AVOIDING ABELIAN SQUARES IN INFINITE PARTIAL WORDS

by

William Severa

A Thesis Submitted to the Faculty of

The Wilkes Honors College

in Partial Fulfillment of the Requirements for the Degree of

Bachelor of Arts in Liberal Arts and Sciences

with a Concentration in Mathematics

Wilkes Honors College of

Florida Atlantic University

Jupiter, Florida

May 2010

AVOIDING ABELIAN SQUARES IN INFINITE PARTIAL WORDS

by

William Severa

This thesis was prepared under the direction of the candidate's thesis advisor, Dr. Terje Hoim, and has been approved by the members of his supervisory committee. It was submitted to the faculty of The Honors College and was accepted in partial fulfillment of the requirements for the degree of Bachelor of Arts in Liberal Arts and Sciences.

SUPERVISORY COMMITTEE:

____
Dr. Terje Hoim

____
Dr. William O'Brien

____
Dean, Wilkes Honors College

____
Date

Acknowledgements

Foremost, I thank my advisor, Dr. Terje Hoim, for her wonderful direction and instruction. Her aid has bettered not only this thesis, but myself as a student and as a mathematician. I also acknowledge Dr. William O'Brien. His careful reading and input has vastly improved the quality of my work. Dr. Francine Blanchet-Sadri, Jane Kim, Sean Simmons, and Robert Mercaş all provided invaluable help while at the REU program of the University of North Carolina Greensboro, which has been critical for the success of this research.

Abstract

Author:	William Severa
Title:	Avoiding Abelian Squares in Infinite Partial Words
Institution:	Harriet L. Wilkes Honors College, Florida Atlantic University
Thesis Advisor:	Dr. Terje Hoim
Concentration:	Mathematics
Year:	2010

Famous mathematician Paul Erdös conjectured the existence of infinite sequences of symbols where no two adjacent subsequences are permutations of one another. It can easily be checked that no such sequence can be constructed using only three symbols, but as few as four symbols are sufficient. Here, we expand this concept to include sequences that may contain 'do not know' characters, called holes. These holes make the undesired subsequences more common. We explore both finite and infinite sequences. For infinite sequences, we use iterating morphisms to construct the non-repetitive sequences with either a finite number of holes or infinitely many holes. We also discuss the problem of using the minimum number of different symbols.

Contents

List of Figures . . . . **vi**

List of Tables . . . . **vii**

List of Algorithms . . . . **viii**

1 Introduction **1**

2 Preliminaries **3**

2.1 Word and Partial Word Basics 3

2.2 Squares and Abelian Squares 6

3 Computer Tools **8**

3.1 Checking for Abelian Square-Freeness 8

3.2 Generating Abelian Square-Free Words 11

4 Finite-Length Partial Words **12**

5 Infinite-Length Partial Words **16**

6 Special Case Alphabets **31**

7 Conclusion **34**

8 References **36**

A Symbol Glossary **39**

List of Figures

1 Vector representation of $e_1e_2e_1e_2$. 32

2 The square and abelian square $e_1e_2e_1e_3e_1e_2e_1e_3$. Positive z-axis is into the page. 33

List of Tables

1 Number of abelian square-free words of lengths 2, 3, 5, 10, and 15 for alphabets of 3, 4, 5, and 6 letters. 11

2 Required alphabet sizes for infinite and bi-infinite words. 34

List of Algorithms

1 A test for abelian square. 9

2 Alternate test for an abelian square. 10

3 Method for determining abelian square-freeness. 11

1 Introduction

Information represented by a stream of symbols such as DNA sequences, binary code, DVD decipher keys, or plain English sentences is common in everyday life. But, how can we characterize these information streams? In mathematics and theoretical computer science, a stream of symbols is called a word. A word is simply a sequence of characters, called letters. By using abstract definitions, results can be applied as long as the definitions are met, regardless of context. For example, a theorem about words is valid in reference to both DNA sequences and binary code, because both are examples of our abstract words. Also, because of the abstract nature, direct applications are often still undiscovered. However, the properties of words are extremely interesting and carry mathematical merit.

In the early 1960s famous mathematician Paul Erdös suggested the existence of a particular class of words that are special in that, despite being infinitely long, no two adjacent subsections are permutations of one another [8]. At this time, less-constrained infinite-length words, where adjacent subsections could not be equal, had already been studied [18, 19]. However, Alexander Evdokimov was the first to construct the words that Erdös has postulated [9]. His word used twenty-five unique characters. It was not until 1992 that theoretical computer scientist Veikko Keränen established that as few as four letters could be used [13]. The words that Erdös conjectured have become known as being infinitely-long and "abelian square-free."

In this thesis we expand the concept of "abelian squares" by defining and exploring infinitely-long abelian square-free *partial* words. Partial words are similar to regular or full words except that any number of letters may be missing.

The concept of partial words is easily applied to imperfect DNA sequences [11]. DNA sequences can be viewed as words but often there are many gaps in them. These gaps can be represented mathematically as gaps in partial words. We call these gaps "holes." The main question of this theory is whether or not infinite abelian square-free partial words exist. Or to the contrary, does a partial abelian square-free word necessarily have a finite length? How many distinct characters are needed to construct an infinite abelian square-free partial word? If such a construction exists, how many holes can an infinite abelian square-free partial word have? The ultimate goal would be to construct an abelian square-free word where we have removed an infinite number of letters.

Fortunately, we have been able to adapt combinatorics-based techniques, such as those employed by Peter Pleasants, to the case of partial words [17]. As such, we have avoided the complicated set-theoretic approach. Instead, computer-aided combinatorics have provided the necessary tools to characterize this new set of words. Generally, we use the computer programs for two different tasks: checking if a word is abelian square-free and generating abelian square-free words.

The thesis is organized as follows. The basics are given in Section 2. Section 3 consists of the computer algorithms used in the research. Section 4 describes finite-length abelian square-free partial words. Particularly, we establish that over a three-letter alphabet, abelian square-free words have bounded length. Furthermore, we explore the conditions where abelian square-free words over a four-letter alphabet have bounded length. In Section 5, we switch to infinitely long words. By using morphisms, we begin by constructing an infinite abelian square-free partial word containing one hole using five different characters. We will then use a similar process to create an infinite abelian square-free partial word using only four characters. Furthermore, we will prove that an infinite abelian square-free word over four letters can have at most one hole. However, more characters allow for words with more holes. In fact, we will construct an infinite abelian square-free word over seven letters which has an infinite number of holes. Finally, we will modify the construction so that the word uses only six letters. In Section 6, we discuss two special case alphabets that provide unique interpretations. Section 7 includes concluding remarks and suggestions for further research.

2 Preliminaries

2.1 Word and Partial Word Basics

The basic set we will be working with is an **alphabet**, a finite set of symbols or letters. The sets $\{a, b, c\}$ and $\{0, 1, 2\}$ are examples of alphabets. The **cardinality** of an alphabet A is denoted by $||A||$. For example, if $A = \{a, b, c, d\}$, then A is a four-letter alphabet ($||A|| = 4$). While we will form our alphabets with the letters a, b, c, and so on, Section 6 shows how particular alphabets can have interesting characteristics.

The letters of an alphabet can be concatenated together to form words. A **word** x is a sequence of letters from an alphabet concatenated together. Let A be an alphabet $\{a, b, c\}$. The following are examples of words: ab, bac, $abaac$. Words made from an alphabet A are elements in the set of all possible words A^*. DNA sequences and binary code are two real-world examples of words. Words may be empty, finite length or infinite length and the length of a word x is denoted by $|x|$. We represent the unique, empty word as ε, defined as $|\varepsilon| = 0$. With the empty word and concatenation, we have a monoid. The **reversal** of a word

$x = a_0a_1 \dots a_n$ is simply the word written backwards and is denoted by $rev(x) = a_na_{n-1} \dots a_1a_0$. Furthermore, it is useful to denote the number of occurrences of a word a within x with the notation $|x|_a$. Also, from time to time we will use **parikh vectors** which represent the number of each letter in a particular word. Specifically, let $A = \{a_0, a_1, \dots a_n\}$ be an alphabet for some $n \in \mathbb{N}$ and let $x \in A^*$ be a word. Then, the parikh vector of x, $\psi(x)$ is defined as

$$\psi(x) = \langle |x|_{a_0}, |x|_{a_1}, \dots, |x|_{a_n} \rangle.$$

It is also necessary to define a type of mapping called a morphism. Given alphabets A and B (usually but not necessarily $A = B$), a **morphism** is a function $A^* \to B^*$ which maps words onto words by concatenating the images of each letter in the preimage. For example, take the simple morphism ϕ over $A = \{a, b\}$ such that

$$\phi(a) = b$$
$$\phi(b) = a.$$

This morphism simply maps a to b and b to a. This is often referred to, in the context of binary values, as a bit flip or bitwise negation. As an example of the above morphism, $\phi(abaabba) = babbaab$. The morphism ϕ is said to be **uniform** when the images are all equal-length. A non-uniform morphism would be, for example,

$$\chi(a) = ab$$
$$\chi(b) = a.$$

So, with the non-uniform morphism we get $\chi(abaabba) = abaababaaab$. As a shorthand notation, we write repeated applications of a morphism as an exponent, that is $\phi(\phi(x)) = \phi^2(x)$. Morphisms are particularly useful for creating infinite and arbitrarily long words.

It is natural to think of a word as a function from the natural numbers (or whole numbers) to a particular alphabet. With this in mind, we assign indices to each of the letters of a word. For convenience, we begin with 0. We denote the letter of a word x at the index a by $x(a)$. Also, we will extend this notation to include intervals. That is, $x[a, b)$ is the a-th through (but not including) the b-th letters of x concatenated together. The resulting interval is a factor of x, but in this context we will refer to such as a **subword**.

In 1999, J. Berstel and L. Boasson expanded the study of words by introducing the concept of **partial words** [2]. Partial words include a provision for missing letters and this is useful for applications in word algorithms and DNA sequencing [11]. A **hole**, represented by $\diamond$, is an unknown character or a wildcard. This is different from a function being undefined at a point. In our case, the word is defined at the hole, we simply do not know what it is. The hole symbol, $\diamond$, is not a letter and so it is not an element of any alphabet. The set of all partial words made from an alphabet A is written $A^*_\diamond$. The words in $A^*_\diamond$ consist of characters from $A \cup \diamond$. As such, all full words are also partial words. Examples include $a\diamond$, *bac*, *ab*$\diamond$*ac*. We define the **domain set**, $D(x)$ of a word x as the set containing the positions of all letters and the **hole set**, $H(x)$, as the set containing the positions of all holes. For clarity, we may refer to a word that has an empty hole set as a **full word**.

For convenience we define new operations containment and compatibility. A word x is **contained** by a word y, denoted by $x \subset y$, if all elements in $D(x)$ are in $D(y)$ and $x(i) = y(i)$, for all $i \in D(x)$. Words x and y are said to be **compatible**, denoted by $x \uparrow y$, if there exists a word z such that $x \subset z$ and $y \subset z$.

2.2 Squares and Abelian Squares

Words x_0, x_1, x_2 are called **factors** of x if $x_0x_1x_2 = x$. Furthermore, a factor that begins a word, such as x_0 in the word x, is called a **prefix**. A factor that ends a word, such as x_2 above, is called a **suffix**. For example, let $x = abaca$. Then, *a*, *b*, *c*, *ab*, *ac*, *ba*, *ca*, *aba*, *bac*, *aca*, *abac*, *baca*, and *abaca* are all the factors of x. A word x_1x_2 is called a **square** if $x_1 = x_2$. Or in other words, x is a square if it is made of two equal subwords. For example, the word *ababbaba* has the squares *bb*, *bab*, *abab*, *baba*. Furthermore, a word which contains no squares as factors is called **square-free**. It is easily seen that a binary alphabet only has finite square-free words. However, using a ternary alphabet, one can already write infinite square-free words. Examples of this include variations on the Thue-Morse word such as *abcacbabcbac*... [16].[1]

Considered as an abstraction of a square, we say a word xy is an **abelian**

[1]The word *abcacbabcbac*... is reached in several steps. First, the morphism $\phi(a) = ab, \phi(b) = ba$ is applied infinitely often to a starting word a. This results in *abbabaabbabab*... which is known as the Thue-Morse word. Then, the inverse of the mapping $\chi(a) = abb, \chi(b) = ab, \chi(c) = a$ applied to the Thue-Morse word gives the square-free word *abcacbabcbac*....

square if the factors x and y are permutations of one another. The words *cabcab* and *cabcba* are both abelian squares. Abelian squares are less constrained than squares and squares are a subset of abelian squares. If we examine our previous example of the word *ababbaba*, all the regular squares listed above are abelian squares but, in addition, we notice that *abab* is the first half of an abelian square because it is directly followed by a factor *baba*. An alternative definition is that words x, y make an abelian square xy if and only if $\psi(x) = \psi(y)$. A morphism $\phi(x)$ is called an **abelian square-free morphism** if x being abelian square-free implies $\phi(x)$ is abelian square-free. Morphisms of this type have interesting characterizations and are, at times, studied independently [6].

We can use the concepts of containment and compatibility to extend the definitions of square and abelian square to the context of partial words as follows:

1. Two partial words x_0, x_1 form a **square** x_0x_1 if $x_0 \uparrow x_1$.

2. Two words $x_0, x_1 \in A_\diamond^*$ form an **abelian square** x_0x_1 if it is possible to substitute letters from A for each hole in such a way that x_0x_1 becomes a full abelian square. That is, two words $x_0, x_1 \in A_\diamond^*$ form an abelian square x_0x_1 if, after a substitution of letters of A for the holes in x_0 and x_1 to get x_0' and x_1', we have $\psi(x_0') = \psi(x_1')$.

Our definition of abelian square-free words now requires refinement. Note that any word containing a hole will necessarily contain the trivial abelian square $a\diamond$ or $\diamond a$, where a is some letter. Thus, we say a word $x \in A_\diamond^*$ is abelian square-free (in the context of partial words) provided it avoids all abelian squares except those of the forms $a\diamond$ and $\diamond a$, $a \in A$. Using these definitions we see that $aba\diamond$ is an abelian square. If we replace the hole with b, this word becomes *abab*, which is a square and abelian square. Actually, every length-4 word made from a binary alphabet contains a square.

The study of abelian square-free words is the main focus of this thesis. We will consider infinite abelian square-free words with both finitely many holes and with infinitely many holes. We will prove that at least a four-letter alphabet is required and sufficient for an infinite abelian square-free partial word with provided there is at least one hole. Although, a larger alphabet is needed to create an abelian square-free word with an infinite number of holes. We will show that at most a six-letter alphabet is required.

3 Computer Tools

When working with words from a combinatorics approach, computer software becomes extremely useful. Software can be used to create words with certain properties or check words for existing properties. In doing so, programs can even prove theorems by exhaustion, if there are a finite number of cases. However, proof by a computer program is impossible when there are an infinite or extremely large number of cases. A common solution in such situations is to prove mathematically that only certain bounded conditions need to be checked. In this section we detail the methods and algorithms used for various purposes including checking for abelian squares and creating abelian square-free words.

3.1 Checking for Abelian Square-Freeness

As stated before, a word is considered abelian square-free if it contains no abelian squares. Our program will systematically check growing factors in the word for abelian squares. In efforts to keep the code cohesive, we need first to be able to determine if a particular word is an abelian square. The pseudocode is listed in Algorithm 1. The first step checks that the length of a word is even, as an odd-lengthed word cannot be an abelian square. Then, we count the number of holes in the word. Finally, we compare the parikh vectors of the first half of the input word to the second half of the input word. This comparison is done component-wise. We subtract the difference for any given number from the number of holes since each hole can "correct" a one-letter difference. This algorithm completes in

Algorithm 1 A test for abelian square.

```
if input.length % 2 ≠ 0 then
  return false
end if
nHoles = 0
for all a ∈ input do
  if a = hole then
    nHoles++
  end if
end for
factor1 = input[0, input.length/2]
factor2 = input[input.length/2, input.length]
for {i = 0; i < alphabetSize; i++} do
  nHoles = nHoles − |ψ(factor1)[i] − ψ(factor2)[i]|
end for
return nHoles ≥ 0
```

$O(N)$ time.[2] The process of creating the parikh vectors is a simple count which is proportional to the length of the word. We make no claim that this algorithm is overly efficient. A faster algorithm may be very well possible. For the full word case, if we are selective in our alphabet, we can create an alternate, simpler,

[2] We use the standard $O(N)$ to represent 'Big O' notation. That is, the program's operation time is bounded by a linear function of the length of the input.

$O(N)$ algorithm. We do this by storing the letters as sequential prime numbers.[3] That is, if our original alphabet is $\{a, b, c\}$, we use $\{2, 3, 5\}$. Then, as shown in Algorithm 2, we can simply multiply the two halves of the input word. If the products match, they make an abelian square. This algorithm could be modified

Algorithm 2 Alternate test for an abelian square.

```
if input.length % 2 ≠ 0 then
  return false
end if
prod1 = 1;
prod2 = 1;
for {i = 0; i < input.length/2; i++} do
  prod1 = prod1 * input[i]
end for
for {i = input.length/2; i < input.length; i++} do
  prod2 = prod2 * input[i]
end for
return prod1 == prod2
```

for partial words by examining the prime factorization of the quotient of the two partial products. However, we do not expect it to be more efficient, in general, than Algorithm 1.

With the ability to check to see if a given word is an abelian square, we can now develop an algorithm to test if a word is abelian square-free. Although inefficient, we can do this exhaustively as shown in Algorithm 3. Remember, we have defined abelian square-free in such a way that abelian squares of length two are deemed trivial. So, first the algorithm checks to see if there are any double letter words. If not, possible abelian squares range from length-four to half the input word length. Factors of all these lengths are checked systematically starting at the beginning of the word. Provided the check for an abelian square is $O(N)$, checking a word

[3] Section 6 provides a further discussion of these alphabets.

for abelian square-freeness is bounded by $O(N^3)$.

Algorithm 3 Method for determining abelian square-freeness.

```
for all a ∈ alphabet do
  if input.contains(a + a) then
    return false
  end if
end for
for {length = 4; length < input.length/2; length = length + 2} do
  for {i = 0; i < input.length − lenght; i + +} do
    if isAbelianSquare(input[i, i + length]) then
      return false
    end if
  end for
end for
return true
```

3.2 Generating Abelian Square-Free Words

We will also use software to generate abelian square-free words. The number of abelian square-free words increases with alphabet size and word length (provided the alphabet is sufficiently large). A small selection of alphabet sizes, word lengths, and the corresponding number of abelian sqaure-free words are listed in Table 1.

n	$\|\|A\|\|$			
	3	4	5	6
2	6	12	20	30
3	12	36	80	150
5	30	264	1140	3,480
10	-	15,360	646,560	8M
15	-	286,296	273M	>3B

Table 1: Number of abelian square-free words of lengths 2, 3, 5, 10, and 15 for alphabets of 3, 4, 5, and 6 letters.

The lengths and sheer amount of abelian square-free words can lead to issues in computing resources, particularly memory, surprisingly early. For example, Table 1 lists 273 million length-15 abelian square-free words for a five-letter alphabet. This means there are approximately 4 billion characters. Using 8 bits per unicode character yields memory requirements of about 4 GiB. Possible solutions include working on smaller sets of words or caching to disk.

Abelian square-free words could be generated by creating all possible words and then only returning the abelian square-free ones. This method, however, is extremely inefficient. It creates and checks $||A||^n$ words, where A is the alphabet and n is the length of the words. Generating and checking this many words not only requires a long computation time but also exacerbates the memory requirement. A better method is to create shorter abelian square-free words and combine these. For example, if we want length-four words, we would check all combinations of two length-two words. If this example is for a four-letter alphabet, the original approach has us check $4^4 = 256$ possibilities, whereas 12 two-letter abelian square-free words combine to form 144 possibilities. If we omit repeats (since these are clearly abelian squares), we only need to check 132 words.

4 Finite-Length Partial Words

In this section, we explore finite-length abelian square-free partial words. Moreover, we look at the implications of using small alphabets (no larger than four letters) and the constraints on possible word lengths these alphabets introduce. Several subtle differences exist between full and partial words. Since a unary alphabet is trivial, and a binary alphabet is simplistic (any word with four characters contains an abelian square), we begin with a ternary alphabet. Many of the results in the following sections are to appear in *Abelian Square-Free Partial Words* by F. Blanchet-Sadri, J. Kim, R. Mercaş, W. Severa, and S. Simmons [3].

Theorem 1. *Over a three-letter alphabet, an abelian square-free partial word with at least one hole has a maximum length of six.*

While the statement can be easily shown using a computer program, it can also be proven directly. However, several intermediate facts are first required. Also, as the word $abacaba \in \{a, b, c\}^*$ illustrates, full abelian square-free ternary words can be longer than six letters. In fact, the maximum length of a full abelian square-free word is seven [17]. While the partial and full word cases are only slightly different, the simple example shows how the inclusion of holes creates abelian squares.

Proposition 1. *All three-letter abelian square-free words over a three-letter alphabet $A = \{a, b, c\}$ take the form aba or abc, up to a renaming of letters.*

Proposition 1 implies the following two lemmas:

Lemma 1. *Let A be a three-letter alphabet. Up to a renaming of letters, the word aba, where $a, b \in A$, must be a prefix, suffix, or both of all length-six abelian square-free partial words $w \in A_\diamond^*$.*

Proof. Suppose, to the contrary, that an abelian square-free word $w \in \{a, b, c\}_\diamond^*$, where $|w| = 6$, does not contain as a prefix or a suffix *aba*, *aca*, *bab*, *bcb*, *cac*, or *cbc*. This implies that $\psi(w[0, 3)) = \langle 1, 1, 1\rangle$ or that each letter appears in the first three letters of w. But also, we know that $\psi(w[3, 6)) = \langle 1, 1, 1\rangle$ or each letter appears in the last three letters of w. Hence, $\psi(w[0, 3)) = \psi(w[3, 6))$ and w contains an abelian square. ■

Lemma 2. *Over a three-letter alphabet $A = \{a, b, c\}$, there does not exist an abelian square-free partial word of the form $u\diamond v$ such that $|u| = 2, |v| = 3$ and $u, v \in A^*$.*

Proof. This statement follows directly from Proposition 1. Let A be a three-letter alphabet and $w \in A_d^*$ be an abelian square-free word of the form $u\diamond v$, where $|u| = 2, |v| = 3$. If a subword of the form aba is a suffix of w, then $w[3, 6) = \diamond aba$, which is an abelian square. Thus, we reach a contradiction. The same is true if v contains a hole. We can now assume that $\psi(w[3, 6)) = \langle 1, 1, 1\rangle$. We can also assume that u is a full word as the hole would produce an abelian square. Clearly $w(1) \neq w(2)$ and there exists some $x \in A$ such that $x \neq w(1)$ and $x \neq w(2)$. Therefore, we can substitute x into the hole and we have an abelian square. ■

We are now ready to prove Theorem 1 which characterizes abelian square-free partial words over a ternary alphabet.

Proof of Theorem 1. Let the alphabet be $A = \{a, b, c\}$. As a shorthand notation we will define $a_i = w(i) \in A$, for some $i \in \mathbb{N}$. We know that the maximum length must be at least 6 as the word $\diamond bcaba$ is abelian square-free. Now, we must show that if there exists a partial word of length seven, where $w = u\diamond v$ with $w \in A_\diamond^*$, $u, v \in A^*$, then the word w contains an abelian square. We can assume that w contains only one hole because any multiple-hole case can be reduced to a one-hole case. Suppose to the contrary that w is abelian square-free. We proceed by examining the possible positions for the hole.

1. If both $|u|$ and $|v|$ are at least 2, Lemma 2 states that w has an abelian square.

2. If $|u| = 1$, and $|v| = 5$, applying Lemma 1 to the first six characters yields $w = a_0\diamond a_0a_3a_4a_5a_6$ or $w = a_0\diamond a_2a_3a_4a_3a_6$. The first form contains the abelian square $a_0\diamond a_0a_3$. So, we assume that w takes the second form. But, Lemma 1 also applies to the suffix $w[1, |w|)$. Hence, we have either $w = a_0\diamond a_2a_3a_4a_3a_4$ or $\diamond = w(1) = a_3$. Either case is a contradiction.

3. The last case is if $|u| = 0$ and $|v| = 6$. As above, the suffix $w[1..|w|)$ must be abelian square-free, so Lemma 1 gives two possible forms: $\diamond a_1a_2a_1a_4a_5a_6$ and $\diamond a_1a_2a_3a_4a_5a_4$. In the first form, we have the abelian square $\diamond a_1a_2a_1$. To avoid a square in the second form, the letters in positions 3 and 5 must be different, i.e. $a_3 \neq a_5$ and clearly $a_1 \neq a_2$. Thus, the subword $w[0..6)$ is $\diamond a_1a_2a_3a_4a_5$, where a_1, a_2 and a_3, a_4, a_5 are distinct. Hence w violates Lemma 1. Therefore, w contains an abelian square.

Based on these three observations above, we conclude that, no length-seven partial word $w \in A_\diamond^*$ with $H(w) \neq \emptyset$ is abelian square-free and the maximum length of such a word is 6. ∎

Expanding to a four-letter alphabet complicates the situation. Direct proofs would be far more difficult, but by using a computer program, it is easy to check that over a four-letter alphabet all words of the form $u\diamond v$, where u, v are words with $|u| = |v| = 12$, contain an abelian square. A direct implication is that that all words with a factor of this form also contain an abelian square. Hence, we can conclude that:

Proposition 2. *Over a four-letter alphabet, no bi-infinite abelian square-free partial word with one hole exists.*

Furthermore, there are limited possibilities for abelian square-free words with three holes. By computer, we verify that over a four-letter alphabet, for all full words u, v, with $|u|, |v| \leq 12$, the partial word $\diamond u \diamond v \diamond$ contains an abelian square.

Proposition 3. *If a word over a four-letter alphabet contains three holes, it has an abelian square.*

Similarly we see that all partial words $\diamond u \diamond v$ contain an abelian square provided either $|u|, |v| \leq 10$ or $|u| = 11, |v| \geq 5$. This means that, over a four-letter alphabet, an abelian square-free word with two holes has bounded length.

Proposition 4. *No infinite abelian square-free word with two holes can be constructed over a four-letter alphabet.*

5 Infinite-Length Partial Words

Originally conceived by Paul Erdös in [8], infinite abelian square-free full words have been discovered over four-letter [13, 14], five letter [17] and larger [10] alphabets. These infinite words are created using several different abelian square-free iterative morphsims. In this thesis, we are interested in Pleasants' morphism.

In the 1970, Pleasants constructed an infinite abelian square-free word using a five-letter alphabet [17]. By design, his abelian square-free morphism creates imbalances by locally increasing the count of a particular letter. For example, a small section may include a very high number of the letter 'a' whereas another would have a high occurrences of the letter 'b.' These imbalances allowed Pleasants to prove that, if an abelian square existed in an image word, then it could be reduced to specific cases of bounded length. These cases were then checked by computer software. This general method has been repeatedly used by subsequent research on abelian square-free words [13–15]. Here we modify this approach to include partial words.

Theorem 2. *There exists an infinitely long abelian square-free partial word with one hole over a five letter alphabet.*

Proof. Let $A = \{a, b, c, d, e\}$ and let the morphism ϕ be defined by

$$\phi(a) = \texttt{aebedebecedecea}$$

$$\phi(b) = \texttt{bacaeacadaeadab}$$

$$\phi(c) = \texttt{cbdbabdbebabebc}$$

$$\phi(d) = \texttt{dcecbcecacbcacd}$$

$$\phi(e) = \texttt{edadcdadbdcdbde}.$$

Note that $|\phi(f)| = 15$, for all $f \in A$, and within $\phi(f)$ there are two occurrences of each letter from the alphabet except one letter, which has 7 occurrences. Also, each $\phi(f)$ begins and ends with the letter f. We will call the image of each letter a block. The morphism ϕ is an abelian square-free morphism as it is merely a cyclic permutation of the morphism given by Pleasants in [17].

We will show that $\diamond\phi^n(a)$ is an abelian square-free partial word, for all integers $n \geq 0$. The hole can act as any letter in A. As such, showing that $f\phi^n(a)$ is (non-trivially) abelian square-free, for all $f \in A$, proves the claim.

If $f \neq a$, then the word $\phi^n(fa)$ is $\phi^n(f)\phi^n(a)$ and, as $\phi(f)$ begins and ends with f, we have $\phi^n(f)\phi^n(a) = f \ldots f\phi^n(a)$. Since fa is abelian square-free, all subwords of $\phi^n(fa) = \phi^n(f)\phi^n(a)$ must be abelian square-free. We simply take the subword $f\phi^n(a)$.

We cannot use the above logic in the case where $f = a$ because the word aa is an abelian square. Instead, we will show that for all abelian square-free words $w = a_0a_1a_2 \ldots a_{l-1} \in A^*$, the word $a\phi(w)$ has no abelian squares other than the initial aa. To do this, we adapt the proof that ϕ is an abelian square-free morphism shown in [17]. Suppose to the contrary that there is some abelian square uv found in $a\phi(w)$, where $u, v, \in A^*$. We will show that it is only necessary to check factors shorter than a maximum bounded length. First we will define, for some word $x \in A^*$ and letter $f \in A$, the number $|x|_{(\phi(f))}$ as the number of non-overlapping occurrences of the block $\phi(f)$ within x including the cases where a suffix of $\phi(f)$ begins the word x or a prefix of $\phi(f)$ ends the word x. Thus, we can determine upper and lower bounds for $|x|_a$ as

$$|x|_a \leq \frac{2}{15}|x| + 5|x|_{(\phi(b))} + \frac{32}{15}$$

and

$$|x|_a \geq \frac{2}{15}|x| + 5|x|_{\phi(b)} - \frac{32}{15}.$$

These inequalities are modified versions of those found in [17]. The first two terms estimate $|x|_a$ based on the number of occurrences of a in each $\phi(f)$, for all $f \in A$. The third term in both inequalities comes from the maximum error, i.e., when the upper bound is too low or vice versa. Now, substituting the factor u for x, we get

$$|u|_a \leq \frac{2}{15}|u| + 5|u|_{(\phi(b))} + \frac{32}{15}. \tag{1}$$

Doing the same for v gives

$$|v|_a \geq \frac{2}{15}|v| + 5|v|_{\phi(b)} - \frac{32}{15}. \tag{2}$$

We have assumed that the factors u and v are abelian squares. Thus, we know that $|u|_a = |v|_a$. Hence, the inequalities (1) and (2) can be combined and they reduce to

$$|u|_{(\phi(b))} \geq |v|_{\phi(b)} - 64/75.$$

For any word x, the numbers $|x|_{\phi(b)}$ and $|x|_{(\phi(b))}$ are counts of occurrences and thereby, always integers. So we can remove the fraction from the inequality which yields $|u|_{(\phi(b))} \geq |v|_{\phi(b)}$. Similarly, we have $|v|_{(\phi(b))} \geq |u|_{\phi(b)}$. Thus, if we cancel as many pairs of blocks of $\phi(b)$ in u and v (which preserves the abelian square uv), for any remaining blocks of $\phi(b)$ fully contained in one factor, there must be part of a block of $\phi(b)$ in the other. The same is true for all other letters in A. Therefore, the factors u and v can be reduced to factors u' and v', each of which is no longer than twice $\phi(a)$ plus possibly an extra a at the beginning. These finite number of cases can be checked by computer to be abelian square-free. So, for an abelian square-free word w, the word $a\phi(w)$ has no abelian squares other than an initial aa. We conclude that, for all integers $n \geq 0$ and abelian square-free words w, the word $a\phi^n(w)$ is assured to be abelian square-free, except for the initial aa.

Clearly, the word a contains no abelian squares and therefore the word $\diamond\phi^n(a)$ is abelian square-free, for all $n \geq 0$. The hole can act as any letter from A and the resulting word is still (non-trivially) abelian square-free. Therefore, there exists an infinitely long abelian square-free partial word over a five letter alphabet. ■

Theorem 2 gives several immediate corollaries.

Corollary 1. *Over a five letter alphabet, there exists an infinite number of abelian square-free partial words with at least one hole.*

Proof. By Theorem 2, there exists an infinitely long abelian square-free partial word with one hole and this hole has an infinite number of subwords, each of which is abelian square-free. Thus, there are an infinite number of abelian square-free partial words over a five-letter alphabet with at least one hole. ■

Corollary 2. *There exists an abelian square-free partial word with one hole over a six-letter alphabet which extends infinitely in both directions.*

Proof. For a word w, let $\phi'(w) = rev(\phi(w))$ with ϕ being the morphism from the proof of Theorem 2. Hence, $\phi'(w)$ is abelian square-free for all abelian square-free words w and $\phi'^n(a)\diamond$ is abelian square-free for all integers $n \geq 0$. Also, let χ be the morphism over the alphabet $B = \{b, c, d, e, f\}$ that is constructed by replacing each a in the definition of ϕ with a new letter f:

$$\chi(b) = \texttt{bfcfefcfdfefdfb}$$

$$\chi(c) = \texttt{cbdbfbdbebfbebc}$$

$$\chi(d) = \texttt{dcecbcecfcbcfcd}$$

$$\chi(e) = \texttt{edfdcdfdbdcdbde}$$

$$\chi(f) = \texttt{febedebecedecef}$$

By construction, χ is an abelian square-free morphism and $\diamond\chi^n(f)$ is abelian square-free for all integers $n \geq 0$.

We will show that $\phi'^n(a)\diamond\chi^n(f) \in A^*$ is abelian square-free for all integers $n \geq 0$, where $A = \{a, b, c, d, e, f\}$. Suppose to the contrary that there is an abelian square w, which is a subword of $\phi'^n(a)\diamond\chi^n(f)$ for some integer $n \geq 0$. Then, the word w must contain parts of both $\phi'(a)$ and $\chi(f)$. Therefore, at least one half, called u, will be a subword of either $\phi'^n(a)$ or $\chi^n(f)$, meaning it contains either a or f but not both and it does not contain the hole. The other half of w, called v, necessarily contains the complementary letter and the hole. Thus, v contains a letter that u does not and u has no holes. Hence, the two halves of w cannot be permutations of each other.[4] Thus, the word $\phi'^n(a)\diamond\chi^n(f) \in A^*$ is abelian square free for all integers $n \geq 0$. ■

Corollary 3. *There exists an abelian square-free partial word with two holes over an eight-letter alphabet which extends infinitely in both directions.*

Proof. Let the morphisms ϕ and ϕ' be defined as above and let A be the alphabet $\{a, b, c, d, e, f, g, h\}$. The word $\phi'^n(a)\diamond fgh\diamond\phi^n(a) \in A^*$ is abelian square-free for all integers $n \geq 0$. If not, then there exists a word w that is an abelian square and w must contain f, g, or h. By the same logic as in the proof of Corollary 2, w must be centered between the holes. By similar logic as that of the proof of Corollary 2, no such word is an abelian square. Therefore, for all integers $n \geq 0$, the word $\phi'^n(a)\diamond fgh\diamond\phi^n(a) \in A^*$ is abelian square-free. ■

While this line of reasoning could be extended to an arbitrary number of holes, it increases the alphabet size as well. As such, it is counter-productive towards our goal of finding minimum alphabet sizes. Instead, we will now focus on minimizing the alphabet size of the one-hole infinite abelian square-free partial

[4] Note that the hole in v cannot cause u to contain the absent letter.

word. Interestingly, Pleasants noted that his methods could be modified to prove the existenence of an infinite abelian square-free word over four letters, but this would require "considerable computation" and longer morphism blocks of about 30 letters [17]. In 1992, V. Keränen did precisely what Pleasants suggested and proved that as few as four letters are sufficient provided there are no holes [13]. Keränen's work did require greater computation and a morphism with much longer image words—85 letters in the image for every letter in the preimage [13]. Here we use a similar morphism to produce a one-sided infinite abelian square-free word with one hole over four letters.

Theorem 3. *There exists an infinitely long abelian square-free word with one hole over a four-letter alphabet.*

Proof. We will use an abelian square-free morphism ϕ over the four-letter alphabet $A = \{a, b, c, d\}$ provided by Keränen [15] that is defined by

$$\begin{aligned}
\phi(a) =& \texttt{abcacdcbcdcadbdcadabacadcdbcbabcbdbadbdcbabcbdcdacd}-\\
& \texttt{cbcacbcdbcbabdbabcabadcbcdcbadbabcbabdbcdbdadbdcbca}\\
\phi(b) =& \texttt{bcdbdadcdadbacadbabcbdbadacdcbcdcacbacadcbcdcadabda}-\\
& \texttt{dcdbdcdacdcbcacbcdbcbadcdadcbacbcdcbcacdacabacadcdb}\\
\phi(c) =& \texttt{cdacabadabacbdbacbcdcacbabdadcdadbdcbdbadcdadbabcab}-\\
& \texttt{adacadabdadcdbdcdacdcbadabadcbdcdadcdbdabdbcbdbadac}\\
\phi(d) =& \texttt{dabdbcbabcbdcacbdcdadbdcbcabadabacadcacbadabacbcdbc}-\\
& \texttt{babdbabcabadacadabdadcbabcbadcadabadacabcacdcacbabd}.
\end{aligned}$$

Here "$-$" denotes concatenation. The length of each block is 102 and the parikh vector of each is a permutation of $\psi(\phi(a)) = \langle 21, 31, 27, 23\rangle$. We will show that the word $\diamond\phi^n(a)$ is abelian square-free for all integers $n \geq 0$. The general method of proof is similar to the five-letter case. We will prove that only finitely-many bounded cases need to be checked and then check these cases. Since ϕ is abelian square-free, it is sufficient to check the cases where a subword uv begins with the hole.

We argue that, for all letters $f \in A$, we can remove blocks of $\phi(f)$ present in both u and v. Furthermore, after we do so, we will have one of two cases. Either u can contain a full block or v can contain a full block but both cases cannot occur simultaneously, and neither u nor v contains more than one block. Suppose to the contrary we cannot cancel blocks from u and v in this way and some subword of $w = \diamond\phi^n(a)$ is an abelian square uv such that, after the cancellation, we have $|u|_{\phi(f)} > 0$ but $|v|_{\phi(f)} = 0$. Note that the opposite case where $|v|_{\phi(f)} > 0$ but $|u|_{\phi(f)} = 0$ cannot occur as it implies that $|u| < |v|$. We can write $uv = \diamond\phi(w_0)\phi(e)\phi(w_1)x$, where $e \in A$, $w_0, w_1, x \in A^*$, $\diamond\phi(w_0)$ is a prefix of u, u is a prefix of $\diamond\phi(w_0e)$, and $\diamond\phi(w_0ew_1)$ is a prefix of uv and $|x| < 102$. We will denote u' as u with a letter substituted for the hole such that $\psi(u') = \psi(v)$. Knowing $\psi(u') = \psi(v)$, we can build a system of equations for all letters in A. For example, the system of equations for the letter a consists of the following five equations:

$$u'|_{\phi(a)} + |u'|_{\phi(b)} + |u'|_{\phi(c)} + |u'|_{\phi(d)} = |v|_{\phi(b)} + |v|_{\phi(c)} + |v|_{\phi(d)} + \Lambda \tag{3}$$

$$21|u'|_{\phi(a)} + 23|u'|_{\phi(b)} + 27|u'|_{\phi(c)} + 31|u'|_{\phi(d)} = 23|v|_{\phi(b)} + 27|v|_{\phi(b)} + 31|v|_{\phi(b)} + \lambda_a \tag{4}$$

$$31|u'|_{\phi(a)} + 21|u'|_{\phi(b)} + 23|u'|_{\phi(c)} + 27|u'|_{\phi(d)} = 21|v|_{\phi(b)} + 23|v|_{\phi(b)} + 27|v|_{\phi(b)} + \lambda_b \tag{5}$$

$$27|u'|_{\phi(a)} + 31|u'|_{\phi(b)} + 21|u'|_{\phi(c)} + 23|u'|_{\phi(d)} = 31|v|_{\phi(b)} + 21|v|_{\phi(b)} + 23|v|_{\phi(b)} + \lambda_c \tag{6}$$

$$23|u'|_{\phi(a)} + 27|u'|_{\phi(b)} + 31|u'|_{\phi(c)} + 21|u'|_{\phi(d)} = 27|v|_{\phi(b)} + 31|v|_{\phi(b)} + 21|v|_{\phi(b)} + \lambda_d \tag{7}$$

Equation (4) is a result of the fact that the number of occurrences of the letter a in the word u' must be equal to the number of occurrences of the letter a in the word v'. Likewise, since the number of occurrences of the letter b must be equal, we have Equation (5). The same logic applied to the letter c provides Equation (6). Equation (7) corresponds to the letter d. The parameter Λ is an error term taking values of $-1, 0, 1$, but can only be non-zero for one system of equations. Each λ_i represents error caused by the opening $\diamond$, $\phi(e)$ or x. Using software-assisted Gaussian elimination, it is easy to see that this system is inconsistent provided that some $\lambda_i \neq 0$. However, the hole in the beginning ensures at least one $\lambda_i \neq 0$. Thus, except for possibly one $\phi(f)$ with $f \in A$, we can cancel $\phi(w_0)$ and $\phi(w_1)$ to yield $\diamond\phi(e)x$ or $\diamond\phi(e)\phi(f)x$. As we remove the same letters from each side, any abelian square is preserved. Also, the preimage of $uv[1..|uv|)$ is abelian square-free, so $\phi(e)x$ or $\phi(e)\phi(f)x$ must be the image of some abelian square-free word. This means that we only need to check the image of abelian square-free words having outputs that are less than three times the block length. For all such words, it is easy to verify that we always have an abelian square-free result. Thus, $\diamond\phi^n(a)$ is abelian square-free for all integers $n \geq 0$. ■

With this new result, we can reduce the alphabet requirements of Corollaries 1, 2, and 3 by one letter and obtain the following stronger corollaries.

Corollary 4. *Over a four letter alphabet, there exists an infinite number of abelian square-free partial words with at least one hole.*

Corollary 5. *There exists an abelian square-free partial word with one hole over a five-letter alphabet which extends infinitely in both directions.*

Corollary 6. *There exists an abelian square-free partial word with two holes over a seven-letter alphabet which extends infinitely in both directions.*

In addition, since Proposition 1 states that over a three-letter alphabet, no infinite abelian square-free partial word exists, we know that four letters is the minimum alphabet size.

Corollary 7. *The minimum alphabet size of an abelian square-free partial word with one hole is four.*

While Proposition 4 states that we cannot build an infinite abelian square-free word over four letters that has two holes, we can increase the alphabet size to accommodate more holes. In fact, provided we have a large enough alphabet, we can introduce an infinite number of holes and still have an abelian square-free word.

Theorem 4. *There exists an infinite partial word with infinitely many holes over a seven-letter alphabet that is abelian square-free.*

Proof. As shown by Keränen, there exist infinite abelian square-free full words over four-letter alphabets [13]. Let w be one such word of infinite length over an four-letter alphabet $A = \{a, b, c, d\}$. We will define a sequence k_j in the following way. There exists some $x, z, y \in A$ so that for infinitely many i we have $w(i-1) = x$, $w(i) = y$, and $w(i+1) = z$. Let k_0 be the smallest integer so that $w(k_0-1) = x$, $w(k_0) = y$ and $w(k_0 + 1) = z$. We define k_j recursively, where k_j is the smallest integer such that $k_j > 5k_{j-1}$, $w(k_j - 1) = x$, $w(k_j) = y$, and $w(k_j + 1) = z$. Moreover, define $A' = A \cup \{e, f, g\}$, where $e, f, g \notin A$. Note that $|A'| = 7$.

We will now define an infinite partial word $w' \in A'^*$ which is abelian square-free. For all $j \in \mathbb{N}$, if $j \equiv 0 \bmod 7$, then let $w'(k_j-1) = e$, $w'(k_j) = \diamond$, $w'(k_j+1) = f$. Similarly, for all $j \in \mathbb{N}$, if $j \not\equiv 0 \bmod 7$, then let $w'(k_j) = g$. For all other indices, let $w'(i) = w(i)$. It should be noted that w' has infinitely many holes.

In order to prove that w' contains no abelian squares, we will assume that it does contain one and obtain a contradiction. Let u and v be two words defined by $u = w'(i) \dots w'(i + l)$ and $v = w'(i + l + 1) \dots w'(i + 2l + 1)$ (so the words u and v both have length $l > 1$) and let uv be an abelian square.

Let J_1 and J_2 be sets defined by $J_1 = \{j : i \le k_j \le i + l\}$ and $J_2 = \{j : i+l+1 \le k_j \le i+2l+1\}$. Thus, $|J_1|$ is the number of the k_j's in u and $|J_2|$ is the number of k_j's in v. We assert that $|J_1| < 3$ and $|J_2| < 2$. To see this, first assume that $|J_2| > 1$. Then there exists a $j \in J_2$ so that $j+1 \in J_2$. However, this implies a contradiction because $l = i+2l+1-(i+l+1) > k_{j+1}-k_j > k_j > i+l \ge i+l-i = l$. Now assume that $|J_1| > 2$. Then there are at least two occurrences of the letter g in u, and for each occurrence of g there must also be a g or a hole in v. However,

g's and holes only occur when $i = k_j$ for some j, so this implies $|J_2| \geq 2$, which violates the claim that $|J_2| < 2$. The facts that $|J_1| < 3$ and $|J_2| < 2$ imply that $|J_1 \cup J_2| < 4$.

Next, we want to show that no holes occur in the abelian square uv. We will only prove that no holes exist in u, as the proof for v is similar. The occurrence of a hole implies that there exists a j so that $i \leq k_j \leq i + l$. Therefore, either e or f occurs in u, since w' must contain either $w'(k_j - 1)$ or $w'(k_j + 1)$. Assume that e occurs; the case where f occurs is similar. Then v must contain either e or a hole, but that implies that $i + l \leq k_{j+7} - 1 \leq i + l + 1$, since $w'(k_{j+7} - 1)$ is the next occurrence in the w' of either e or a hole. Thus $j, j + 1, .., j + 6 \in J_1 \cup J_2$, which implies that $|J_1 \cup J_2| \geq 6 > 3$, a contradiction, so no such hole can exist.

Therefore, we can assume all characters in uv are letters in A'. Define w'' as follows. If $w'(i) = e$, then $w''(i) = z$. If $w'(i) = f$, then $w''(i) = y$. If $w'(i) = g$, then $w''(i) = x$, otherwise $w''(i) = w'(i)$. In other words, we have mapped e, f, and g back to their original values. Let $u' = w''(i) \cdots w''(i+l), v' = w''(i + l + 1) \cdots w''(i + 2l + 1)$. Obviously $u'v'$ is still an abelian square, but by construction $u'v' = w(i) \cdots w(i + l)w(i + l + 1) \cdots w(i + 2l + 1)$, which implies that w contains an abelian square. This is a contradiction. Therefore, w' avoids abelian squares. ∎

Using a similar construction, we can reduce the required alphabet size by one. The overall argument is the same as in the proof of Theorem 4 but we need to address several more details.

Theorem 5. *There exists an infinite partial word with infinitely many holes over a six-letter alphabet that is abelian square-free.*

The following remark will aid in proving Theorem 5:

Remark 1. *Let A be a k-letter alphabet, $uv \in A^*$, $a_i \in A$ and $a_k \notin A$. Replace an equal number of a_i's in both u and v with the same number of a_k's, yielding a new word $u'v'$. If uv was originally an abelian square, $u'v'$ remains an abelian square. Similarly, if uv was originally abelian square-free, $u'v'$ remains abelian square-free.*

Proof of Theorem 5. As before, there exists an abelian square-free word w of infinite length over an four-letter alphabet $A = \{a, b, c, d\}$ [13]. Define a sequence k_j as follows. Choose some $x, y \in A$ so that for infinitely many j we have $w(j-1) = x$, $w(j) = y$, and $w(j+1) = x$. Let k_0 be the smallest integer so that $w(k_0 - 1) = x$, $w(k_0) = y$, and $w(k_0 + 1) = x$. We define k_j recursively, where k_j is the smallest integer such that $k_j > 5k_{j-1}$, $w(k_j - 1) = x$, $w(k_j) = y$, and $w(k_j + 1) = x$. Also, let the alphabet $A' = A \cup \{e, f\}$, where $e, f \notin A$. This means that $|A'| = 6$.

As before, we will substitute e and f for x and y to form an infinite word $w' \in A'^*$ that is abelian square-free. For all $j \in \mathbb{N}$, if $j \equiv 0 \bmod 7$, then let $w'(k_j - 1) = e$, $w'(k_j) = \diamond$, and $w'(k_j + 1) = f$. For all $j \in \mathbb{N}$, if $j \equiv 1 \bmod 7$, let $w'(k_j) = e$. For $j \not\equiv 0 \bmod 7$ and $j \not\equiv 1 \bmod 7$, let $w'(k_j) = f$. Everywhere else let $w'(i) = w(i)$. In order to prove that w' contains no abelian squares, we will assume that it does contain one and obtain a contradiction. Let $u = w'(i) \cdots w'(i+l)$, $v = w'(i + l + 1) \cdots w'(i + 2l + 1)$ and uv be an occurrence of an abelian square.

Let the sets J_1 and J_2 be defined as in the proof of Theorem 4. That is to say, let $J_1 = \{j : i \leq k_j \leq i + l\}$ and $J_2 = \{j : i + l + 1 \leq k_j \leq i + 2l + 1\}$. Then, we claim that $|J_1| < 3$ and $|J_2| < 2$. The reasoning for $|J_2| < 2$ is exactly the same as above. However, the rationale for the claim $|J_1| < 3$ is slightly different. Assume to the contrary that $|J_1| > 2$. Then there are at least two occurrences of the letter f in u, and for each occurrence of f, there must also be an f or a hole in v. Holes and the letter f only occur at indices that are equal to k_j, for some j. Therefore, the assumption that $|J_1| > 2$ implies $|J_2| \geq 2$. We can again conclude that $|J_1 \cup J_2| < 4$.

Next, we will show that abelian square uv contains no holes. First, we will prove none occur in u. There are three subcases:

1. If u begins with a hole, then the letter e corresponding to $j \equiv 1 \bmod 7$ either occurs in u or it does not. If e is in u, then e is also contained in v. However, this implies that $|J_2| > 1$, a contradiction. If e does not occur in u, the word u contains only $\diamond f$. The letter e corresponding to $j \equiv 1 \bmod 7$ is contained in v, but f must also occur in v. This implies that $|J_2| > 1$, a contradiction.

2. If the hole is in the middle of u, then both e and f are in u, and must also be contained in v. However, this can only occur if $|J_2| > 2$, a contradiction.

3. If u ends with a hole, then there is at least one e in u, which implies that there is also an e in v. This, together with the recursive definition of k_j, means that $|v| \geq k_{j+1} - k_j > k_j - k_{j-2}$. Since $|u| = |v| > k_j - k_{j-2}$, u also contains two f's. Then v must also contain two f's, but this cannot be resolved with $|J_2| < 2$.

Likewise, the case where a hole is in v contains three subcases.

1. If v begins with a hole at $w(k_{j+1}) = w(i+l+1)$, then there is necessarily an f in v. This implies that there is exactly one f in u. Let this f in u occupy $w(k_j)$. We claim that $|\psi(u_0 f u_1 e) - \psi(f v_0)| > 1$, where $u_0 = w(i) \cdots w(k_j - 1)$, $u_1 = w(k_j+1) \cdots w(i+l-1)$, and $v_0 = w(i+l+3) \cdots w(i+2l+1)$. We

may prove this by carefully examining the construction of uv. In the original $w(i)\cdots w(i+2l+1)$, there must be at least two letters $a_0, a_1 \in A$ such that $|w(i)\cdots w(i+l)|_{a_0} \neq |w(i+l+1)\cdots w(i+2l+1)|_{a_0}$, and similarly for the number of a_1. Replacing the letter in $w(i+l)$ with e and in $w(i+l+2)$ with f results in four letters with an unequal number of occurrences, while there is at most one $a_2 \in A'$ such that $|u_0 w(k_j) u_1 e|_{a_2} = |w(i+l+1) f v_0|_{a_2}$. Substituting f into $w(k_j)$ and $\diamond$ into $w(k_{j+1}) = w(i+l+1)$ yields uv. There are three cases to consider. If the hole takes the value of any letter in A, then there remains exactly one e in uv, making it abelian square-free. If the hole takes the value of f, then $|uv|_f$ is odd, so uv is not an abelian square. If the hole takes the value of e, we have replaced x and y in $w(i)\cdots w(i+l)$ with f and e, respectively, while in $w(i+l+1)\cdots w(i+2l+1)$ we have replaced x and y with f and e, respectively. Extending Remark 1, since $w(i)\cdots w(i+2l+1)$ is abelian square-free, uv must also be abelian square-free.

2. If the hole in v is in the middle of v, then there is an e in v and a corresponding e in u. This implies that $|J_1| \geq 6 > 2$, a contradiction.

3. If v ends with a hole, then there is an e in v and a corresponding e in u. This implies that $|J_1| \geq 6 > 2$, a contradiction.

Therefore, no holes are present in uv and we can assume that all characters in uv are letters in A'. Due to the fact that $|J_1 \cup J_2| < 4$, the abelian square uv occurs only if both u and v contain exactly one f and no e's. Define u' and v' as follows. If $w'(i) = f$, then $w''(i) = y$. Otherwise, $w''(i) = w'(i)$. The word $u'v'$ remains an abelian square, but we have a subword of our original abelian square-free word w. Thus, we may conclude that the infinite word w' is an abelian square-free partial word containing infinitely many holes. ■

By further modification to the construction of the word w', the alphabet size can be further reduced to five. For a proof of the next theorem, see [3].

Theorem 6. *There exists an infinite abelian square-free partial word that contains an infinite number of holes over a five-letter alphabet.*

6 Special Case Alphabets

Throughout the thesis we have used a general, abstract approach and this approach is usually beneficial. However, using specific alphabets can allow for interesting, alternative interpretations. In this section we will examine words, squares and abelian squares using two such alphabets. We will see that these alphabets can provide geometric interpretations and convenient definitions.

We begin by using alphabets made of orthogonal unit vectors. We define an alphabet $E_n = \{e_1, e_2, \ldots e_n\}$ to be the standard basis of $\mathbb{R}^n$. Of course this means that $||E_n|| = n$. Furthermore, a word $w \in E_n^*$ is a path of vectors (end-to-end) in n-space which, for convenience, begins at the origin. For example, we can represent $e_1e_2e_1e_2$ with the sequence of unit vectors in Figure 1. The characterization of squares remains largely the same as in general. However, abelian squares now have geometric meaning. Suppose we have an abelian square uv. Then, we know that the number of each e_i in u is the same as in v. Hence, we know that

$$\sum_{j=0}^{|u|-1} u(j) = \sum_{j=0}^{|v|-1} v(j).$$

In other words, the resultant sum of the components of u is the same as the

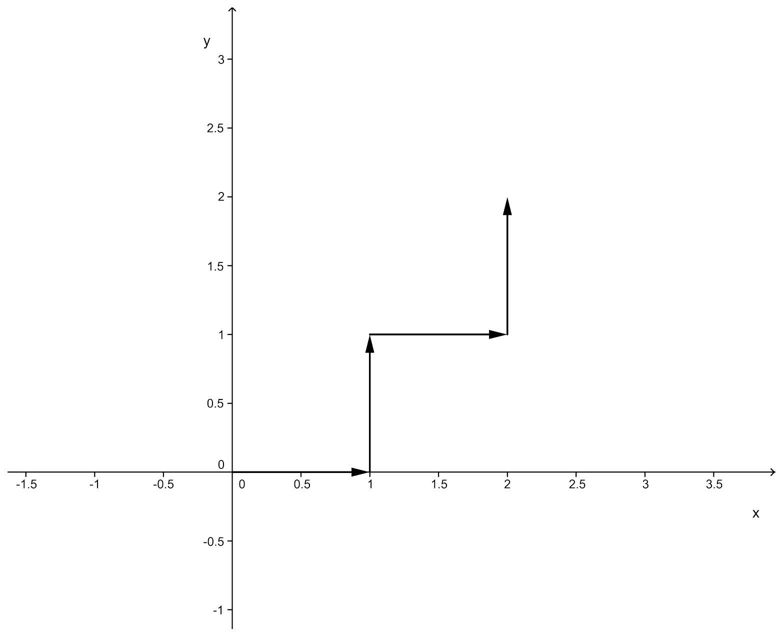

Figure 1: Vector representation of $e_1e_2e_1e_2$.

resultant sum of the components in v. This means that uv is an abelian square if and only if the origin, midpoint, and endpoint are collinear. An example of this is shown in Figure 2. So if we define a sequence w' as the sequence of points formed by the endpoints of the vectors in w, the word w is abelian square-free if the points w'_i, w'_{i+n}, w'_{i+2n} are not collinear, for all $i, n \in \mathbb{N}$.

Another interesting set of alphabets are P_n, where P_n denotes the first n primes. These alphabets are central to Algorithm 2. Since prime factorizations are unique, the product of the letters of a word form a unique integer up to a rearrangement of the letters. Hence, these alphabets are convenient when working with abelian squares. An abelian square is a word uv, where $\prod_{i=0}^{|u|-1} u(i) = \prod_{i=0}^{|v|-1} v(i)$. This definition of abelian square provides a logically simple way to check if a word is an abelian square, which is shown in Algorithm 2. In addition, this type of checking is aided by the fact that all modern computer processors are optimized

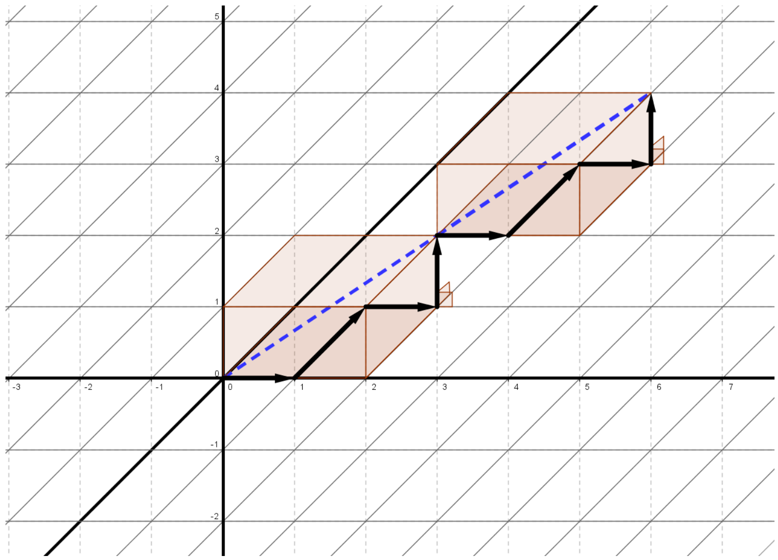

Figure 2: The square and abelian square $e_1e_2e_1e_3e_1e_2e_1e_3$. Positive z-axis is into the page.

for highly efficient integer multiplication.

We will now provide a way in which we could check if certain partial words are abelian squares. Suppose we have a partial abelian square $uv \in P_n^*$, for some n, where v contains no holes. Let us denote the number of holes in u with n. Since, by definition, there exists some full word $u'v$ such that $uv \subset u'v$ and $u'v$ is an abelian square, we know that every letter in u also exists in v. Recall that in this case all the letters are prime numbers. Hence, we know that $\prod_{k \in D(u')} u'(k)$ divides $\prod_{k=0}^{|v|-1} v(k)$. Let q be the quotient of $\prod_{k=0}^{|v|-1} v(k) / \prod_{k \in D(u)} u'(k)$. If the prime factorization of q contains n factors, we know that there exist n letters that we can use to form u' such that $u'v$ is an abelian square. Hence, we conclude that uv is an abelian square.

Unfortunately, this method has several flaws. First, the prime factorization of q is a time-consuming operation. This operation alone hinders the efficiency of any implementations. Second, the method does not easily generalize to all partial words. The problem is that if v contains holes, then $\prod_{k\in D(u)} u(k)$ does not necessarily divide $\prod_{k\in D(v)} v(k)$. Nonetheless, the use of multiplication instead of counts makes these alphabets noteworthy.

7 Conclusion

We see that, despite the extra freedom awarded by holes, abelian squares are avoidable in partial words. Moreover, the required alphabet sizes are not entirely different than in the case of full words. Table 2 shows that, in the single-hole, single-direction case, the minimum alphabet size is exactly the same as in the full word case, namely 4. Furthermore, by extending the alphabet to five letters we construct a bi-infinite abelian square-free word. In fact, we know that as few as five letters can be used to construct an infinite abelian square-free word with an infinite number of holes.

	No Holes	1 Hole	2 Holes	Infinitely-Many Holes
Single-Direction	4	4	5	5
Bi-Infinite	4	5	5	5

Table 2: Required alphabet sizes for infinite and bi-infinite words.

In reference to future research, there are at least three immediate, intriguing areas. First is the issue of distance between holes. The constructions in the proofs of Theorem 4 and Theorem 5 produce words with holes that are arbitrarily far apart. In fact, the increasing distance between holes is essential to the intermediate claims used in the proof. Hence, we are interested in a construction with holes being a bounded distance apart. What size alphabet is required to construct such a word? And, how close can the holes be? Second, is it possible to construct a word that is abelian square-free after arbitrary insertion of holes? This type of results have been found for regular repetitions in partial words [5, 16]. However, no such research has been published on abelian repetitions and it is unknown if appropriate words exist. Lastly, we suggest the investigation of abelian overlap-free words. An overlap is a repetition that shares center characters (a word of the form xyx) and overlaps have been studied both in full and partial word contexts [1, 4, 7, 12]. It would be interesting to extend this idea to abelian overlaps or abelian squares which share center characters. However, we have no conjecture as to the existence of abelian overlap-free words.

8 References

[1] J.-P. Allouche and J. Shallit. Sums of digits, overlaps, and palindromes. *Discrete Mathematics and Theoretical Computer Science*, 4:1–10, 2000.

[2] J. Berstel and L. Boasson. Partial words and a theorem of Fine and Wilf. *Theoretical Computer Science*, 218:135–141, 1999.

[3] F. Blanchet-Sadri, J. Kim, R. Mercaş, W. Severa, and S. Simmons. Abelian square-free partial words. *Language and Automata Theory and Applications*, To Appear.

[4] F. Blanchet-Sadri, R. Mercaş, A. Rashin, and E. Willett. An answer to a conjecture on overlaps in partial words using periodicity algorithms. In A. H. Dediu, A. M. Ionescu, and C. Martín-Vide, editors, *Language and Automata Theory and Applications 2009*, volume 5457 of *Lecture Notes in Computer Science*, pages 188–199, Berlin, Germany, 2009. Springer-Verlag.

[5] F. Blanchet-Sadri, R. Mercaş, and G. Scott. A generalization of Thue freeness for partial words. *Theoretical Computer Science*, 410(8-10):793–800, 2009.

[6] A. Carpi. On abelian squares and substitutions. *Theoretical Computer Science*, 218(1):61–81, 1999.

[7] J. Cassaigne. Counting overlap-free binary words. In *STACS '93: Proceedings of the 10th Annual Symposium on Theoretical Aspects of Computer Science*, pages 216–225, London, UK, 1993. Springer-Verlag.

[8] P. Erdös. Some unsolved problems. *Magyar Tudományos Akadémia Matematikai Kutató Intézete*, 6:221–254, 1961.

[9] A. Evdokimov. Strongly asymmetric sequences generated by a finite number of symbols. *Doklady Akademii Nauk SSSR*, 179:1268–1271, 1968. In Russian. English translation in *Soviet Math. Dokl.* 9:536–539, 1968.

[10] A. Evdokimov. The exitence of a basis that generates 7-valued iteration-free sequences. *Diskret. Analiz*, 18:25–30, 1971.

[11] V. Halava, T. Harju, and T. Kärki. Square-free partial words. *Information Processing Letters*, 108(5):290–292, 2008.

[12] V. Halava, T. Harju, T. Kärki, and P. Séébold. Overlap-freeness in infinite partial words. *Theoretical Computer Science*, 410(8-10):943–948, 2009.

[13] V. Keränen. Abelian squares are avoidable on 4 letters. In W. Kuich, editor, *ICALP '92: Proceedings of the 19th International Colloquium on Automata, Languages and Programming*, pages 41–52, London, UK, 1992. Springer-Verlag.

[14] V. Keränen. New abelian square-free DT0L-languages over 4 letters. In *Proceedings of the Fifth International Arctic Seminar*, Murmansk, Russia, 2002. Murmansk State Pedagogical Institute.

[15] V. Keränen. The image words of g85, g98, and 200 more a2f endomorphisms of length 102—115, 2007. `http://south.rotol.ramk.fi/keranen/research/a2fsubstitutions/web/a2fEndomorphismsOver4Letters_Lengths102-115_200inNumber+g85&g98.html`.

[16] F. Manea and R. Mercaş. Freeness of partial words. *Theoretical Computer Science*, 389(1-2):265–277, 2007.

[17] P. Pleasants. Non repetitive sequences. *Proceedings of the Cambridge Philosophical Society*, 68:267–274, 1970.

[18] A. Thue. Über unendliche Zeichenreihen. *Norske Vid. Selsk. Skr. I, Mat. Nat. Kl. Christiana*, 7:1–22, 1906. Reprinted in *Selected Mathematical Papers of Axel Thue*, T. Nagell, editor, Universitetsforlaget, Oslo, Norway 139–158, 1977.

[19] A. Thue. Über die gegenseitige Lage gleicher Teile gewisser Zeichenreihen. *Norske Vid. Selsk. Skr. I, Mat. Nat. Kl. Christiana*, 1:1–67, 1912. Reprinted in *Selected Mathematical Papers of Axel Thue*, T. Nagell, editor, Universitetsforlaget, Oslo, Norway 413–478, 1977.

A Symbol Glossary

A^*	All full words made of letters in A	3
$A_\diamond^*$	All partial word made of letters in A	5
$\|\|A\|\|$	Size of alphabet A	3
$D(x)$	Domain set of x	6
ε	Empty word	3
E_n	Standard Basis of $\mathbb{R}^n$	31
$H(x)$	Hole set of x	6
$\psi(w)$	Parikh vector of w	4
ϕ^n	n applications of the ϕ	5
$\mathbb{N}$	Natural numbers; $0, 1, 2, \ldots$	-
$O(N)$	Big O	9
P_n	The first n primes	32
$\mathrm{rev}(x)$	Reversal of x	3
$\|w\|$	Length of w	3
$\|w\|_a$	Number of occurrence of a in w	4
$\|w\|_{(x)}$	Number of partial occurrences of x in w	18
$w(i)$	i-th letter of w	5
$w[i, j)$	$w(i)w(i+1)\ldots w(j-1)$	5
$\diamond$	Hole	5
$\blacksquare$	Halmos, End-of-Proof	-
$\subset$	Containment	6
$\uparrow$	Compatibility	6
$-$	Concatenation	22