\ell}ba^2\ a^{2\ell}ba^2\ a^{2\ell-4}, \dots, a^{2\ell}ba^{\ell}\ a^{2\ell}ba^{\ell}\}$
$= \{(a^{2\ell}b)^{2+\frac{2\ell}{2\ell+1}}, (a^{2\ell}ba)^{2+\frac{2\ell-2}{2\ell+2}}, (a^{2\ell}ba^2)^{2+\frac{2\ell-4}{2\ell+3}}, \dots, (a^{2\ell}ba^{\ell})^2\}$

Each such possible source text has a different period. And for each such source, the number of the introduced corruptions is exactly 1. The second b is replaced by an a.

It is clear that $|C| = O(n)$, because of the following reason. Observe that the third occurrence of the period is not complete, only a suffix of it occurs. It begins with $a^{2\ell}$ and for each successive item, the length of the suffix of the third period occurrence decreases by 2, until it becomes the empty string. Thus, $|C| = \ell + 1 \approx \frac{n}{6}$.

It is easy to generalize the example to the case where k corruptions are allowed and there are $k+1$ full occurrences of the period, in other words, we still have one uncorrupted periodic occurrence. In this case, the constructed example would has $\Omega(\frac{n}{k(k+1)}) = \Omega(n)$ indistinguishable period candidates.

▶ **Example 3.** $T = a^{\ell k}ba^{\ell k^2+k-\ell-1}$, where $n = \ell k^2 + \ell k + k$, $k \geq 2$, and $\ell \geq k$. We show that there are $\Omega(\frac{n}{k(k+1)})$ indistinguishable period candidates.

The original periodic source of the text T is one of the following.
$C = \{(a^{\ell k}b)^k\ a^{\ell k}, (a^{\ell k}ba)^k\ a^{\ell k-k}, (a^{\ell k}ba^2)^k\ a^{\ell k-2k}, \dots, (a^{\ell k}ba^{\ell})^k\}$
$= \{(a^{\ell k}b)^{k+\frac{\ell k}{\ell k+1}}, (a^{\ell k}ba)^{k+\frac{\ell k-k}{\ell k+2}}, (a^{\ell k}ba^2)^{k+\frac{\ell k-2k}{\ell k+3}}, \dots, (a^{\ell k}ba^{\ell})^k\}$

Thus, $|C| = \ell + 1 = \Theta(\frac{n}{k(k+1)})$.

4 Results

Our main contributions are the following.

▶ **Theorem 4.** *Let k be a fixed integer value. Let P be a period for which $Ham(T, T_P) \leq k$, and P has at least $2k+1$ full occurrences in T. Then the number of possible candidates for P is at most 1.*

▶ **Theorem 5.** *Let $k \geq 2$ be a fixed integer value. Let P be a period for which $Ham(T, T_P) \leq k$, and P has at least $2k$ full occurrences in T. Then the number of possible candidates for P is at most 2.*

▶ **Theorem 6.** *Let P be a period for which $Ham(T, T_P) \leq \lfloor \frac{n}{2p} \rfloor$, and $p \leq \frac{t}{3}$. Then the number of possible candidates for P is at most $2\log_3(n) = O(\log n)$.*

5 k-Type Upper Error Bounds

In this section, we prove Theorems 4 and 5. Doing so requires a few new lemmas. We begin by examining the case where $k = 1$. Then we generalize our results to any $k \in \mathbb{N}$.

Let P, Q be two period candidates of T, with lengths p, q, respectively. Without loss of generality, assume $p > q$. We recall a useful lemma that was proven when P fully occurs at least twice.

▶ **Lemma 7** ([3]). *For any two periods P and Q, if they both fully occur at least twice, then $Ham(T_P, T_Q) \geq 2$.*

In [3], it is claimed that for the general case $Ham(T_P, T_Q) \geq \frac{n}{p}$. Our Corollary 10 is exactly this result. However, the previous proof fails to handle properly the case where $\left\lfloor \frac{n}{p} \right\rfloor$ is odd. We take care of this missed case.

▶ **Lemma 8.** *Let T be a text and assume there is at most one corruption error in T. Further assume that T is a corruption of an original periodic string T_P where the period P fully occurs at least 3 times in T (neither the period nor its length is part of the input) . Then there is a single period candidate for the original text.*

Proof. Let us assume in contradiction that there are P and Q two possible period candidates for T, $P \neq Q$. We begin by showing that $p \neq q$. If $p = q$, there are three occurrences of P and Q which are of the same length, but $Ham(T_P, T_Q) \leq 2$. It means that one full occurrence of P is equal to the corresponding full occurrence of Q, namely, $P = Q$. As a consequence, $p \neq q$. The proof is divided into three cases.

Case 1. When $q \mid p$. Let $k = \frac{p}{q}$.

P is primitive, against each full occurrence of P there are Q^k. That is, each full occurrence of P in T_P should cause at least 1 mismatch with T_Q, otherwise P is not primitive, thus $Ham(T_P, T_Q) \geq 3$. However, one error is assumed thus, $Ham(T_P, T_Q) \leq Ham(T, T_P) + Ham(T, T_Q) = 2$. Thus, we got a contradiction.

Case 2. When $q \mid 2p$ and $q \nmid p$.

First, we observe that q must be even. Let $Q = Q_P Q_S$, where Q_P and Q_S are exactly the first and last half of Q of lengths $q/2$. We observe the prefix of length $\frac{3q}{2}$ of the three first full occurrences of P. It is easy to see that against the first and third occurrence of P in T_P, there are $Q_P Q_S Q_P$ in T_Q. On the other hand, against the second full occurrence of P there are $Q_S Q_P Q_S$. It is clear that $Q_P \neq Q_S$, otherwise Q is not primitive. Thus, we must have at least 3 mismatches between T_P and T_Q, in contradiction.

Case 3. Otherwise, when $q \nmid 2p$ and $q \nmid p$.

From the alignment lemma, the number of mismatches between T_P and T_Q is at least 4, in contradiction. ◀

▶ **Corollary 9.** *For any two periods P and Q that fully occur at least three times, then $Ham(T_P, T_Q) \geq 3$.*

▶ **Corollary 10.** *Let $k \geq 2$. For any two periods P and Q that fully occur at least k times, then $Ham(T_P, T_Q) \geq k$.*

Proof. If k is even then use Lemma 7 on all disjoint consecutive pairs of full occurrences of P against the rotations of Q. If k is odd, use Lemma 7 for all full occurrences of P except its last three full occurrences. Handle these occurrences by Corollary 9. ◀

We now have the tools to prove the main theorems. The proofs can be found in the full version of this paper.

▶ **Theorem 4.** *Let k be a fixed integer value. Let P be a period for which $Ham(T, T_P) \leq k$, and P has at least $2k + 1$ full occurrences in T. Then the number of possible candidates for P is at most 1.*

▶ **Theorem 5.** *Let $k \geq 2$ be a fixed integer value. Let P be a period for which $Ham(T, T_P) \leq k$, and P has at least $2k$ full occurrences in T. Then the number of possible candidates for P is at most 2.*

6 Tighter Bounds for $2\log_3 n$ Candidates

In this section we relax the constraints of Amir et al. [3] regarding the number of allowed mismatches, and yet provide a much smaller candidate set. The main tool in achieving this is a string combinatorics theorem that improves the best known lower bound on the Hamming distance between two periodic strings.

6.1 Lower bound on the Hamming distance between strings

We start by improving the best known lower bound on the number of errors between two periodic strings. We provide a new nontrivial expression that tightens the lower bound and give tight examples. Our formula results from a careful analysis of the *Turning Points*, *Windows*, and *Adjacency Strings* of the two strings.

▶ **Definition 11.** *A string P is called* Non-trivial if it contains at least two distinct characters.

▶ **Theorem 12.** *Let P, Q be non-periodic, non-trivial strings of lengths p, q, respectively, s.t. $q < p, q \nmid p$, and let $p = aq + b, \quad 0 < b < q$, then*

1. $\forall m, 4 \leq m < \frac{q}{\gcd(p,q)}, \quad Ham(P^m, Q^\infty) \geq m(a+1) - 2 + \left\lfloor \frac{mb}{q} \right\rfloor$
2. $\forall m, 2 \leq m < \frac{q}{\gcd(p,q)}, \quad Ham(P^m, Q^\infty) \geq m(a+1) - 2$

In the next subsections, we provide the proof of Theorem 12 in-depth.
For pedagogical reasons and for simplicity and comprehensibility, we will prove the theorem for strings of co-prime lengths and binary alphabet, $\Sigma = \{\alpha, \beta\}$, and then show a reduction from the general case to the simplified cases.

Troughout the proof, we will assume w.l.o.g that $p > q$ and that α appears in Q at least as many times as β. We will also let a, b satisfy $p = aq + b, \quad a, b > 0$.

We begin with definitions that will help in understanding the motive and correctness of the proof and methods.

6.1.1 Groundwork

▶ **Definition 13** (Index mapping function)**.** *The* index mapping function *is defined as follows:*

$$\delta_m : [0, p-1] \to [0, q-1], \quad \textit{where} \quad \delta_m(i) = i + (m-1)p \mod q$$

The index mapping function maps the index of the m_{th} occurrence of $P[i]$ in P^m to the corresponding index in Q^∞. Note, that the mapping function is also dependent on the lengths of P, Q. We refrain from indexing the function with these symbols for simplicity's sake.

▶ **Definition 14** (Adjacency string)**.** *Let P, Q be strings of co-prime lengths, and $p > q$. The adjacency string $\tilde{Q}_p$ is a string of length q that satisfies $\forall i, 0 \leq i < q, \tilde{Q}_p[i] = Q[i \cdot p \mod q]$. The adjacency subset-string $\tilde{P}_q$ is a multi-string of length q that satisfies*

$$\forall i, 0 \leq i < q, \tilde{P}_q[i] = \{P[j] | j \equiv_q i \cdot p\}$$

A *subset string* is a string where each string-position might contain several characters. The term was originally defined at [6]. Note that in our case, the same character can recur multiple times at a single position.
The motivation behind this representation is to encapsulate the index-mapping function. It holds that $\forall m, i, j$ where $j \equiv_q i \cdot p$ implies $\tilde{Q}_p[(j+m) \mod q] = Q[\delta_{m+1}(i)]$. In words, it means that a character in $\tilde{P}$ that aligns against $\tilde{Q}_p[i]$, will align against $\tilde{Q}_p[i+1]$ in its next recurrence. $\tilde{P}$ was defined for mere convenience; $Q[i] = \tilde{Q}_p[j] \rightarrow P[i+n \cdot q] \in \tilde{P}_q[j]$, for all $i + n \cdot q < p$ [1].
We will often omit the subscript and simply write $\tilde{Q}, \tilde{P}$. Fig. 2 shows an example of an adjacency string. In this example, one can see that if a character σ in P is aligned with $\tilde{Q}[i]$, the next occurrence of σ will align with $\tilde{Q}[(i+1) \mod q]$.

P=0123456 $\tilde{P}_4$ ={0,4}3{2,6}{1,5}
Q=abcd $\tilde{Q}_7$ =adcb

P^4 =0123456012345601234560123456
Q^7 =abcdabcdabcdabcdabcdabcdabcd

Figure 2 Example for adjacency string, Definition 14.

During the proof, we refer to the symbol "β" as *Black node*, and to "α" as *White node.*

▶ **Definition 15.** *The* m-forward window $W^f_m(i)$ *is the multi-set of characters in* Q *that align against* $\tilde{P}[i]$ *in the next* m *repetitions of* P*, which is* $\{\tilde{Q}[i], ..., \tilde{Q}[i+m-1]\}$
The m-backward window $W^b_m(i)$ *is the multi-set of characters in* P *that align against* $\tilde{Q}[i]$ *in the next* m *repetitions of* P*, which is* $\{\tilde{P}[i], ..., \tilde{P}[i-m+1]\}$

In Fig. 3 we see an example for backwards and forwards windows. One can see that $W^b_3(2)$ contains all the characters that touch $\tilde{Q}[2]$ in the next 3 repetitions of P, and that $W^f(0)$ contains all the characters that touch $\tilde{P}[0]$ in the next 3 repetitions. It is also apparent that W^b_3 contains the character '$0'$ twice.

P =01234067 $\tilde{P}_5$ ={0,0}3{1,6}4{2,7}
Q =abcde $\tilde{Q}_7$ =adbec

P^3 =01234067|01234067|01234067
Q^∞=abcdeabc|deabcdea|bcdeabcd

$$W^b_3(2) = \left\{\tilde{P}[2], \tilde{P}[1], \tilde{P}[0]\right\} = \{1, 6, 3, 0, 0\}$$

P^3 =01234067|01234067|01234067
Q^∞=abcdeabc|deabcdea|bcdeabcd

$$W^f_3(0) = \left\{\tilde{Q}[0], \tilde{Q}[1], \tilde{Q}[2]\right\} = \{a, d, b\}$$

Figure 3 Example for backward and forward windows, Definition 15.

[1] The converse is not true for, as the mapping function maps to characters in Q.

▶ **Definition 16** (Heavy index)**.** *We say that i is a* heavy index *if it satisfies* $|\tilde{P}[i]| = \left\lceil \frac{p}{q} \right\rceil = a + 1$.

We will call the last b characters of P *heavy characters.*

▶ **Corollary 17.** *$\forall m, i$, the number of characters in $W^b_m(i)$ are at least $am + \left\lfloor \frac{mb}{q} \right\rfloor$.*

Proof. Considering $\tilde{P}$ has at least a characters at every index, and the number of heavy indices in any m repetitive recurrences is at least $\left\lfloor \frac{mb}{q} \right\rfloor$, giving the required result. The heavy indices are distributed equally in $\tilde{P}$ because in P, the if the heavy indices are $0, 1, ..., j-1$, then on the next recurrence they will align with $j, j+2, ..., 2(j-1)$, and so on, and by abstract algebra the distribution of heavy indices in $\tilde{P}$ is equal, though not probabilistic. ◀

▶ **Definition 18 (Turning points).** *Let $t_1, t_2, ..., t_\ell$ be the indices such that $\tilde{Q}_p[t_i] \neq \tilde{Q}_p[t_i + 1 \mod q]$. We call these indices* Turning points.

This definition will help in counting mismatches; Turning points create mismatches, as a character that matches a turning point will create a mismatch in the next recurrence of P.

See Fig. 4 for an example of Turning points.

P =0101010101010101

Q =0100010011011

$\tilde{Q}_7$ =0001101111000

Figure 4 Example for Turning points, Definition 18.

▶ **Corollary 19.** *ℓ is even.*

Proof. Given that turning points are the only indices that satisfy $q[t_i] \neq q[t_{i+1}]$, and therefore if ℓ is odd, then it can be expressed as $2\ell' + 1$, and

$$q[t_1] \neq q[t_2] ... \neq q[t_{2\ell'+1}] \neq q[t_1] \rightarrow$$
$$q[t_1] = q[t_3] = ... = q[t_{2\ell'+1}] \neq q[t_1] \rightarrow$$
$$q[t_1] \neq q[t_1]$$ ◀

▶ **Corollary 20.** *$\ell \geq 2$*

Proof. Let us assume that $\ell = 0$. Since there are no turning points, there is no index s.t. $\tilde{Q}_p[i] \neq \tilde{Q}_p[i+1]$, and consequently the string Q is trivial, a contradiction.
This means $\ell \geq 1$, and by using Corollary 19, $\ell \geq 2$. ◀

▶ **Corollary 21.** $Ham(P^2, Q^\infty) \geq \sum_{i=1}^{\ell} |\tilde{P}[i]| \geq a\ell.$

Proof. Let i be a turning point. By the definition of adjacency string, all the characters in $\tilde{P}[i]$ will align in the next two repetitions against $\tilde{Q}[i], \tilde{Q}[(i+1) \mod q]$. By the definition of turning point, $\tilde{Q}[i] \neq \tilde{Q}[(i+1) \mod q]$ and accordingly each character in $\tilde{P}[i]$ will cause exactly one mismatch. Hence, the minimal number of mismatches is $\sum_{i=1}^{\ell} |\tilde{P}[i]|$. Considering that $\forall i, \tilde{P}[i]$ contains at least a characters, the expression is at least $a\ell$. ◀

Given the above facts, we can now proceed to the proof of Theorem 12.

6.1.2 Co-prime proof

In this subsection, we constrain p, q to be co-prime, namely, $\gcd(p, q) = 1$.

Proof. We will divide the proof to 3 cases, and treat each of them separately.

▶ **Case 1** ($\ell \geq 4.$). When $m = 2$, it holds that

$$Ham(P^2, Q^\infty) \geq 4a = 2(a+1) - 2 + 2a > 2(a+1) - 2 + \left\lfloor \frac{2b}{q} \right\rfloor.$$

We can,accordingly, assume that $m \geq 3$.

Using Corollary 17, at least $4\left\lfloor \frac{mb}{q} \right\rfloor$ indices in the m-backward windows of the turning points are heavy; So using Corollary 21, every $m+1$ repetitions we get at least that many mismatches in addition to the already-calculated number. Set $e_m = \left\lfloor \frac{mb}{q} \right\rfloor$. Given that $m \geq 3$, it holds that either $e_m \geq 1$, or $e_{m+2} \leq 1$ [2], and thus $e_{m+2} \leq 4e_m + 1$.

Using the above results, we show that $Ham(P^{2m}, Q^\infty)$ causes more mismatches than allowed for $Ham(P^{2m+1}, Q^\infty)$.

$$\begin{aligned}
Ham(P^{2m}, Q^\infty) &\geq 4am + 4e_{2m-1} \\
&\geq 4am + e_{2m+1} - 1 \geq 4am - 1 + e_{2m+1} \\
&\geq (2m+1)(a+1) + 2am - a - 2m - 2 + e_{2m+1} \\
&\geq (2m+1)(a+1) + 2(m-1)(a-1) - 3 + a + e_{2m+1} \\
&\geq (2m+1)(a+1) - 2 + e_{2m+1}
\end{aligned}$$ ⌟

In the rest of the cases, there are only two turning points. We denote, for simplicity, t_w, t_b as the turning points from white to black and from black to white, respectively.

▶ **Case 2.** [$\ell = 2, \quad (t_b - t_w) \mod q = 1$] In words, it means that there is only one occurrence of the character β in Q. Let i_b be an index such that $\tilde{Q}[i_b] = \beta$. By the assumptions, this index is unique.

Let $W = W_m^b(i_b)$. Let $c = |W|$, and set c_w to be the number of white characters in W. The number of black characters in the window, denoted by c_b satisfies $c_b = c - c_w$. Using Corollary 17, we can claim $c \geq ma + \left\lfloor \frac{mb}{q} \right\rfloor$.

If $c_b = 0$, then the black character that must exist in P (since it is not trivial), did not align against the only black character of Q, and thus created m mismatches, so we can assume w.l.o.g that at least one black character appears in the window, as it reduces the number of errors by at least 1.

Putting it all together and maximizing c_w to be $c-1$, leads to

$$\begin{aligned}
Ham(P^m, Q^\infty) &\leq c_w \cdot 1 + (c - c_w) \cdot (m-1) \\
&\leq_1 (c-1) \cdot 1 + m(c - (c-1))(m-1) \\
&= m + c - 2 \\
&= m + am + \left\lfloor \frac{mb}{q} \right\rfloor - 2 \\
&= m(a+1) - 2 + \left\lfloor \frac{mb}{q} \right\rfloor \qquad (1)
\end{aligned}$$

⌟

[2] If $\frac{3b}{q} < 1$, then $\frac{5b}{q} < 2$

▶ **Case 3** ($\ell = 2, \quad (t_b - t_w) \mod q \neq 1$)**.** First, define $d = (t_b - t_w) \mod q$.

▶ **Corollary 22.** *The number of mismatches caused by each character on index i in the adjacency string $\tilde{P}$ after m repetitions is the minimum between the number of white nodes and black nodes in the m-forward windows of i.*

Proof. Each character creates a mismatch by either aligning with a white index or a black index, so the smallest number of mismatches a character can cause is the minimum between the black and white indices in its next occurrences. ◀

We consider three cases here; the first is where $m \geq d+1$. The second is where $m \leq d$, and also $m \geq 4$. The last is $m \leq d, m \in \{2,3\}$.

▶ **Case 3.1** ($m \geq d+1$)**.** Recall that there are at least as many white characters as black. Mark $W_w = W^b_{m-1}(t_w)$, and $W_b = W^b_2(t_w+2)$. Since $d \geq 2$ and $q \geq m+1$, there are no overlaps between the windows, and all the indices in W_b are black.

In W_w, all of the characters will align against both t_w, t_w+1, and hence create a mismatch. Note that each of the indices $\{t_w - 1, ..., t_w - m + 3\}$ will also align against $t_w - 1$ and $t_w + 2$, hence creating two mismatches. Thus, the total sum of mismatches is at least $a(2+2((t_w-1)-(t_w-m+3)+1)) = 2a(m-2)$, as each index of $\tilde{P}$ has at least a characters.

Considering that $m > d$, $t_w + 1$ will align against $t_b + 1$ and $t_w + 2$ will align against $t_b + 2$, and as a consequence we get at least 2 mismatches for each index, regardless of m. Therefore the total number of mismatches is at least $2a$, which leads to a total sum of $2a(m-1)$ mismatches in both windows. Evaluating this value:

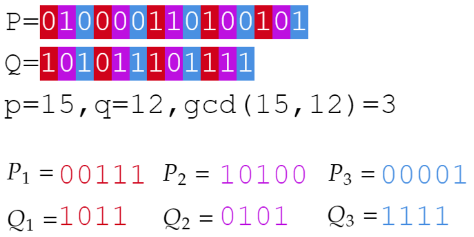
$$\begin{aligned}
&2a(m-1) = 2am - 2a \\
&= am + am - 2a \\
&= m(a+1) + a(m-2) - m \\
&\geq m(a+1) - 2 \text{ (by minimizing } m \text{ to 2)}
\end{aligned}$$

We now show $\left\lfloor \frac{mb}{q} \right\rfloor$ additional mismatches. The number of heavy indices in the m-backwards window of $t_w + 1$ is at least $\left\lfloor \frac{mb}{q} \right\rfloor$, and all of the characters in the window create at least one mismatch, leading to $\left\lfloor \frac{mb}{q} \right\rfloor$ additional mismatches. ⌟

▶ **Case 3.2** ($4 \leq m, \quad m \leq d$)**.** First, it is trivial (yet crucial) to see that $d \geq 4$. Using the previous method, we consider $W^b_{m-1}(t_w)$. Given that $m \leq d$, and using Corollary 22, any character at position $t_w - i$ will cause $\min(i+1, m-i-1)$ mismatches, so we have $1 + 2 + ... + \lfloor \frac{m}{2} \rfloor + ... + 2 + 1 \geq a(1 + 2(m-3) + 1) = 2a(m-2)$ mismatches. The same claim can be made for $W^b_{m-1}(t_b)$, summing up to at least $4a(m-2)$ mismatches after m repetitions.

$$\begin{aligned}
&Ham(P^m, Q^\infty) \geq 4a(m-2) \\
&\geq m(a+1) - m + 3am - 8a \geq m(a+1) + m(3a-1) - 8a \\
&\geq m(a+1) + 4(3a-1) - 8a \geq m(a+1) + 12a - 8a - 4 \\
&\geq m(a+1) + 4(a-1) \geq m(a+1)
\end{aligned}$$

We now need to find $\left\lfloor \frac{mb}{q} \right\rfloor - 2$ additional mismatches. The inequality $\left\lfloor \frac{(m-1)b}{q} \right\rfloor \geq \left\lfloor \frac{mb}{q} \right\rfloor - 1$ holds, and accordingly we have an additional $2\left\lfloor \frac{(m-1)b}{q} \right\rfloor \geq 2(\left\lfloor \frac{mb}{q} \right\rfloor - 1) \geq \left\lfloor \frac{mb}{q} \right\rfloor - 2$ mismatches. ⌟



```
P=010000110100101
Q=101011101111
p=15,q=12,gcd(15,12)=3
```

P_1 = 00111 P_2 = 10100 P_3 = 00001

Q_1 = 1011 Q_2 = 0101 Q_3 = 1111

Figure 5 Example for strings decomposition (Definition 25).

▶ **Case 3.3** ($m \le d, \quad 2 \le m \le 3$)**.** In this case, we only have to show that $Ham(P^m, Q^\infty) \ge m(a+1)-2$.
If $m = 2$, then we have $2a = m(a+1)-2$ mismatches directly from Corollary 21.
If $m = 3$, then $t_w, t_w - 1, t_b, t_b - 1$ will all create at least one mismatch, which results in $4a$ mismatches, and

$$4a = 3a + a = 3(a+1) - 2 + a - 1 \ge 3(a+1) - 2 = m(a+1) - 2$$ ⌟

◀

6.1.3 Non-divisible proof

The case of $gcd(p,q) = 1$ was proven in Subsection 6.1.2. We now prove that the theorem is true for the more general version, where $q \nmid p$. Let $gcd(p,q) = g > 1$, and let $q' = \frac{q}{g}, p' = \frac{p}{g}$, and $p' = a'q' + b'$.

▶ **Corollary 23.** $\left\lfloor \frac{mb}{q} \right\rfloor = \left\lfloor \frac{mb'}{q'} \right\rfloor$

Proof.

$$\left\lfloor \frac{mb}{q} \right\rfloor = \left\lfloor \frac{m(p \mod q)}{q} \right\rfloor = \left\lfloor \frac{m(g(a'q'+b') \mod gq')}{gq'} \right\rfloor = \left\lfloor \frac{m(gb' \mod gq')}{gq'} \right\rfloor = \left\lfloor \frac{mgb'}{gq'} \right\rfloor = \left\lfloor \frac{mb'}{q'} \right\rfloor$$

◀

▶ **Corollary 24.** $a' = a$

We will now decompose P and Q to smaller strings of co-prime lengths.

▶ **Definition 25** (Decomposed strings)**.** *Given strings P, Q, $\gcd(p,q) = g$, the decomposed string of P at index i is $P_i = P[i]P[i+g]...P[i+(p'-1)g]$. The decomposed strings of Q are defined respectively.*

▶ **Corollary 26.** *$\exists i$ s.t. P_i is non-trivial.*

Proof. If such an index does not exist, for each index i, P_i is trivial, hence $\forall i, P_i = c^{p'}$, and accordingly $P = (P[0]...P[g-1])^{p'}$, and P is periodic, a contradiction to our assumptions. The same claim can be made about Q. ◀

▶ **Definition 27.** *Let P, Q be strings, and let i, j be the minimal indices such that P_i and P_j are not trivial. We will say P, Q are* aliens *if $i \ne j$, and we will say P, Q are* similar *if $i = j$. If $\gcd(p,q) = 1$, then P, Q are aliens regardless.*

The latter definition is rather synthetic, since we only consider the minimal indices.

▶ **Case 1** (P, Q are aliens)**.** Let i, j be the minimal indices s.t. P_i, Q_j are non-trivial. In this case, $Ham(P^m, Q^\infty) \geq Ham(P_i^m, c_i^\infty) + Ham(c_j^{p'm}, Q_j^\infty)$, where $P_i = c_i^{p'}, Q_j = c_j^{q'}$.

Now,

$$Ham(P_i^m, c_i^\infty) = m \cdot Ham(P_i, c_i^{p'}) \geq m \tag{1}$$

$$\begin{aligned} Ham(c_j^{p'm}, Q_j^\infty) &\geq m \cdot Ham(c_j^{p'}, Q_j^{\left\lfloor \frac{p'm}{q'} \right\rfloor}) \\ &\geq \left\lfloor \frac{p'm}{q'} \right\rfloor = \left\lfloor \frac{m(a'q'+b')}{q'} \right\rfloor \\ &= ma' + \left\lfloor \frac{mb'}{q'} \right\rfloor = ma + \left\lfloor \frac{mb}{q} \right\rfloor \end{aligned} \tag{2}$$

So,

$$Ham(P^m, Q^\infty) \geq Ham(P_i^m, c_i^\infty) + Ham(c_j^{p'm}, Q_j^\infty) \geq m + am + \left\lfloor \frac{mb}{q} \right\rfloor = m(a+1) + \left\lfloor \frac{mb}{q} \right\rfloor$$ ⌟

▶ **Corollary 28.** *If P, Q are alien strings of non co-prime lengths, then $Ham(P^m, Q^\infty) \geq m(a+1) + \left\lfloor \frac{mb}{q} \right\rfloor$, for all $m > 0$.*

▶ **Case 2** (P, Q are similar)**.** Let i, j be the minimal indices s.t. P_i, Q_j are not trivial. It holds that

$$Ham(P^m, Q^\infty) \geq Ham(P_i^m, Q_i^\infty) \geq m(a'+1) - 2 + \left\lfloor \frac{mb'}{q'} \right\rfloor = m(a+1) - 2 + \left\lfloor \frac{mb}{q} \right\rfloor$$

Which is exactly the required expression, with the LCM occurring more often. ⌟

6.2 Candidates for periods

▶ **Definition 29.** *Let T be a text, and P a string. We say that P is a* periodic seed *(or* seed*) of T, if $Ham(T, P^\infty) = k \to t \geq 2kp$ and P is non-periodic.*

▶ **Lemma 30.** *Let P, Q be non-trivial strings of co-prime lengths (i.e., $\gcd(p, q) = 1$), then $Ham(P^q, Q^p) \geq d(p-2) + q$, where d is the number of occurrences of the less frequent character in either P or Q.*

Proof. Let P, Q be non-trivial strings of co-prime lengths p, q. Let σ_S be the number of occurrences of a character σ in a string S.

In the LCM, i.e., when the strings P, Q appear enough times to perfectly align with each other, each characters of P aligns with each character of Q exactly once (using basic abstract algebra properties). It means that $Ham(P^q, Q^p) = \alpha_P \beta_Q + \beta_P \alpha_Q$.

Seeing that P, Q are both binary strings implies $\beta_P = p - \alpha_P$, and $\beta_Q = q - \alpha_Q$.

Assume $\alpha_Q \geq \beta_Q$, and that $\beta_Q \geq d \geq 1$.

$$\begin{aligned} Ham(P^q, Q^p) &= \alpha_P \beta_Q + \beta_P \alpha_Q \\ &= \alpha_P (q - \alpha_Q) + (p - \alpha_P) \alpha_Q \\ &\geq_1 (p-1)(q - \alpha_Q) + 1 \alpha_Q \\ &= (p-1)d + (q-d) \\ &= d(p-2) + q \end{aligned}$$

(1) is true since $a_Q \geq b_Q$, so be minimizing b_P we minimize the expression. ◀

▶ **Lemma 31.** *For alien strings Q, P s.t. $q \leq \frac{p}{4}$, $\gcd(p,q) = 1$ and $p = aq + b, 2 \leq m < q$,*

$$Ham(P^{nq+m}, Q^\infty) \geq n(p+q-2) + m\left\lceil \frac{p}{q} \right\rceil - 2 + \left\lfloor \frac{mb}{q} \right\rfloor$$

Proof. Using Theorem 12, we only need to show this is true when $d \geq 2, m \leq d$, where d is $(t_b - t_w) \mod q$, as defined. Because the case where $d = 1 \vee m \geq 4 \vee m > d$ is proved regardless of m, d and when $m \leq 1$ the expression can simply evaluate to 0. If $q \leq \frac{p}{4}$, then either $m \geq 4$ or $n \geq 1$. One case is already proven, so we can presume $n \geq 1$. Naturally, $m \leq 3$, as otherwise the lemma is trivially proven.
By Lemma 30, the distance at the LCM is $d(p-2)+q$. Consider the distance at the LCM in the original expression: $p+q-2$. The difference between these values is $(d-1)(p-2) \geq p-2$. This means the number of mismatches is increased by at least $p-2$ every LCM.

By Theorem 12, $Ham(P^m,Q^\infty) \geq m(a+1)–2$, and we attempt to show that $Ham(P^m,Q^\infty) \geq m(a+1) - 2 + \left\lfloor \frac{mb}{q} \right\rfloor$, leaving us to show that the additional $p-2$ mismatches are greater than $\left\lfloor \frac{mb}{q} \right\rfloor$. Evaluate:

$$p - 2 = aq + b - 2 \geq b > \left\lfloor \frac{mb}{q} \right\rfloor$$ ◀

▶ **Theorem 32.** *Let T be a text, and let P be a seed of T such that $p \leq \frac{t}{4}$, then for all strings Q s.t. $q < p, q \nmid p$, then Q can not be a seed of T.*

Proof. Let P be a seed of T of length $p \leq \frac{t}{4}$, and assume Q is a seed of T, and $q < p, q \nmid p$. In this proof, we use the same notations as in Theorem 31.
Set $occ_p = \left\lfloor \frac{t}{p} \right\rfloor = nq + m, 0 \leq m < q$, and set $p = aq + b, 1 \leq b < q$. $nq + m \geq 4$.
Obviously, $t < (nq + m + 1)p$. Therefore the maximum number of mismatches between T and P^∞, marked as k_p is at most $\left\lfloor \frac{nq+m}{2} \right\rfloor$. The number of occurrences of Q in T is $occ_q = \left\lfloor \frac{t}{q} \right\rfloor = \left\lfloor \frac{(nq+m+1)p-1}{q} \right\rfloor$, and thus the number of mismatches between T and Q^∞, marked as k_q accordingly is at most $\left\lfloor \frac{occ_q}{2} \right\rfloor$.

By the triangle inequality, $Ham(P^{nq+m}, Q^\infty) \leq Ham(T, P^\infty) + Ham(T, Q^\infty) \leq k_p + k_q$. We will show that $Ham(P^{nq+m}, Q^\infty) > k_p + k_q$, which will lead to a contradiction.

We split the proof into two cases - in the first case P, Q are aliens, and in the second, P, Q are similar (See Definition 27).

6.2.1 Proof for alien strings

Begin by evaluating the minimal number of errors:

$$Ham(P^{nq+m}, Q^\infty) \geq n(p+q-2) + m(a+1) - 2 + \left\lfloor \frac{mb}{q} \right\rfloor$$

Since we required $p \leq \frac{t}{4}$, then $n \geq 1$ or $m \geq 4$. We consider both cases.

▶ **Case 1** ($n = 0, m \geq 4$). Begin by re-evaluating the previous expressions by setting $n = 0$, which will lead to significantly shorter expressions.

$$Ham(P^{nq+m}, Q^\infty) = n(p+q-2) + m(a+1) - 2 + \left\lfloor \frac{mb}{q} \right\rfloor = m(a+1) - 2 + \left\lfloor \frac{mb}{q} \right\rfloor \quad (1)$$

$$occ_p = m \quad (2)$$

$$occ_q = \frac{\left\lfloor \frac{(m+1)p-1}{q} \right\rfloor}{2} \quad (3)$$

Again split into two cases:

1. $b \geq \frac{q}{2}$
2. $b < \frac{q}{2}$

▶ **Case 1.1** ($b \geq \frac{q}{2}$)**.** In this case, $\left\lfloor \frac{mb}{q} \right\rfloor \geq \left\lfloor \frac{mq}{2q} \right\rfloor = \left\lfloor \frac{m}{2} \right\rfloor$. Seeing that the latter is exactly the value of k_p, let's us subtract $\left\lfloor \frac{m}{2} \right\rfloor$ from both sides of the equation, which leads to the following inequality:

$$m(a+1) - 2 > k_q = \left\lfloor \frac{\left\lfloor \frac{(m+1)p-1}{q} \right\rfloor}{2} \right\rfloor$$

Given that we required $b \geq \frac{q}{2}$, we will evaluate $\left\lfloor \frac{(m+1)p-1}{q} \right\rfloor$ in worst-case settings in terms of mismatches, i.e. $b = q - 1$.

$$\left\lfloor \frac{(m+1)p-1}{q} \right\rfloor \leq \left\lfloor \frac{(m+1)((a+1)q-1)-1}{q} \right\rfloor = \left\lfloor \frac{q(m+1)(a+1)-m-2}{q} \right\rfloor \leq (a+1)(m+1)-1$$

What left to prove is $m(a+1) - 2 > \left\lfloor \frac{(a+1)(m+1)-1}{2} \right\rfloor$.

$$\begin{aligned} m(a+1) - 2 > \left\lfloor \frac{(a+1)(m+1)-1}{2} \right\rfloor &\rightarrow 2m(a+1) - 4 > (a+1)(m+1) - 1 \\ &\rightarrow 2am + 2m - 4 > am + a + m + 1 - 1 \\ &\rightarrow am + m - a - 4 > 0 \\ &\rightarrow a(m-1) + m - 4 > 0 \\ &\rightarrow 2m - 5 > 0 \rightarrow m \geq 3 \end{aligned}$$

$\lrcorner$

▶ **Case 1.2** ($b < \frac{q}{2}$)**.** In this case, we suppose that $\left\lfloor \frac{mb}{q} \right\rfloor = 0$, and again evaluate $\left\lfloor \frac{(m+1)p-1}{q} \right\rfloor$ in worst-case settings ($b = \frac{q}{2}$).

$$(m+1)p - 1 \leq (m+1)((a+0.5)q - 1) - 1 \leq (m+1)(a+0.5)q - 1$$

Using the above result, the inequality $\left\lfloor \frac{(m+1)p-1}{q} \right\rfloor \leq (m+1)(a+0.5) - 1$ holds, and thereby the resulting inequality is

$$m(a+1) - 2 > \left\lfloor \frac{(m+1)(a+0.5)-1}{2} \right\rfloor + \left\lfloor \frac{m}{2} \right\rfloor$$

Further evaluation leads to

$$\begin{aligned} & m(a+1) - 2 > \left\lfloor \frac{(m+1)(a+0.5)-1}{2} \right\rfloor + \left\lfloor \frac{m}{2} \right\rfloor \\ &\rightarrow 4m(a+1) - 8 > (m+1)(2a+1) - 2 + 2m \\ &\rightarrow 4am + 4m > 2am + m + 2a + 2m + 1 + 8 - 2 \\ &\rightarrow 2am + m - 2a \geq 8 \\ &\rightarrow 2a(m-1) + m \geq 8 \\ &\rightarrow 2 \cdot 3 + 4 \geq 8 \qquad \text{(Setting } a = 1, m = 4\text{)} \end{aligned}$$

$\lrcorner$

▶ **Case 2** ($n \geq 1, m \in \{0,1\}$)**.** This case is rather complicated. Begin by examining the case $m = 0$ which is the easier one.

▶ **Case 2.1** ($m = 0$)**.** Recall that $nq + m \geq 4 \rightarrow nq \geq 3 \rightarrow q \geq 3 \vee n \geq 2$. We require $nq \geq 3$ and not $nq \geq 4$, so we will also be able to use the expression when $m = 1$.
In this case, the number of occurrences of Q in T is $occ_q \leq np + a$, so we need to show that $n(p + q - 2) > \lfloor \frac{np+a}{2} \rfloor + \lfloor \frac{nq}{2} \rfloor$. We will multiply it by 2 to obtain

$$\begin{aligned}&2n(p + q - 2) > np + a + nq\\&\rightarrow n(p + q - 4) \geq a + 1\\&\rightarrow n((a + 1)q - 3) \geq a + 1\\&\rightarrow_1 n(2q - 3) \geq 2 \rightarrow q \geq 3 \vee n \geq 2\end{aligned}$$

⌟

(1) - Because we are incrementing a increments the left hand-side by qn, and the right hand-side by only 1 ($qn > 1$).

▶ **Corollary 33.** $Ham(Q^{a+m+1}, P^2) \geq 2m, \quad m < a$

Proof. First, define $P_{\rightarrow i}$ as the string cyclic shift of P i positions to the right.
Consider the hamming distance $Ham(Q^{m+1}, Q^{m+1}_{\rightarrow(q-b)})$. The first string is aligned with the first repetition of P, the other is aligned with the second repetition of P.
Since there are m full repetitions of Q in both strings, there are at least $2m$ mismatches. Using the triangle inequality:

$$2m \leq Ham(Q^{m+1}, Q^{m+1}_{\rightarrow(q-b)}) \leq Ham(Q^{m+1}, P) + Ham(P, Q^{m+1}_{\rightarrow(q-b)}) \leq Ham(Q^{a+m+1}, P^2)$$

◀

▶ **Corollary 34.** *If Q is a seed of a text with tail size $t = t - (nq + 1)p$ s.t. $t \geq 2q - b$, then Q is also the seed of a text with tail size $2q - b - 1$.*

The proof is trivial: any additional occurrence of Q in T will cause at-least 2 more verified mismatches. We allow one mismatch for 2 occurrences so it is always possible to reduce the number of mismatches by 2 and the number of occurrences by 1.

▶ **Lemma 35.** *If $Ham(Q^{a+1}, P^2) = 0$, then $b \geq 2$*

Proof. By definition, $q(a + 1) > p$. If $Ham(Q^{a+1}, P^2) = 0$, then $P = Q^a Q[..b]$, and that $P[..q - b] = Q[b..] \rightarrow Q[..q - b] = Q[b..]$.
Assume $b = 1$. This means $Q[..q - 1] = Q[1..]$, or

$$Q[0] = Q[1], Q[1] = Q[2], ...Q[-2] = Q[-1]$$

Which implies that Q is a trivial periodic string, contradiction. ◀

▶ **Case 2.2** ($m = b = 1$)**.** In this case, Q can occur at most a times in the tail without creating an extra mismatch. Thus, we will suppose it occurs exactly a times in the tail.
$n(p + q - 2) > \lfloor \frac{np+a}{2} \rfloor + \lfloor \frac{nq+1}{2} \rfloor$

This is almost exactly the same as the case where $m = 0$, with the difference of k_p, which changed from $\lfloor \frac{nq}{2} \rfloor$ to $\lfloor \frac{nq+1}{2} \rfloor$. If either n or q are even, it is exactly the same expression as before, and accordingly we presume $q \neq 2 \rightarrow q \geq 3 \rightarrow p \geq 4$.

Evaluating the expression in a similar manner to case 2.1 will lead to the inequality $n(2q - 3) \geq 3$, and by setting $q = 3, n = 1$, we get $1 \cdot (2 \cdot 3 - 3) = 3 \geq 3$. ⌟

▶ **Case 2.3** ($m = 1, b > 1$)**.** This case is trivial. Requiring $b \geq 2$ results in $p \geq q + 2$, which in turn causes an additional mismatch on the LCM expression $p + q - 2$, but creates only one additional occurrence for Q, and the problem can be reduced to previous case. ⌟

▶ **Case 3** ($n \geq 1, m \geq 2$)**.** This is the simplest case of the three. We have $p > q \geq 3$, as $m < q < p$, and $m \geq 2$.

Proof. The length of T is at most $(nq + m + 1)p - 1$. The string Q occurs in nqp characters exactly np times. And $(m + 1)p - 1 = (m + 1)(aq + b) - 1 = maq + mb + aq + b - 1$, which contains $a(m + 1) + \left\lfloor \frac{b(m+1)-1}{q} \right\rfloor$ instances of Q.
The inequality $\left\lfloor \frac{b(m+1)-1}{q} \right\rfloor \leq \left\lfloor \frac{bm}{q} \right\rfloor + 1$ holds, and as a result we claim that the number of occurrences of Q in T is $\left\lfloor \frac{b(m+1)-1}{q} \right\rfloor + 1$.

The expression $\left\lfloor \frac{bm}{q} \right\rfloor$ also appears in $Ham(P^m, Q^\infty)$, so we can subtract it from both sides, which will simplify the expression to $n(p+q-2)+m(a+1)-2 > \left\lfloor \frac{np+a(m+1)+1}{2} \right\rfloor + \left\lfloor \frac{nq+m}{2} \right\rfloor$. Multiply it by 2, and get the following:

$$\begin{aligned}
&2n(p + q - 2) + 2m(a + 1) - 4 > np + a(m + 1) + 1 + nq + m \\
&\to n(p + q - 4) + 2am + 2m - 4 > am + a + 1 + m \\
&\to n((a + 1)q - 3) + am - a + m \geq 6 \\
&\to_1 n(2q - 3) + 2m - 1 \geq 6 \\
&\to_2 3n + 2m \geq 7 \to 3 \cdot 1 + 2 \cdot 2 \geq 7
\end{aligned}$$

(1) - Setting a to minimum ($a = 1$)
(2) - Setting q to minimum ($q = 3$) ◀

⌟

6.2.2 Proof for similar strings

We now present the proof of the theorem when $\gcd(p, q) > 1$, and $i = j$.

We use the definition of P_i, Q_i, p', q' as usual, and also define T_i as $T[i]T[i + g][T + 2g]....$ Because t is not necessarily divisible by g some T_i might be shorter than the others.
Let i be the index such that both P_i, Q_i are non-trivial. The inequality $\forall j, T_j \geq 4p'$ holds as $t \geq 4pg$.
As k_p is the maximal number of full occurrences of P in T, the number of occurrences of P_i in T_i is at least k_p. Proof for k_q is the same.

We now have Q_i, P_i, T_i, where $P_i \leq \frac{|T_i|}{4}$ and $gcd(P_i, Q_i) = 1$, and therefore P_i and Q_i cannot both be seeds of T_i, or in other words, $Ham(p_i^{nq+m}, q_i^\infty) > k_q + k_p$. In contrast, we know that $Ham(p^{nm+q}, q^\infty) \geq Ham(p_i^{nq+m}, q_i^\infty) > k_q + k_p$, and P, Q cannot both be seeds. ◀

▶ **Definition 36** (Binary Renaming Function)**.** *A binary renaming function is a function from general alphabet to binary alphabet, $\delta : \Sigma \to \{a, b\}$. The generalized renaming function $\delta^* : \Sigma^* \to \{a, b\}^*$ is defined in the standard way:*
1. $\delta^*(\epsilon) = \epsilon$
2. $\delta^*(P) = \delta(P[0])\delta^*(P[1..])$

▶ **Lemma 37.** *Let P, Q be non-trivial strings over Σ. W.l.o.g $p \geq q$. $\forall \delta^*, m$, $Ham(P^m, Q^\infty) \geq Ham(\delta^*(P)^m, \delta^*(Q)^\infty)$*

Proof. Let e_i be an indicator on mismatch on index i between P^m and Q^∞, and e'_i be defined respectively for $\delta^*(P^m)$, $\delta^*(Q^\infty)$. Since $e_i = 0$ indicates a match, then $e'_i = 0$ as well, which implies $e'_i \le e_i$. The contrary is not true. The direct implication is our lemma. ◀

▶ **Corollary 38.** *Theorem 12 is true for general alphabet.*

▶ **Corollary 39.** *Theorem 43 is true for general alphabet.*

Proof. Assume there are $P, Q \in \Sigma^*$ that contradict 43. This means that $Ham(T, P^\infty) + Ham(T, Q^\infty) \le k_p + k_q$, but using Lemma 37,

$$\exists \delta^*, Ham(T, P^\infty) + Ham(T, Q^\infty) \ge Ham(\delta^*(T), \delta^*(P)^\infty) + Ham(\delta^*(T), \delta^*(Q)^\infty) >_1 k_q + k_p$$

And (1) is true directly from Theorem 43. ◀

6.3 The Number of Candidates

▶ **Lemma 40.** *Let T be text, and let P, Q be periodic seeds of T, such that $p > q, q \nmid p, p \le \frac{t}{3}$. Then, $\exists g$ s.t. $p = 3g, q = 2g$.*

The proof appears in the full version of the paper,

▶ **Corollary 41.** *Given a text T, there is at most one periodic seed P of length $\frac{t}{4} < p \le \frac{t}{3}$.*

▶ **Corollary 42.** *If P, Q are seeds of a text T and $p = q$ then $Ham(P, Q) \le 1$*

Proof. Assume $Ham(P, Q) \ge 2$, and that $\ell = \left\lfloor \frac{t}{p} \right\rfloor$. Using the definition of a seed, $Ham(P^\ell, Q^\ell) \le Ham(T, P^\infty) + Ham(T, Q^\infty) \le \ell$, at the same time $Ham(P^\ell, Q^\ell) = 2\ell$, contradiction. ◀

▶ **Theorem 43.** *There are at most $2\log_3 n$ candidates of length $\ell \le \frac{t}{3}$*

Proof. We begin by proving a useful lemma regarding the number of candidates of divisible length.

▶ **Lemma 44.** *If there are two candidates of length g, there is at most one candidate of length $2g$.*

The proof appears in the full version of the paper.

▶ **Corollary 45.** *If all of the periods of length $\ell \le n$ divide n, then there are at most $2\log_3 n$ periods of length $\le n$.*

The proof appears in the full version of the paper. ◀

▶ **Corollary 46.** *There are at most 2 periods that are canonical.*

Proof. As we have proved, if there are 3 different periods, then the length of the shortest period, divides the length of the longer periods, and by simple induction the minimal period's length divides the lengths of the rest of the periods, and is canonical to them.
If there are no 3 different periods then we can call all of the periods canonical. ◀

▶ **Corollary 47** (Generalized gcd-theorem). *Let T be a text, and let P, Q be seeds of T that occur at least 3 full times in T, then T has a seed of length $\gcd(p, q)$.*

7 The Error Upper Bounds Hierarchy

The previously known [3] upper error bound was $\lfloor \frac{n}{(2+\epsilon)p} \rfloor$ and led to $O(\log_{1+\epsilon} n)$ period candidates. To achieve a hierarchy of upper bounds, we follow similar techniques to those of [3]. The innovation is the new *Hamming distance* error bounds that we define. All the upper bounds in this section are of (n, p, ϵ)-type. We do not get rid of ϵ at the moment. However, we feel that by methods similar to those of Section 6, the ϵ can be eliminated from the error bounds.

▶ **Lemma 48.** *Let T be the input text. For any period P, assume that $Ham(T, T_P) \leq \frac{n}{(2+\epsilon)\cdot(2^{(i)})^{((\log^{(i)} p)^{1+\epsilon})}}$. Then there are $O(\log^{(i+1)} n)$ candidates for P, for any $i \geq 0$. Note that, for $i = 0$, the upper bound is $Ham(T, T_P) \leq \frac{n}{(2+\epsilon)\cdot p^{1+\epsilon}}$*

▶ **Lemma 49.** *Let T be the input text. For any period P, assume that $Ham(T, T_P) \leq \frac{n}{(2+\epsilon)^{p+1}}$. Then there are $O(\log^* n)$ candidates for P*

▶ **Lemma 50.** *Lemmas 48 and 49 have tight examples.*

8 Conclusions and Open Problems

We showed that $\lfloor \frac{n}{2p} \rfloor$ is a tight upper error bound and it results in $2\log_3 n$ candidates if P fully occurs at least 3 times in T. Otherwise, if P fully occurs twice (maybe with a tail) then we provide an example with $\Omega(n)$ candidates.

It is easy to show that our result can be generalized to *c-pseudo local metrics* [3]. Such a metric Δ behaves like *Hamming distance* in the manner that for every pair of equal length strings T and S, $\Delta(T, S) \leq c \cdot Ham(T, S)$. An example of such a metric is *swap distance*, where the distance counted by the number of swaps of two consecutive letters (each letter can only participate in a single swap). It is easy to verify that $\lfloor \frac{n}{2cp} \rfloor$ is a tight upper bound, and the result of *canonic period seed* can also be applied to such metrics. By a similar fashion, replacing $(2+\epsilon)$ by $(2c+\epsilon)$ in the hierarchy upper bounds allows the hierarchy generalization to *c-pseudo local metrics.*

In this paper, corruptions are of the replacement type. That is, the error count is the number of changed letters. It is of interest to analyze different types of error distance metrics besides the *Hamming Distance*. One example of such a metric, that has been considered in the recovery algorithms literature, is the *Edit Distance* [2, 14].

It is of interest to apply the ideas of section 6 to the $\log^{(i)}$ hierarchy and eliminate the dependence on ϵ.

Finally, can the hierarchy be extended to the other side? Come up with tight error bounds that lead to a set of n^ϵ period candidates, where $\epsilon < 1$.

References

1 S. Aluru, A. Apostolico, and S. V. Thankachan. Efficient alignment free sequence comparison with bounded mismatches. In *Proc. 19th Research in Computational Molecular Biology Conference, RECOMB*, pages 1–12, 2015.

2 A. Amir, M. Amit, G.M.Landau, and D. Sokol. Period recovery of strings over the hamming and edit distances. *Theortetical Computer Science*, 710:2–18, 2018.

3 A. Amir, E. Eisenberg, A. Levy, E. Porat, and N. Shapira. Cycle detection and correction. *ACM Trans. Alg.*, 9(1):13, 2012.

4 A. Amir and D. Keselman. Maximum agreement subtree in a set of evolutionary trees - metrics and efficient algorithms. *Proc. FOCS 94*, pages 758–769, 1994.

5 A. Amir, A. Levy, M. Lewenstein, R. Lubin, and B. Porat. Can we recover the cover? In *Proc. 28st Annual Symposium on Combinatorial Pattern Matching (CPM)*, LIPICS, 2017.

6 A. Amir, M. Lewenstein, and E. Porat. Approximate subset matching with “don’t care”s. In *Proc. 12th ACM-SIAM Symp. on Discrete Algorithms (SODA)*, pages 305–306, 2001.

7 R.L. Cann, M. Stoneking, and A.C. Wilson. Mitochondrial DNA and human evolution. *Nature*, 325(6099):31–36, 1987.

8 M. Farach, S. Kannan, and T. Warnow. A robust model for finding optimal evolutionary trees. *Proc. 25th Annual ACM Symposium on the Theory of Computing*, pages 137–145, 1993.

9 M. Farach, T. M. Przytycka, and M. Thorup. Computing the agreement of trees with bounded degrees. *Proc. 3rd European Symposium on Algorithms*, pages 381–393, 1995.

10 N. J. Fine and H. S. Wilf. Uniqueness theorems for periodic functions. *Proc. Amer. Math. Soc.*, 16:109–114, 1965.

11 S. Har-Peled and S. Mahabadi. Proximity in the age of distraction: Robust approximate nearest neighbor search. In *Proc. 28th ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1–15, 2017.

12 S. Heydrich and A. Wiese. Faster approximation schemes for the two-dimensional knapsack problem. In *Proc. 28th ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 79–98, 2017.

13 C. Kalaitzis. An improved approximation guarantee for the maximum budgeted allocation problem. In *Proc. 27th ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1048–1066, 2016.

14 T. Kociumaka, J. Radoszewski, W. Rytter, J. Straszyński, T. Waleń, and W. Zuba. Faster recovery of approximate periods over edit distance. In *Proc. 25th International Symposium on String Processing and Information Retrieval (SPIRE)*, LNCS, pages 233–240. Springer, 2018.

15 L.A.B. Kowada, D. Doerr, S. Dantas, and J. Stoye. New genome similarity measures based on conserved gene adjacencies. In *Proc. 20th Research in Computational Molecular Biology Conference, RECOMB*, pages 204–224, 2016.

16 A. Ojewole, J.D. Jou, V.G. Fowler, and B.R. Donald. Bbk* (branch and bound over k*): A provable and efficient ensemble-based algorithm to optimize stability and binding affinity over large sequence spaces. In *Proc. 21st Research in Computational Molecular Biology Conference, RECOMB*, pages 157–172, 2017.

17 A. Sobih, A. I. Tomescu, and V. Mäkinen. Metaflow: Metagenomic profiling based on whole-genome coverage analysis with min-cost flows. In *Proc. 20th Research in Computational Molecular Biology Conference, RECOMB*, pages 111–121, 2016.

18 M. Steel and T. Warnow. Kaikoura tree theorems: Computing the maximum agreement subtree. *Information Processing Letters*, 48(2):77–82, 1993.

19 M. A. Steel and D. Penny. Distributions of tree comparison metrics - some new results. *Syst. Biol.*, 42:126–141, 1993.

20 E. Ukkonen. A linear-time algorithm for finding approximate shortest common superstrings. *Algorithmica*, 5:313–323, 1990.

21 J. C. Venter and Celera Genomics Corporation. The sequence of the human genome. *Science*, (291):1304–1351, 2001.

22 C. Wulff-Nilsen. Approximate distance oracles for planar graphs with improved query time-space tradeoff. In *Proc. 27th ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 351–362, 2016.