The Penney’s Game with Group Action

Sean Li and Tanya Khovanova

Abstract. Consider equipping an alphabet $A$ with a group action which partitions the set of words into equivalence classes which we call patterns. We answer standard questions for Penney’s game on patterns and show non-transitivity for the game on patterns as the length of the pattern tends to infinity. We also analyze bounds on the pattern-based Conway leading number and expected wait time, and further explore the game under the cyclic and symmetric group actions.

1. Introduction

The Penney ante game, or Penney’s game, is a two-player game with a fair coin. The two players, Alice and Bob, each pick a word consisting of $H$s and $T$s, where $H$ is for heads and $T$ is for tails. Both words are of the same length. The coin is then flipped repeatedly, and the person who chose the word that appears first is declared the winner. Fixing the words that Alice and Bob choose, what are the odds that Alice wins?

Many other interesting questions are related to this game. What is the expected amount of flips, also known as the expected wait time, until Alice’s word appears? How many words of a given length avoid Alice’s word? Bob’s word? Both? What is the best choice for Bob if he knows Alice’s word?

The game first appeared in 1969 as a problem submitted to the Journal of Recreational Mathematics by Walter Penney [9]. It was later popularized by Martin Gardner [5,6], who introduced the Conway leading numbers that allow us to easily calculate the odds for the game. Not only that, but many things can be expressed through Conway leading numbers. For example, the expected wait time for a particular word is twice the Conway leading number. The same method is described in the Winning Ways for Your Mathematical Plays [2]. Collings in 1982 [3] generalized Conway leading numbers to the case when we do not start from scratch, but from a given string.
This game can be generalized to alphabets consisting of more than two letters. The generalization of these results to larger alphabets was explored by Guibas & Odlyzko in 1981 [7].

Penney’s game is a famous classic example of a non-transitive game. For example, the word $HHT$ has better odds than $HTT$, the word $THH$ has better odds than $HHT$, while the word $THH$ does not have better odds than $HTT$. In addition, Bob can always choose a word with better odds that Alice, no matter what word she chooses. The best choice for Bob for any alphabet was discussed in [7] and later finalized by Felix in 2006 [4].

We cover all the details related to the original game in Sect. 2.

Although not within the scope of this paper, there are many preexisting variations of Penney’s game. A popular variation uses various other objects in place of a coin, such as a finite deck of cards in Humble & Nishiyama [8] or a roulette wheel in Vallin [10]. Many more variations of this game were studied by Agarwal et al. [1].

In this paper, we introduce a variation of this game where Alice chooses a pattern, representing a collection of words, rather than a single word. For example, she can bet that three identical flips will appear first. In response, Bob can bet that three flips that form an alternating sequence of heads and tails will appear first instead.

Formally, we define a group action on the alphabet that swaps $H$ and $T$ and lets two words be equivalent if they are within the same orbit of the group action. In our particular example above, Alice bets on the two words $HHH$ and $TTT$, while Bob bets on the two words $HTH$ and $THT$.

We generalize this example to any group action. The goal of the paper is to generalize the classic Penney’s game within this new framework. To do this, we calculate the odds of winning depending on the chosen patterns, and also expand upon many known results for the original game. We present the following road-map of the paper.

We start Sect. 3 with a motivating example, then define the group action and the notion of a pattern. We also list interesting potential groups for study.

In Sect. 4, we define the correlation, correlation polynomial, period, and Conway leading number for two patterns. We make a few expository claims about their properties and discuss the similarities and differences between the theory of patterns and the theory of words.

In Sect. 5, we prove the theorem that describes the generating functions that avoid a given set of patterns and also the functions when a particular pattern appears for the first time while avoiding other patterns.

In Sect. 6, we calculate the expected wait time of a given pattern. We discuss patterns of fixed length with maximal and minimal wait times.

In Sect. 7, we calculate the odds of Alice winning the game, given the choice of patterns for Alice and Bob, thus generalizing Conway’s formula. We use this result to also compute the expected length of the game on two patterns.

In Sect. 8.1, we focus on the cyclic group and the group action that cycles all the letters of the alphabet. We show via bijection that the game on patterns
under a cyclic group is equivalent to the game on words that are one letter shorter.

In Sect. 8.2, we focus on the symmetric group: the group that shuffles all the letters of the alphabet. We provide many examples and calculate the exact lower bound for Conway leading numbers. We find that for this group, it is not always possible for Bob to find a word with better odds than Alice’s word. More precisely, there exist words that have better odds than other words of the same length.

2. The Original Problem

In this section, we cover what is known about Penney’s game. For more detail one can check [2–7, 9].

The Penney’s game is a two-player game with a fair coin. The two players, Alice and Bob, each pick a word consisting of Hs and Ts. The coin is then flipped repeatedly, and the person who chose the word that appears first is declared the winner. For instance, say that Alice and Bob pick HHH and HTH, respectively. Then in the sequence of flips HHTTHTTTHTH, Bob wins on the eleventh flip. A natural question then arises. Given Alice’s and Bob’s words, what are the odds that either player wins?

Penney’s game is a classic non-transitive game. Here is an example of a loop of preferences:

• if Alice picks HTT, Bob picks HHT to have a higher chance of winning;
• if Alice picks HHT, Bob picks THH to have a higher chance of winning;
• if Alice picks THH, Bob picks TTH to have a higher chance of winning;

and

• if Alice picks TTH, Bob picks HTT to have a higher chance of winning.

Moreover, if Bob can choose his word after he knows Alice’s word, he always has a winning strategy regardless of the word Alice chooses. For words of length 3, Bob can always find a word that makes his odds at least 2 to 1, see for example [2, 4]. The fact that Bob can always choose a word of the same length as Alice’s word with better odds implies non-transitivity.

2.1. Correlation Polynomials and Conway Leading Numbers

In this section, we cover theory needed to determine the explicit probabilities of winning are also known for any pair of words. In the literature, this theory is expressed in terms of correlation polynomials, or equivalently, Conway leading numbers.

We first fix an alphabet $A$ of $q$ letters. In the classical Penney’s game, the alphabet consists of two letters, $H$ and $T$. A word $w(1)w(2)\ldots w(\ell)$ of length $\ell$ is the concatenation of $\ell$ elements of $A$; this may also be viewed as an element of the $\ell$-fold Cartesian product $A^\ell$. We let $w(i, j) = w(i)w(i+1)\ldots w(j)$ denote the consecutive substring of letters between the $i$-th and $j$-th letter. In this paper we are mostly interested in prefixes $w(1, j)$ and suffixes $w(\ell - j + 1, \ell)$ of length $j$. 
Let \( v \) and \( w \) be two words, with \( v \) having length \( \ell \). The correlation vector \( \vec{C}(v, w) \) is a vector \( (C_0, \ldots, C_{\ell-1}) \), where \( C_i \) is equal to 1 if the first \( \ell - i \) letters of \( w \) match the last \( \ell - i \) letters of \( v \), i.e. \( w(1, \ell - i) = v(i + 1, \ell) \), and is equal to 0 otherwise. The correlation polynomial is defined as \( C_{v,w}(z) = C_0 z^0 + \cdots + C_{\ell-1} z^{\ell-1} \). Finally, we define the Conway leading number (CLN), denoted \( vLw \), as the value of \( \vec{C}(v, w) \) viewed as a string written in base \( q \), where \( q \) is the number of letters in the alphabet. In terms of the correlation polynomial, we have
\[
vLw = q^{\ell - 1} C_{v,w} \left( \frac{1}{q} \right).
\]

Example 2.1. For \( (v, w) = (HTHT, HTHTH) \), we may compute the correlation vector \( \vec{C}(v, w) = (1, 0, 1, 0) \), the correlation polynomial \( C_{v,w}(z) = 1 + z^2 \), and the Conway leading number \( vLw = 2^3 + 2^1 = 10 \).

For the special case of when \( v = w \), we call \( \vec{C}(w, w) \) and \( C_{w, w}(z) \) the autocorrelation vector and autocorrelation polynomial, respectively.

Example 2.2. The autocorrelation vector of \( w = HTHTH \) is \( \vec{C}(w, w) = (1, 0, 1, 0, 1) \), and the autocorrelation polynomial is \( C_{w,w}(z) = 1 + z^2 + z^4 \).

The coordinates of the autocorrelation vector depend on each other. If a word has period \( s \), it also has period \( ks \) for all \( k \geq 1 \). If a word has periods \( s \) and \( t \), it also has period \( s + t \). For example, if \( C_1 = 1 \), then all the letters in the word are the same, which implies that \( C_i = 1 \) for all \( i < \ell \). Generally, if a word has periods \( s < t \) it does not necessarily have period \( t - s \).

Example 2.3. The word \( HTHHTH \) has periods 3 and 5, but not period \( 5 - 3 = 2 \).

However, for sufficiently long words the implication is true. The following proposition and its corollary are proven in [7].

Proposition 2.4. If a word \( w \) of length \( \ell \geq s + t \) has periods \( s < t \), then \( w \) also has period \( t - s \).

Corollary 2.5. If a word \( w \) of length \( \ell \) has least period \( s \) and period \( t \) not divisible by \( s \), then \( t \geq \lfloor (\ell + 1)/2 \rfloor + 1 \).

2.2. Generating Functions

In this section, we determine a closed form generating function for Penney’s game in terms of the correlation polynomial. A set of words is reduced if no word \( w \) is a substring of another word \( w' \) in the set. For instance, the set \( \{HTH, TTHTH\} \) is not reduced.

Suppose we have a reduced set of \( k \) words \( S = \{w_1, w_2, \ldots, w_k\} \) with lengths \( \ell_1, \ell_2, \ldots, \ell_k \), composed of letters from an alphabet of size \( q \). Let \( A(n, S) \) denote the number of strings of length \( n \) which avoid all words in \( S \). We define
\[
G(z, S) = \sum_{n=0}^{\infty} A(n, S) z^n
\]
to be the generating function which describes the number of words avoiding all words in $S$. Similarly, let $T_{w_i}(n, S)$ denote the number of strings of length $n$ which avoid all words in $S$, except for a final appearance of $w_i$ at the end of the word. We call such strings first occurrence strings. Then we define the generating function

$$G_{w_i}(z, S) = \sum_{n=0}^{\infty} T_{w_i}(n, S)z^n.$$  

When the set $S$ in question is obvious, we drop it to shorthand, so $G(z) = G(z, S)$ and $G_{w_i}(z) = G_{w_i}(z, S)$.

Remark. The reason we reduce the set $S$ is the following. Suppose that $w$ is a substring of $w'$. Then whenever $w'$ appears at the end of a string, then $w$ will appear too, i.e. $G_{w'}(z, S) = 0$ or a degeneracy.

The following theorem on the avoiding set $S$ is proven in [7].

**Theorem 2.6.** The generating functions $G(z)$, $G_{w_1}(z)$, $G_{w_2}(z)$, ..., $G_{w_k}(z)$ satisfy the following system of linear equations:

\[
\begin{align*}
(1 - qz)G(z) + G_{w_1}(z) + G_{w_2}(z) + \cdots + G_{w_k}(z) &= 1 \\
G(z) - z^{-\ell_1}C_{w_1, w_1}(z)G_{w_1}(z) - \cdots - z^{-\ell_k}C_{w_k, w_1}(z)G_{w_k}(z) &= 0 \\
G(z) - z^{-\ell_1}C_{w_1, w_2}(z)G_{w_1}(z) - \cdots - z^{-\ell_k}C_{w_k, w_2}(z)G_{w_k}(z) &= 0 \\
& \quad \vdots \\
G(z) - z^{-\ell_1}C_{w_1, w_k}(z)G_{w_1}(z) - \cdots - z^{-\ell_k}C_{w_k, w_k}(z)G_{w_k}(z) &= 0 
\end{align*}
\]

The following corollary is also proven in [7].

**Corollary 2.7.** If $k = 1$, i.e. our set $S$ has a single word $w$ of length $\ell$, we have

\[
G(z) = \frac{C_{w, w}(z)}{z^\ell + (1 - qz)C_{w, w}(z)},
\]

\[
G_{w}(z) = \frac{z^\ell}{z^\ell + (1 - qz)C_{w, w}(z)}.
\]

Remark. Observe that the denominator is the same for both $G(z)$ and $G_{w}(z)$, implying that the corresponding sequences follow the same recurrence relations with different initial terms.

### 2.3. Expected Wait Time

In this section, we determine the expected wait time for a word $w$. When our set $S$ consists of one word $w$, then $G_{w}(z)$ is the generating function describing the number of strings that end with $w$ and do not contain $w$ otherwise. Thus, the expected wait time is $zC_{w, w}(z)$ evaluated at $z = \frac{1}{q}$. The result is the following formula for the expected wait time:

\[
q^\ell C_{w, w} \left( \frac{1}{q} \right) = q \cdot \text{wLw}.
\]
This gives us a closed form for the expected wait time for any word in terms of its autocorrelation. Note that in this case, we are mostly interested where \( q = 2 \), the expected wait time is twice the Conway leading number.

Example 2.8. For the word \( w = HTHT \) that we discussed before, we have \( wLw = 1010_2 = 10 \). The expected wait time for the word \( HTHT \) is twice the Conway leading number which can also be calculated using the autocorrelation polynomial as \( 2^4(1 + \frac{1}{2^2}) = 20 \).

2.3.1. Bounds on the Wait Time. As \( C_0 = 1 \) for any word, the shortest expected wait time for a word of length \( \ell \) is \( q^\ell \). We call such words non-self-overlapping: no proper suffix is equal to a prefix. For instance, the word \( HHTHT \) is non-self-overlapping.

On the other hand, the largest expected wait time is achieved when \( C_i = 1 \) for all \( i < \ell \), which is true for any word consisting entirely of \( H \)s or entirely of \( T \)s. For example, the expected wait time for \( HHH \) is 14.

Example 2.9. With \( (q, \ell) = (2, 5) \), the words

\[
HHHHHT, HHHTT, HHTHT, HHTTT, HTHTT, HTTTT
\]

all have autocorrelation vector \( (1, 0, 0, 0, 0) \) and CLN of \( 2^5 - 1 = 16 \).

2.4. Odds for the Game

In this section, we go back to Penney’s game and determine the winning odds for Alice. Suppose Alice’s and Bob’s words are \( w_1 \) and \( w_2 \) respectively. Then from Theorem 2.6 we have

\[
(1 - qz)G(z) + G_{w_1}(z) + G_{w_2}(z) = 1
\]

\[
G(z) = z^{-\ell_1}C_{w_1,w_1}(z)G_{w_1}(z) - z^{-\ell_2}C_{w_2,w_1}(z)G_{w_2}(z) = 0
\]

\[
G(z) = z^{-\ell_1}C_{w_1,w_2}(z)G_{w_1}(z) - z^{-\ell_2}C_{w_2,w_2}(z)G_{w_2}(z) = 0.
\]

The odds that Alice wins the game is the same as the odds that \( w_1 \) appears before \( w_2 \) in a randomly generated string, which is

\[
\frac{G_{w_1}(\frac{1}{q})}{G_{w_2}(\frac{1}{q})}.
\]

The odds equals

\[
\frac{q^{\ell_2}(C_{w_2,w_2}(\frac{1}{q}) - C_{w_2,w_1}(\frac{1}{q}))}{q^{\ell_1}(C_{w_1,w_1}(\frac{1}{q}) - C_{w_1,w_2}(\frac{1}{q}))},
\]

or, in terms of Conway leading numbers,

\[
\frac{w_2Lw_2 - w_2Lw_1}{w_1Lw_1 - w_1Lw_2}.
\]

In classical literature, this is known as Conway’s formula for the two-player Penney’s game [2].
Example 2.10. Suppose Alice selects $HTHT$, which has an expected wait time of 20 flips. Moreover, let Bob select $THTT$, which has an expected wait time of 18 flips. Surprisingly, despite the fact that Alice’s wait time is longer, she wins with probability $\frac{9}{14}$.

The probability that the game takes $n$ flips is exactly $\frac{1}{q^n}(T_{w_1}(n)+T_{w_2}(n))$, and thus the expected length of the game is

$$\sum_{n=0}^{\infty} \frac{n}{q^n}(T_{w_1}(n)+T_{w_2}(n)) = \frac{G'_{w_1}(\frac{1}{q}) + G'_{w_2}(\frac{1}{q})}{q}.$$

In terms of Conway leading numbers, the expected length of the game is

$$q \cdot \frac{(w_1Lw_1)(w_2Lw_2) - (w_1Lw_2)(w_2Lw_1)}{(w_1Lw_1 + w_2Lw_2) - (w_1Lw_2 + w_2Lw_1)}.$$

2.5. Optimal Strategy for Bob

In this section, we lay out and prove a winning strategy for Bob. Suppose Alice picks a word $w_1$, and Bob wants to pick a word $w_2$ to maximize his odds of winning the game. His odds of winning are

$$\frac{w_1Lw_1 - w_1Lw_2}{w_2Lw_2 - w_2Lw_1},$$

so his best beater would be a word that makes $w_1Lw_2$ relatively small and $w_2Lw_1$ relatively big compared to the other CLNs. As shown in [4,7], a word $w_2$ of the form $w^*w_1(1)w_1(2)\ldots w_1(\ell - 1)$, i.e. a word for which $w_2(2, \ell) = w_1(1, \ell - 1)$ fits the bill quite nicely. In fact, the following theorem is proven in [7].

**Theorem 2.11.** Bob’s best strategy is to pick a word $w_2$ for which $w_2(2, \ell) = w_1(1, \ell - 1)$. In fact, this strategy always gives him odds $> 1$ of winning; these odds approach $q/(q - 1)$ as $\ell \to \infty$.

The proof of this theorem relies heavily on Corollary 2.5 on periods. The exact choice of letter to pick for $w_2(1)$ is determined in [4].

**Example 2.12.** If Alice picks the word $HTHTH$, Bob’s best strategy is to pick the word $HHTHT$. This gives him a 7 : 2 odds of winning.

3. Patterns in Words and Group Action

In this section, we introduce the notion of a pattern representing a set of words, which we use to define Penney’s game with patterns, the central object of study in the paper.
3.1. A Motivating Example
In the classical game, a word is generated by a sequence of letters. A natural extension of a word is a pattern, where we identify a group of similar words with a single string of characters.

Explicitly, for the case \( q = 2 \) we may identify a word composed of \( H \)'s and \( T \)'s with its conjugate, or the result of replacing \( H \)'s with \( T \)'s and vice versa. Alice can choose a pattern for her word. For example, she can decide that all three characters are the same, effectively choosing two words \( HHH \) and \( TTT \).

Bob can choose a pattern where the characters alternate, essentially picking \( HTH \) and \( THT \). That means if the game proceeds \( THHTH \), then Bob wins.

We represent the fact that Alice wants all characters the same as a pattern \( aaa \), where \( a \) could be either \( H \) or \( T \). In other words, to identify both a word and its conjugate collectively, we take the word beginning with \( H \), replace all \( H \)'s with lowercase \( a \)'s and \( T \)'s with lowercase \( b \)'s.

Example 3.1. The pattern \( aaa \) represents the two words consisting of the same letter: \( HHH \) and \( TTT \). The pattern \( aba \) represents two words with alternating letters: \( HTH \) and \( THT \).

We can play Penney’s game with patterns. Suppose Alice picks pattern \( aaa \), and Bob picks pattern \( aba \). Then they flip a coin. If three of the same flips in a row appear first, Alice wins. If three alternating flips in a row appear first, then Bob wins.

3.2. Group Action
We can also extend patterns in an alphabet with two letters, \( H \) and \( T \), to patterns in larger alphabets. Let \( \mathcal{A} \) be an alphabet of size \( q \). We assume that the letters in the alphabet are: \( A, B, C, \) and so on.

Consider a subgroup \( G \subseteq S_q \), where \( S_q \) is a permutation group on \( q \) elements. Group \( G \) acts on the alphabet; formally, we consider the corresponding group action \( \varphi : G \times \mathcal{A} \to \mathcal{A} \). For shorthand, we denote \( \varphi(g, x) = g \cdot x \). Elements of the group send letters to letters, and thus words to words. The order of the group \( G \) is denoted as \(| G |\).

We use \( \text{Orb}_G(w) \) to denote the orbit of the word \( w \) under the action of group \( G \). Two words \( v \) and \( w \) are equivalent if they belong to the same orbit of the group, notated \( v \sim w \). These orbits split the set of words into equivalence classes.

Example 3.2. Consider the group action \( S_3 \) on three letters \( \{A, B, C\} \) which permutes the three letters. Then \( \text{Orb}_{S_3}(ABC) = \{ABC, ACB, BAC, BCA, CAB, CBA\} \), while \( \text{Orb}_{S_3}(AAA) = \{AAA, BBB, CCC\} \).

Within each orbit \( \text{Orb}_G(w) \), we select a canonical representative. By convention, we choose it to be the lexicographically earliest word in this class. We denote the canonical representative as \( s(\text{Orb}_G(w)) \). Thus, we have \( \text{Orb}_G(s(\text{Orb}_G(w))) = \text{Orb}_G(w) \). We call \( s(\text{Orb}_G(w)) \) a pattern. To distinguish between words and patterns, we use lowercase letters for patterns. The canonical representative of a pattern is a word, so we can use the same operation on
patterns as on words. For example, when we take the last letter of a pattern, we assume that we take the last letter of its canonical representative.

Example 3.3. We have $s(\text{Orb}_{S_3}(BCB)) = aba$.

Example 3.4. Under the group action $G = S_3$ which permutes all letters, the 3-letter patterns are divided into 5 orbits with the corresponding canonical representatives: $aaa$, $aab$, $aba$, $abb$, $abc$.

4. Group Action and Correlation

In this section, we examine the properties of correlation under the group action by $G$. Our first proposition shows that the correlations on words are invariant with respect to the group action $\varphi$.

Proposition 4.1. For any two words $v$, $w$ and $g \in G$, we have $\vec{C}(v, w) = \vec{C}(g \cdot v, g \cdot w)$.

Proof. Consider the $i$-th bit of $\vec{C}(v, w)$, where the length of $v$ is $\ell$. It is equal to 1 if and only if $v(\ell + 1 - i, \ell) = w(1, i)$. Because $g$ is bijective on letters, it also must be bijective on words, so $(g \cdot v)(\ell + 1 - i, \ell) = (g \cdot w)(1, i)$. Thus, all bits are equal, and the two correlations are identical. □

Take any orbit of equivalent words $V = \text{Orb}_G(v)$. The sum of the correlations of the word $w$ with all words in orbit $V$ is the same for all words in the orbit of $w$.

Corollary 4.2. For any two equivalent words $w_1 \sim w_2$, we have

$$\sum_{v_i \in V} \vec{C}(w_1, v_i) = \sum_{v_i \in V} \vec{C}(w_2, v_i) \quad \text{and} \quad \sum_{v_i \in V} \vec{C}(v_i, w_1) = \sum_{v_i \in V} \vec{C}(v_i, w_2).$$

Proof. By definition, there is some $g \in G$ such that $g \cdot w_1 = w_2$. As $V$ is a complete orbit, we have that $g \cdot V = V$. In other words,

$$\sum_{v_i \in V} \vec{C}(w_1, v_i) = \sum_{v_i \in V} \vec{C}(g \cdot w_1, g \cdot v_i) = \sum_{v_i \in V} \vec{C}(w_2, v_i),$$

so the two sums are equal as desired. The second claim follows similarly. □

4.1. The Weight of a Substring in a Word

The stabilizer $\text{Stab}_G(w)$ of a word $w$ is the set $\{g \in G \mid g \cdot w = w\}$. Note that two words in the same orbit have the same stabilizers. Thus, the stabilizer of a pattern $p$ is well-defined; denote this stabilizer as $\text{Stab}_G(p)$.

Definition 4.3. Given a word $w$ and its substring $w(i, j)$, the weight of the substring in the word $w$ is the number of words $v$ in the orbit $\text{Orb}_G(w)$ such that $v(i, j) = w(i, j)$.

Example 4.4. Consider the word $BBC$ in an alphabet with 3 letters and group $S_3$, then the corresponding orbit is $\{AAB, AAC, BBA, BBC, CCA, CCB\}$. Hence, the weight of the suffix $C$ is 2 as there are two words $BBC$ and $AAC$ in the orbit that have suffix $C$. 
The number of words in the orbit of a word \( w \) is \(|G / \text{Stab}_G(w)|\). If \( x \) is a substring in \( w \), then \( \text{Stab}_G(w) \) is a subgroup of \( \text{Stab}_G(x) \). Thus, the number of words in the orbit of \( w \) that have \( x \) as a substring in the same place is \(|G / \text{Stab}_G(x)|\). Therefore, the weight of the substring \( x \) in the word \( w \) equals \(|\text{Stab}_G(x)| / |\text{Stab}_G(w)|\).

4.2. Correlation Polynomials and Conway Leading Number for Patterns

Now we define the correlation polynomial between two patterns \( p \) and \( p' \) that do not have to be the same length. Let \( \ell \) be the length of \( p \).

**Definition 4.5.** The **correlation vector** between two patterns \( p \) and \( p' \) is denoted \( \vec{C}(p, p') = (C_0, \ldots, C_{\ell-1}) \), where \( C_i \) is defined as follows.

- If the suffix \( x \) of \( p \) of length \( \ell - i \) is equivalent to the prefix of \( p' \) of length \( \ell - i \), then \( C_i \) is the weight of the suffix \( x \) in the word \( p \),
- If they are not equivalent, then \( C_i = 0 \).

This definition shares a few similarities with the correlation vector on words. The first entry \( C_0(p, p') \) is 1 if \( p \sim p' \) and is 0 otherwise. Moreover, all entries of \( \vec{C}(p, p') \) are integers.

**Definition 4.6.** The **correlation polynomial** of \( p \) and \( p' \) is defined as \( C_{p,p'}(z) = C_0 z^0 + \cdots + C_{\ell-1} z^{\ell-1} \). It is a polynomial of degree at most \( \ell - 1 \).

Similar to before, the Conway leading number between two patterns is defined as the value of the correlation vector interpreted as a string in base \( q \) and it is denoted as \( p \mathcal{L} p' \).

**Definition 4.7.** The **Conway leading number (CLN)** for two patterns \( p \) and \( p' \) is

\[
p \mathcal{L} p' = q^{\ell-1} \mathcal{C}_{p,p'} \left( \frac{1}{q} \right).
\]

We provide another definition of correlation with the following proposition.

**Proposition 4.8.** Pick any word \( v \) belonging to the orbit represented by the pattern \( p \), and \( w \) belonging to the orbit represented by \( p' \). Then

\[
\vec{C}(p, p') = \sum_{v_i \in \text{Orb}_G(v)} \vec{C}(v_i, w).
\]

**Proof.** First notice that

\[
\sum_{v_i \in \text{Orb}_G(v)} \vec{C}(v_i, w) = \frac{1}{|\text{Stab}_G(v)|} \sum_{g \in G} \vec{C}(g \cdot v, w).
\]

Focus on a single \( j \)-th entry of the formula above. This entry is equal to the number of elements \( g \in G \) for which the suffix \( g \cdot v(j+1, \ell) \) of length \( \ell - j \) matches the prefix \( w(1, \ell - j) \). If these two words are not equivalent, then the count is 0. If the two words are equivalent, the count is equal to the order of the stabilizer of \( v(j+1, \ell) \). Thus, the \( j \)-th entry of the equation above equals \( C_j \) in the definition of the correlation as desired. \( \square \)
The proposition above shows another way to define the correlation between patterns. Note that we cannot swap \( v \) and \( w \) in this definition. This is due to the following formula

\[
|\text{Stab}_G(v)| \sum_{v_i \in \text{Orb}_G(v)} \vec{C}(v_i, w) = |\text{Stab}_G(w)| \sum_{w_i \in \text{Orb}_G(w)} \vec{C}(v, w_i),
\]

which is true since after multiplying each vector sum by the constant preceding it, the \( i \)-th entry are either 0 or are both the order of the stabilizer of \( w(1, \ell - i) \).

**Example 4.9.** Consider the autocorrelation vector \( \vec{C}(abc, abc) \) with respect to the group \( S_3 \). The weight of the suffix \( abc \) in \( abc \) is 1. The weight of the suffix \( bc \) in \( abc \) is 1, the weight of the suffix \( c \) in \( abc \) is 2. Thus the correlation vector is \((1, 1, 2)\). Another calculation can be done using Proposition 4.8. The autocorrelation is

\[
\vec{C}(ABC, ABC) + \vec{C}(ACB, ABC) + \vec{C}(BAC, ABC) + \vec{C}(BCA, ABC) + \vec{C}(CAB, ABC) + \vec{C}(CBA, ABC)
\]

which evaluates to \((1, 0, 0) + (0, 0, 0) + (0, 0, 1) + (0, 1, 0) + (0, 0, 1) = (1, 1, 2)\). The corresponding correlation polynomial is thus \( C_{abc, abc}(z) = 1 + z + 2z^2 \).

### 4.3. Correlation Vector

In this section, we generalize the notion of correlation vector from words to patterns. We say that a pattern \( p = p(1)p(2)\ldots p(\ell) \) has period \( i \) if the \( i \)-th entry \( C_i \) of the vector \( \vec{C}(p, p) \) is nonzero, i.e. if

\[
p(1)p(2)\ldots p(\ell - i) \sim p(i + 1)p(i + 2)\ldots p(\ell),
\]

or, equivalently, there exists \( g \in G \) such that \( g \cdot p(1, \ell - i) = p(i + 1, \ell) \).

Note that if \( p \) has period \( s \) when the equivalence is realized by the group element \( g \), then because \( g \cdot (g \cdot p_i) = p_{i+2s} \) for \( i + 2s \leq \ell \), we see that \( p \) also has period \( 2s \). In this fashion, we get the following proposition in the fashion of Sect. 2.

**Proposition 4.10.** If a pattern has period \( s \), it also has period \( ks \) for all \( k \geq 1 \). More generally, if a pattern has period \( s \) and period \( t \), it also has period \( s + t \).

Like in Sect. 2, the reverse implication is not true; if a pattern has periods \( s < t \), it does not necessarily have period \( t - s \).

**Example 4.11.** The pattern \( p = abcdace \) with symmetric group \( G = S_5 \) has \( \vec{C}(p, p) = (1, 0, 1, 1, 2, 6, 24) \). In particular, it has periods 2, 3, 4, 5, and 6, but not period 1.

The reason that the above example does not satisfy the reverse implication is that the elements

\[
g_2 = \begin{pmatrix} a & b & c & d & e \\ c & d & a & e & b \end{pmatrix}, \quad g_3 = \begin{pmatrix} a & b & c & d & e \\ d & a & e & c & b \end{pmatrix}
\]
are the unique elements in $G$ for which $g_2 \cdot p(1, 5) = p(3, 7)$ and $g_3 \cdot p(1, 4) = p(4, 7)$. But they do not commute: $g_2 g_3 \cdot p(3) = b \neq d = g_3 g_2 \cdot p(3)$, meaning that $p$ cannot be extended to an eighth letter while having periods 2 and 3.

It thus makes sense that the implication is true for sufficiently long patterns since a large number of letters forces the respective group elements for each period to commute. In fact, we have the following lemma.

**Lemma 4.12.** Let $p$ be a pattern with length $\ell \geq (q + 1)s + t$ with periods $s < t$. Then $t - s$ is also a period of $p$.

To prove this, we need a quick lemma.

**Lemma 4.13.** Let $p$ be a pattern with length $\ell \geq qs$ and period $s$. Then each of the letters of $p$ appear somewhere in the first $qs$ letters of $p$.

**Proof.** By definition, there is an element $g \in G$ for which $g \cdot p(1, \ell - s) = p(s + 1, \ell)$. Suppose the letter $x$ appears in $p$ and has index $k$: $x = p(k)$. Then the letter $y = p(k \mod s)$ must be in the same orbit as $x$ under $g$. Consider the sequence

$$y, \quad g \cdot y, \quad g^2 \cdot y, \quad \ldots, \quad g^{q-1} \cdot y.$$ 

Because the orbit has at most $q$ letters, this sequence contains the entire orbit of $y$. In particular, it must contain $x$. So there is some integer $i$ for which $x = g^i \cdot y$, which has index less than $qs$. \hfill \Box

We now prove Lemma 4.12.

**Proof of Lemma 4.12.** Because $p$ has period $s$, there is an element $g_s \in G$ for which $g_s \cdot p(1, \ell - s) = p(s + 1, \ell)$. Similarly, because $p$ has period $t$, there is an element $g_t \in G$ for which $g_t \cdot p(1, \ell - t) = p(t + 1, \ell)$.

Thus, for all $1 \leq i \leq qs$, we have $g_s g_t \cdot p(i) = p(i + s + t) = g_t g_s \cdot p(i)$. By Lemma 4.13, the letter $p(i)$ ranges over all letters in $p$ as we vary $i$ in this range. So $g_s$ and $g_t$ commute for all letters in $p$. This implies $g_s^{−1}$ and $g_t$ also commute for all letters in $p$. In particular, we have

$$g_s^{−1} g_t \cdot p(1, \ell + s - t) = [g_s^{−1} g_t \cdot p(1, \ell - t)][g_s^{−1} g_t \cdot p(\ell - t + 1, \ell - t + s)] = [g_s^{−1} g_t \cdot p(1, \ell - t)][g_t g_s^{−1} \cdot p(\ell - t + 1, \ell - t + s)] = [p(−s + t + 1, \ell - s)][p(\ell - s + 1, \ell)] = p(t - s + 1, \ell),$$

so $p$ has period $t - s$. \hfill \Box

In particular, from Lemma 4.12 we have the following corollary.

**Corollary 4.14.** If $p$ has length $\ell$, least period $s$ and period $t$ not divisible by $s$, then $t \geq \ell/(q + 2) + 1$.

**Proof.** If $t$ is the least period not divisible by $s$, then $t - s$ cannot be a period. The contrapositive of Lemma 4.12 implies that $\ell \leq (q + 1)s + t - 1$. Since $s + 1 \leq t$, we have $\ell \leq (q + 2)t - q - 2$ or $t \geq \ell/(q + 2) + 1$, as desired. \hfill \Box
4.4. Non-self-Overlapping Patterns

Recall that, for a word $w$ of length $\ell$, the vector $\vec{C}(w, w) = (C_0, C_1, \ldots, C_{\ell-1})$ satisfies $C_0 = 1$ for $1 \leq i < \ell$, so all CLNs are at least $q^{\ell-1}$. This bound is exact. It is achieved when no proper suffix is equivalent to a prefix, that is for non-self-overlapping words.

We call a pattern **non-self-overlapping** if no proper prefix is equivalent to suffix. Such a pattern can only exist if the alphabet is not one orbit of group $G$, that is, if there are two letters that are not equivalent to each other. Indeed, the last letter and the first letter of a pattern should not be equivalent in a non-self-overlapping pattern.

We call a pattern **almost-non-self-overlapping** if no proper prefix is equivalent to a suffix, except for the prefix of length 1.

**Example 4.15.** Consider pattern $aaa \cdots aaab$. If $a$ and $b$ belong to different equivalent classes under the group action, then the pattern is non-self-overlapping. Otherwise, it is almost-non-self-overlapping.

4.5. Lower Bound for CLN

The CLN lower bound $q^{\ell-1}$ for words is not always achievable for patterns. We strengthen the lower bound with the following claim.

**Theorem 4.16.** Consider the lowest CLN achieved by any pattern of length $\ell$.

(i) If the action by $G$ is not transitive over the alphabet, then the lowest possible CLN is $q^{\ell-1}$.

(ii) Otherwise, the lowest possible CLN is between $q^{\ell-1} + 1$ and $q^{\ell-1} + q - 1$ inclusive.

**Proof.** To begin, we know that $C_0 = 1$, and so $pLp \geq q^{\ell-1}$.

If the group action by $G$ is not transitive, then there are two letters $A$ and $B$ that belong to different orbits. As such, any non-self-overlapping word in the alphabet \{A, B\} of length $\ell$ is in an orbit whose corresponding pattern has CLN equal to $q^{\ell-1}$, proving (i).

If the action is transitive, then the last character of pattern $p$ is equivalent to the first character, or $C_{\ell-1} \geq 1$ and therefore $pLp \geq q^{\ell-1} + 1$.

Consider the pattern $p = aa \ldots ab$. The number of elements in the orbit of the word $s(p)$ that end in $B$ does not exceed $q - 1$. It follows that $pLp \leq q^{\ell-1} + q - 1$. This pattern $p$ provides the exact values for the lower bound for groups $\mathbb{Z}/q\mathbb{Z}$ and $S_q$. \hfill $\square$

Note that a word that achieves the lower bound for words may not correspond to a pattern that achieves the lower bound for patterns.

**Example 4.17.** For $q = 2$, the word $AABB$ has autocorrelation $(1, 0, 0, 0)$, thus, achieving the lower bound for CLN for words. If we consider the group $S_2$, then the pattern $aabb$ has autocorrelation $(1, 0, q - 1, q - 1)$, which does not match the minimum CLN.

In general, there does not appear to be a simple rule to generate all patterns achieving the lower bound.
5. Generating Functions

In this section, we determine explicit generating functions for Penney’s game with patterns, generalizing Theorem 2.6 to sets of patterns. A reduced set of patterns is a set where there are no patterns \( p \) and \( p' \), such that some substring of \( p' \) defines a pattern equivalent to \( p \). For instance, the set \( \{aba, aabcb\} \) for \( G = S_3 \) is not reduced, as \( aba \sim bcb \). Note that if a set of patterns is reduced, the joint set of words formed by the union of their orbits is also reduced. The reverse direction does not hold: for instance, the set of words \( \{ABA, AABCB\} \) is reduced while the corresponding set of patterns \( \{aba, aabcb\} \) is not.

Fix a group \( G \) to form our group action; let \( S = \{p_1, p_2, \ldots, p_k\} \) denote a reduced set of \( k \) patterns with lengths \( \ell_1, \ell_2, \ldots, \ell_k \), all composed of letters from an alphabet of size \( q \). It is important that the set \( S \) is reduced for the same reasons as before.

Once more, define

- \( A(n, S) \) to be the number of words (not patterns) of length \( n \) not containing any subword represented by any pattern \( p \in S \); and
- \( T_{p_i}(n, S) \) to be the number of words (not patterns) of length \( n \) not containing any subword represented by any pattern \( p \in S \) other than a single word represented by the pattern \( p_i \) at the end of the word; this suffix is required.

Then we set the generating functions

\[
G(z, S) = \sum_{k=0}^{\infty} A(k, S) z^k, \quad G_{p_i}(z, S) = \sum_{k=0}^{\infty} T_{p_i}(k, S) z^k,
\]

and sometimes drop \( S \) when it is clear which set \( S \) of patterns we are referring to. We verify a quick proposition about generating functions of equivalent words. Given a set of patterns \( S \), we denote by \( S \) the set of words that corresponds to the union of the orbits of all the patterns.

**Proposition 5.1.** Take a set \( S \) of words which is invariant under the group action \( \varphi \) induced by the group \( G \). Then for any equivalent words \( v \sim w \) in \( S \), we have \( G_v(z, S) = G_w(z, S) \).

**Proof.** Take a \( g \in G \) for which \( g \cdot w = v \). Then for any first occurrence word \( x \) containing \( w \), the word \( g \cdot x \) is a first occurrence word containing \( v \). This word still avoids all other words in \( S \), since \( S \) is invariant under \( G \).

As \( g \) is invertible, we have \( T_w(n, S) = T_v(n, S) \) and the two generating functions are identical. \( \square \)

Finally, let the orbits corresponding to the patterns \( p_1, p_2, \ldots, p_k \) have sizes \( r_1, r_2, \ldots, r_k \) respectively. We derive a system of equations in the manner of Theorem 2.6.

**Theorem 5.2.** The generating functions \( G(z), G_{p_1}(z), G_{p_2}(z), \ldots, G_{p_k}(z) \) satisfy the following system of equations:

\[
(1 - qz)G(z) + G_{p_1}(z) + G_{p_2}(z) + \cdots + G_{p_k}(z) = 1
\]
new set of words is also reduced. The first equation of Theorem 2.6 becomes

\[ G(z) - \frac{1}{r_1} z^{-\ell_1} C_{p_1,p_1}(z) G_{p_1}(z) - \ldots - \frac{1}{r_k} z^{-\ell_k} C_{p_k,p_1}(z) G_{p_k}(z) = 0 \]

\[ G(z) - \frac{1}{r_1} z^{-\ell_1} C_{p_1,p_2}(z) G_{p_1}(z) - \ldots - \frac{1}{r_k} z^{-\ell_k} C_{p_k,p_2}(z) G_{p_k}(z) = 0 \]

\[ \vdots \]

\[ G(z) - \frac{1}{r_1} z^{-\ell_1} C_{p_1,p_k}(z) G_{p_1}(z) - \ldots - \frac{1}{r_k} z^{-\ell_k} C_{p_k,p_k}(z) G_{p_k}(z) = 0. \]

**Proof.** As before, we denote the orbit of words represented by \( p_i \) as \( \text{Orb}_G(p_i) \). The key is to evaluate Theorem 2.6 with the set of words

\[ S = \bigcup_{i=1}^k \text{Orb}_G(p_i) = \text{Orb}_G(p_1) \sqcup \text{Orb}_G(p_2) \sqcup \cdots \sqcup \text{Orb}_G(p_k), \]

or the set of words composed of the union of the orbits represented by \( p_1, p_2, \ldots, p_k \). These orbits are pairwise disjoint since \( S \) is reduced; in addition, this new set of words is also reduced. The first equation of Theorem 2.6 becomes

\[ 1 = (1 - qz)G(z,S) + \sum_{w \in \text{Orb}_G(p_1)} G_w(z,S) + \cdots + \sum_{w \in \text{Orb}_G(p_k)} G_w(z,S) \]

\[ = (1 - qz)G(z) + G_{p_1}(z) + \cdots + G_{p_k}(z). \]

In addition, for any word \( v \in \text{Orb}_G(p_j) \), we have \( \sum_{w \in \text{Orb}_G(p_i)} C_{w,v}(z) = C_{p_i,p_j}(z) \) by our alternative definition of correlation. Moreover,

\[ G_{p_j}(z,S) = \sum_{w \in \text{Orb}_G(p_j)} G_w(z,S) = r_j G_v(z,S) \]

by Proposition 5.1. Thus, the equation from Theorem 2.6 corresponding to a word \( w \in \text{Orb}_G(p_i) \) becomes

\[ G(z) - \frac{1}{r_1} z^{-\ell_1} C_{p_1,p_1}(z) G_{p_1}(z) - \ldots - \frac{1}{r_k} z^{-\ell_k} C_{p_k,p_1}(z) G_{p_k}(z) = 0, \]

as desired. \( \square \)

### 5.1. Normalized Correlation for Patterns

In this section, we introduce normalized correlation. Note the striking similarity between Theorem 2.6 and Theorem 5.2. Namely, we can get from Theorem 2.6 to the other by replacing the correlation polynomial for \( C_{w_i,w_j}(z) \) with \( \frac{1}{r_i} C_{p_i,p_j}(z) \). To formalize this, we define the normalized correlation vector for patterns.

**Definition 5.3.** Let \( p \) and \( p' \) be patterns, and let \( p \) represent an orbit of size \( r \). The **normalized correlation vector** \( \tilde{C}^*(p,p') \) is equal to \( \frac{1}{r} \tilde{C}(p,p') \). Similarly, the **normalized correlation polynomial** \( \tilde{C}^*_p(z) \) is equal to \( \frac{1}{r} \tilde{C}_{p,p'}(z) \).

While we default to the previous definition of correlation, normalized correlation has many nice properties. Let \( p \) have length \( \ell \). First, by definition the \( i \)-th entry of \( \tilde{C}^*_p(p,p') \) is zero if the prefix \( p(1, \ell - i) \) and the suffix \( p'(i + 1, \ell) \) are not equivalent, and \( \frac{1}{r} |\text{Stab}_G(p(1, \ell - i))/\text{Stab}_G(p)| = |\text{Stab}_G(p(1, \ell - i))/\text{Stab}_G(p)| \).
if the prefix and suffix are equivalent. The \( i \)-th entry can be interpreted as the fraction of the elements of \( G \) which fix \( p(1, \ell - i) \), which is a rational number between 0 and 1.

We can also think of normalized correlation as an average of \( |G| \) correlations. Specifically, pick a word \( v \) from the orbit represented by \( p \), and a word \( w \) from the orbit represented by \( p' \). Using Proposition 4.8 we can write

\[
\bar{C}^*(p, p') = \frac{1}{|G|} \sum_{g \in G} \bar{C}(g \cdot v, w),
\]

which is the average of the \( |G| \) correlations consisting of a varying word from the orbit represented by \( p \) and a fixed word from the orbit represented by \( p' \).

Finally, normalized correlation is fixed regardless of whether we choose to vary the word associated to \( p \) and fix the word associated to \( p' \), or vice versa. Explicitly, we have

\[
\bar{C}^*(p, p') = \frac{1}{|G|} \sum_{g \in G} \bar{C}(g \cdot v, w) = \frac{1}{|G|} \sum_{g \in G} \bar{C}(v, g \cdot w).
\]

To derive the pattern analogues of various results for words, we obey the following principle.

**Main Principle** For any result on words that is derived from Theorem 2.6 (such as generating functions, odds, etc.), we replace any word correlation polynomial \( C_{v,w}(z) \) with the normalized pattern correlation polynomial \( \bar{C}_{p,p'}(z) \) to get the corresponding result derived from Theorem 5.2.

For instance, the following corollary is immediate.

**Corollary 5.4.** For \( k = 1 \) we get

\[
G(z) = \frac{C_{p,p}(z)}{z^\ell + (1 - qz)C_{p,p}(z)} = \frac{C_{p,p}(z)}{rz^\ell + (1 - qz)C_{p,p}(z)},
\]

\[
G_p(z) = \frac{z^\ell}{z^\ell + (1 - qz)C_{p,p}(z)} = \frac{rz^\ell}{rz^\ell + (1 - qz)C_{p,p}(z)}.
\]

We see that here, as with words, the denominator is the same for both functions.

### 6. The Expected Wait Time

In this section, we examine the expected wait time for a pattern. Given that each letter appears with equal probability \( \frac{1}{q} \), the expected wait time is just \( q^\ell \bar{C}^*(p, p) = \frac{1}{q} \sum_{p' \in \mathbb{Q}} q^\ell \bar{C}(p, p') = \frac{1}{q} \cdot pLp \). In terms of the Conway leading number, the expected wait time is \( \frac{q}{r} \cdot pLp \).

**Example 6.1.** For \( q = 2 \), the pattern \( p = abab \) has \( \bar{C}(p, p) = (1, 1, 1, 1) \) and orbit size 2, and thus \( pLp = 11112 = 15 \). Therefore, the pattern has expected wait time \( \frac{q}{r} \cdot (15) = 15 \).
We may rewrite the expected value formula as

\[ \frac{1}{|G|} \sum_{i=1}^{\ell} q^i |\text{Stab}_G(p)| \mathcal{C}_i(p, p), \]

by the orbit-stabilizer theorem. Note that \( |\text{Stab}_G(p)| \mathcal{C}_i(p, p) \) is equal to 0 if \( p(1, \ell - i) \neq p_{i+1, \ell} \), and \( |\text{Stab}_G(p(1, \ell - i))| \) otherwise.

### 6.1. Bounds on the Expected Wait Time

We present two similar results, explicitly giving patterns that achieve the lowest and highest possible expected wait time.

**Proposition 6.2.** Let \( x \in A \) be the letter with the largest stabilizer. Then the greatest expected wait time of a pattern of length \( \ell \) is

\[ \frac{|\text{Stab}_G(x)|}{|G|} \left( q + q^2 + \cdots + q^\ell \right), \]

achieved by the pattern \( p = xx \ldots x \).

**Proof.** In fact, we make the stronger claim that the maximal value of each individual coefficient \( |\text{Stab}_G(p)| \mathcal{C}_i(p, p) \) of the expansion of \( \frac{1}{r} p \mathcal{L} p \) is achieved at \( xx \ldots x \), using the same letter \( x \) for each \( i = 0, 1, 2, \ldots, \ell - 1 \). This certainly implies the original claim. Assume \( \mathcal{C}_i > 0 \), so it remains to maximize \( |\text{Stab}_G(p)| \mathcal{C}_i(p, p) = |\text{Stab}_G(p(1, \ell - i))| \).

Suppose the prefix \( p(1, \ell - i) \) contains the letters \( a_1, a_2, \ldots, a_k \). Then the size of the stabilizer of the prefix is at most the size of the stabilizer of \( a_1 \), so we may as well assume the prefix (and the pattern) consists of a single letter. Then we select the letter \( x \) with the largest stabilizer (if there are ties, pick one arbitrarily) to comprise our pattern. \( \square \)

**Proposition 6.3.** Let \( \ell \geq q + 1 \) be an integer. If the action by \( G \) is transitive, then the least expected wait time of a pattern of length \( \ell \) is \( q^\ell / |G| \). Otherwise, the least expected wait time is \( q^\ell / |G| + 1 \).

**Proof.** Consider an almost-self-non-overlapping pangrammatic (a word containing every letter) \( p = a_1 a_2 \ldots a_{q-1} a_q \ldots a_q \). We have

1. \( |\text{Stab}_G(p)| = 1 \) as \( p \) contains all \( q \) letters of \( A \), and
2. \( \mathcal{C}_i = 0 \) for all \( 1 \leq i \leq \ell - 1 \) as \( p \) is almost-self-non-overlapping.

We first tackle the case where the action is nontransitive. Pick \( x \) and \( y \) such that \( x \notin \text{Orb}_G(y) \). Without loss of generality we can assume that \( a_1 = x \) and \( a_q = y \). Then \( \mathcal{C}_1 = 0 \).

Note that this pattern achieves the least possible value of \( \frac{1}{r} \mathcal{C}_i \) for all \( 1 \leq i \leq \ell - 1 \). Moreover, since \( \frac{1}{r} \mathcal{C}_0 = |\text{Stab}_G(p)| / |G| \) and \( |\text{Stab}_G(p)| = 1 \) for a pattern \( p \) containing all letters in \( A \), such a pattern also minimizes \( \mathcal{C}_0 \) as well. Thus, this pattern achieves the least possible expected wait time.

If the action by \( G \) is transitive, then we must have \( \mathcal{C}_{\ell-1} > 0 \). The pattern \( p \) still minimizes \( \mathcal{C}_i \) for \( 0 \leq i \leq \ell - 2 \). Moreover, note \( \frac{1}{r} \mathcal{C}_{\ell-1} = |\text{Stab}_G(p(1, 1))| / |G| \). Because all letters are in the same orbit, they all have
a stabilizer of size $|G|/q$ by the orbit-stabilizer theorem. Thus, $\frac{1}{q}C_{\ell-1} = 1/q$ is a constant, and as such $p$ still minimizes every $C_\ell$ and therefore the expected wait time. The claim now follows.

\[ \square \]

7. Odds

In this section, we compute the winning probabilities for a game with patterns using Theorem 5.2 and the Main Principle.

**Theorem 7.1.** Suppose Alice and Bob pick the patterns $p_1$ and $p_2$, with lengths $\ell_1, \ell_2$ and orbit sizes $r_1, r_2$ respectively. We assume that patterns are such that $\mathcal{S} = \{p_1, p_2\}$ is reduced. Then the odds of Alice winning the game are:

$$
\frac{q^{\ell_2}(C_{p_2, p_2}(\frac{1}{q}) - C_{p_2, p_1}(\frac{1}{q}))}{q^{\ell_1}(C_{p_1, p_2}(\frac{1}{q}) - C_{p_1, p_1}(\frac{1}{q}))} = \frac{r_1q^{\ell_2}(C_{p_2, p_2}(z) - C_{p_2, p_1}(z))}{r_2q^{\ell_1}(C_{p_1, p_1}(z) - C_{p_1, p_2}(z))} = \frac{r_1}{r_2}. \frac{p_2L_{p_2} - p_2L_{p_1}}{p_1L_{p_1} - p_1L_{p_2}}.
$$

We see that to get to the formula for patterns from the formula for words in calculating the odds, we need to replace the correlation between words with the correlation between patterns and then multiply the result by $\frac{C_{\ell_1}}{C_{\ell_2}}$.

Given the adjusted odds, we can adjust the expected length of the game.

**Corollary 7.2.** The expected length of the game is

$$
q \cdot \frac{(p_1L_{p_1})(p_2L_{p_2}) - (p_1L_{p_2})(p_2L_{p_1})}{r_1(p_2L_{p_2} - p_2L_{p_1}) + r_2(p_1L_{p_1} - p_1L_{p_2})}.
$$

**7.1. Optimal Strategy for Bob**

Recall from Sect. 2 that for the game on words, Bob has a winning strategy. Specifically, if Alice picks the word $w(1)w(2)\ldots w(\ell)$ the best choice for Bob is the word $w^*w(1)w(2)\ldots w(\ell-1)$ for some $w^*$ which makes his odds of winning greater than 1. In this section, we show the following pattern analogue of the strategy for words.

**Theorem 7.3.** Fix $q$ and let $\ell$ be sufficiently large. If Alice picks the pattern $p_1 = p_1(1)p_1(2)\ldots p_1(\ell)$, then Bob’s best beater is a pattern $p_2$ for which $p_2(2, \ell) = p_2(1, \ell - 1)$. Bob can always choose such a word such that his odds of winning exceed 1. Moreover, as $\ell \to \infty$ these winning odds approach $|\text{Stab}_G(p_1)|/|\text{Stab}_G(p_2)| \cdot q/(q-1)$.

Note that, in this case, we prove Bob wins for sufficiently large $\ell$. In Sect. 8.2, we provide a family of examples with small $\ell$ where Alice actually has a winning strategy.

To prove this, we generalize the methods used in [7], which showed for words satisfying $w_2(2, \ell) = w_1(1, \ell - 1)$, the quantity $w_1Lw_2$ is relatively negligible while $w_2Lw_1$ is relatively large, causing Bob’s odds $\frac{w_1Lw_1 - w_1Lw_2}{w_2Lw_2 - w_2Lw_1}$ to be large. This all stems from the results on periods in Sect. 2, which we generalized in Sect. 4. We now use those generalizations to prove Theorem 7.3.

Beforehand, we prove a lemma, now showing that Bob can pick a pattern satisfying certain conditions.
Lemma 7.4. If Alice chooses the pattern $p_1 = p_1(1)p_1(2)\ldots p_1(\ell)$, Bob can pick a pattern $p_2 = p_2(1)p_1(1)\ldots p_1(\ell - 1)$ such that:

- If $p_2$ has a period $t$, then $t \geq \ell/(q + 2) + 1$.
- We have $|\text{Stab}_G(p_2)| \leq |\text{Stab}_G(p_1)|$.

Proof. Let $s$ be the least nontrivial period of $p_1$. If $s \geq \ell/(q + 2) + 1$, then we are trivially done, so assume $s \leq \ell/(q + 2) + 1$. Then by Lemma 4.13 the letter $p_1(\ell)$ appears in the first $qs \leq \ell - 1$ letters of $p_2$, namely $p_1(1, \ell - 1)$, so it appears in $p_2$, implying $p_2$ contains all of the letters in $p_1$ and so $|\text{Stab}_G(p_2)| \leq |\text{Stab}_G(p_1)|$.

It suffices to pick $p_2$ such that $ks$ is not a period of $p_2$ for any $k$: if we do, the smallest possible period of $p_2$ is at least the smallest period of $p_1$ not divisible by $s$, which we know to be at least $\ell/(q + 2) + 1$. Note that either $p_1(s)$ and $p_1(s + 1)$ are the same letter, or different letters. If they are the same, then $p_1(ks) = p_1(ks + 1)$ as well, and picking $p_2(1) \neq p_1(1)$ is enough: that way,

$$p_2(1)p_2(2) = p_2(1)p_1(1) \not\sim p_1(ks)p_1(ks + 1) = p_2(ks + 1)p_2(ks + 2),$$

since the former has two different letters and the latter has two of the same letter, and so $p_2$ does not have period $ks$ for any $k$. Similarly, if $p_1(s) \neq p_1(s + 1)$, then pick $p_2(1) = p_1(1)$. $\square$

We are now ready to prove Theorem 7.3.

Proof of Theorem 7.3. Let Bob pick a pattern that satisfies the two conditions in Lemma 7.4. We claim Bob has winning odds for sufficiently large $\ell$, i.e. there is an $M$ for which $\ell > M$ implies

$$\frac{1}{r_1}p_1\mathcal{L}p_1 - \frac{1}{r_1}p_1\mathcal{L}p_2 > 1.$$

To do this, we bound each of the terms

$$\frac{1}{r_m}p_m\mathcal{L}p_n = C^\ast_m(p_m, p_n)q^{\ell - 1} + \cdots + C^\ast_{\ell - 1}(p_m, p_n).$$

Note that all normalized coefficients are between 0 and 1.

First, because $C^\ast_{t+1}(p_1, p_2) > 0$ if and only if $p_2$ has period $t$, i.e. $t \geq \lceil \ell/(q + 2) \rceil + 1$, all of the coefficients $C^\ast_0(p_1, p_2), \ldots, C^\ast_{\lceil \ell/(q + 2) \rceil + 1}(p_1, p_2)$ are zero, and since all other coefficients are at most 1 we have the bound

$$\frac{1}{r_1}p_1\mathcal{L}p_2 \leq q^{[(q+1)\ell/(q+2)]-3} + q^{[(q+1)\ell/(q+2)]-4} + \cdots + 1 = O(q^{(q+1)\ell/(q+2)}).$$

Similarly, since $C^\ast_t(p_2, p_2) = C^\ast_{t+1}(p_2, p_1)$, both of which are nonzero if and only if $t$ is a period of $p_2$, we may bound the denominator:

$$\frac{1}{r_2}p_2\mathcal{L}p_2 - \frac{1}{r_2}p_2\mathcal{L}p_1$$

$$= \sum_{i=0}^{\ell-1} C^\ast_i(p_2, p_2)q^{\ell-1-i} - \sum_{i=0}^{\ell-1} C^\ast_i(p_2, p_1)q^{\ell-1-i}$$
In this section, we discuss specific results for two groups:

\[ \text{8. Specific Groups} \]

In Sect. 8.1, where we show this case is equivalent to the \( q = 2 \) case on words. \( \square \)

8. Specific Groups

In this section, we discuss specific results for two groups \( G \): the cyclic group \( \mathbb{Z}/q\mathbb{Z} \), and the symmetric group \( S_q \).
8.1. Cyclic Group

In this section, we consider Penney’s game equipped with a group action induced by \( G = \mathbb{Z}/q\mathbb{Z} \) that cycles the alphabet. We assume that the letters in the alphabet \( \mathcal{A} \) have an assigned order, and thus may be numerically indexed with the residues modulo \( q \). Then the group action \( \varphi \) acts by translation, where translation by \( g \) sends the letter numbered \( i \) to the letter numbered \( i + g \) (mod \( q \)). It is immediately clear that each pattern represents an orbit of size \( q \). This group action on words is known classically as a Caesar shift.

As the letters are numbered, we may “subtract” one letter from another, obtaining a set of differences. The key step is to note that these differences are preserved under translation by \( g \). We formalize this logic with the following definition.

**Definition 8.1.** The adjacency signature \( S(p) \) of a pattern \( p \) is a word \( S = s(1) \ldots s(\ell - 1) \) in which the index of \( s(i) \) is the unique element of \( G \) sending \( p(i) \) to \( p(i + 1) \) under \( \varphi \).

Consider a given word \( w \) and its orbit \( \text{Orb}_G(w) \) of size \( q \). Since the adjacency signature operation is invariant under translation by \( g \in G \), every word in an orbit under \( \varphi \) maps to the same adjacency signature. This map is invertible, so this grouping forms a \( q \)-to-1 bijection between words of length \( \ell \) and adjacency signatures of length \( \ell - 1 \).

Now, as usual, let \( \mathcal{A}(n, \{p\}) \) and \( T_p(n, \{p\}) \) denote the number of avoiding and first occurrence outputs of length \( n \) for the pattern \( p \).

**Theorem 8.2.** Let \( n \) be a nonnegative integer, and \( p \) a pattern of length \( \ell \).

- If \( n = 0 \), we have \( \mathcal{A}(n, \{p\}) = 1 \) and \( T_p(n, \{p\}) = 0 \).
- If \( n \geq 1 \), we get \( \mathcal{A}(n, \{p\}) = q\mathcal{A}(n-1, \{S(p)\}) \) and \( T_p(n, \{p\}) = qT_S(p)(n-1, \{S(p)\}) \).

**Proof.** We focus on the avoiding function \( \mathcal{A} \); the proof of the second part is similar. Consider the aforementioned \( q \)-to-1 map from words to adjacency signatures, formed by grouping words into orbits of size \( q \). The number of orbits whose words avoid \( p \) is exactly the number of adjacency signatures that avoid \( S(p) \). Any word of length \( n - 1 \) is a valid adjacency signature, so the latter quantity is simply \( \mathcal{A}(n - 1, \{S(p)\}) \). Since each orbit has \( q \) words, there are \( q\mathcal{A}(n - 1, \{S(p)\}) \) words that avoid \( p \). \( \square \)

Due to this theorem, we develop the following governing principle.

**Main Principle** The adjacency signature map \( S(p) \) develops a bijection between results in the classic Penney’s game and the \( \mathbb{Z}/q\mathbb{Z} \)-Penney’s game.

This principle drives the simple proofs of all of the following \( \mathbb{Z}/q\mathbb{Z} \)-based generalizations of classic word avoidance results.

**Corollary 8.3.** (Generating functions) For any pattern \( p \), we have the relations

\[
\mathcal{G}(z, \{p\}) = 1 + qz\mathcal{G}(z, \{S(p)\}), \quad G_p(z, \{p\}) = qzG_{S(p)}(z, \{S(p)\}).
\]

**Corollary 8.4.** (Correlation) Let \( p_1 \) be a pattern with length \( \ell \), and \( p_2 \) be another pattern. For \( i \leq \ell - 2 \), the entry \( C_i(p_1, p_2) \) is exactly \( C_i(S(p_1), S(p_2)) \). In addition, we have \( C_{\ell-1}(p_1, p_2) = 1 \).
Corollary 8.5. (CLN & Wait time) The Conway leading number between a pattern \( p \) of length \( \ell \) and another pattern \( p' \) is \( pLp' = 1 + qS(p)LS(p') \). The expected wait time of \( p \) is exactly \( 1 + qS(p)LS(p) \).

Corollary 8.6. (Winning strategy) If Alice picks a pattern \( p_1 \) such that \( S(p_1) = s_1(1)s_1(2) \ldots s_1(\ell - 1) \), then Bob’s best strategy is to pick a pattern \( p_2 \) whose adjacency signature is of the form \( S(p_2) = s' s_1(1)s_1(2) \ldots s_1(\ell - 2) \); namely, Bob picks a \( p' \) for which \( S(p'(2, \ell)) = S(p(1, \ell - 1)) \). This is a winning strategy; in particular, the \( \mathbb{Z}/q\mathbb{Z} \)-Penney’s game is still non-transitive.

8.2. Symmetric Group

In this section, we look at the \( S_q \)-Penney’s game, where each element permutes letters.

8.2.1. Lower Bound on the CLN. First, we strengthen the lower bound of a pattern’s CLN, derived in Sect. 4, from within the interval \([q^{\ell - 1} + 1, q^{\ell - 1} + q - 1]\) to an exact bound.

Proposition 8.7. Fixing the group \( G = S_q \) to form our group action, the least possible CLN for a pattern \( p \) of length \( \ell \) is exactly \( q^{\ell - 1} + q - 1 \).

Proof. We showed in Sect. 4 that the minimum is between \( q^{\ell - 1} + 1 \) and \( q^{\ell - 1} + q - 1 \), so it suffices to show \( pLp \geq q^{\ell - 1} + q - 1 \) for all \( p \). Since every letter of \( A \) is within a single orbit, the last entry \( C_{\ell - 1} \) of the autocorrelation vector must be nonzero, so it is equal to \( |G_{p(1)}|/|\text{Stab}_G(p)| = (q - 1)!/|\text{Stab}_G(p)| \), which is either equal to 1 or at least \( q - 1 \).

Suppose \( C_{\ell - 1} = 1 \), meaning that the stabilizer of the last letter of \( p \) has the same order as the stabilizer of \( p \). This can only mean that \( p = aa \ldots a \) for some letter \( a \in A \), which has CLN \( q^{\ell - 1} + \cdots + q + 1 \geq q^{\ell - 1} + q - 1 \) for \( q \geq 2 \). On the other hand, if \( C_{\ell - 1} \neq 1 \), then \( C_{\ell - 1} \) is at least \( q - 1 \) and the CLN is at least \( q^{\ell - 1} + q - 1 \).

To show achievability, note the pattern \( p = aa \ldots ab \) works. The bound is therefore sharp, and the claim follows.

We also partially characterize which patterns achieve this lower bound.

Corollary 8.8. For \( q \geq 4 \), the lower bound is only achieved by patterns with two distinct letters.

Proof. Patterns with exactly one distinct letter clearly don’t work. Now suppose that \( p \) contains at least three distinct letters. Then \( |\text{Stab}_G(p)| \leq (q - 3)! \) and \( C_{\ell - 1} \geq (q - 1)(q - 2) > (q - 1) \), implying \( pLp > q^{\ell - 1} + q - 1 \), whence the lower bound is only achieved by words with two distinct letters.

Remark. The case \( q = 3 \) is special; there are patterns with three distinct letters that achieve the minimal CLN. For instance, with \( (q, \ell) = (3, 4) \), the following patterns have a CLN of \( 3^{4 - 1} + 3 - 1 = 29 \): \( aaab, aaba, aabc, abaa, abbb, abcc \).
8.2.2. Odds. Using our generalized Conway’s formula, we generate some interesting results in the $S_q$ game. In Fig. 1 we let $(q, \ell) = (4, 4)$ and show every pattern $p$ with its best beater $p'$, denoted as $p \rightarrow p'$. We label each arrow with Bob’s winning odds. The data for this graph was generated with a program, found at the repository https://github.com/seanjli/penneys-game-patterns.

Note that for this choice of $q$ and $\ell$, the game is non-transitive. Namely, in Fig. 1, we have a non-transitive cycle of length 5: $aabc \xrightarrow{7:5} abbc \xrightarrow{2:1} abcc \xrightarrow{4:3} abac \xrightarrow{3:2} abc \xrightarrow{9:5} aabc$. Unlike the original Penney’s game, in the game with patterns, Alice may sometimes win. Suppose Alice picks the pattern $p_1$ and Bob picks $p_2$.

**Proposition 8.9.** For $\ell < (q + 2)/2$, the pattern $p_1 = a_1a_2\ldots a_\ell$ (i.e. a pattern with $\ell$ different letters) has better odds against any other pattern of the same length.

**Proof.** Recall that the odds that Bob wins are exactly

$$\frac{|G|_1 Lp_1}{|G|_2 Lp_2} - \frac{|G|_2 Lp_2}{|G|_1 Lp_1}.$$
We replace $|G|_a C_i(p_a, p_b) = |G| C_i^*(p_a, p_b)$ for sake of brevity. We scale the numerator and denominator by $|G|$: for example,

$$
\frac{|G|}{r_1} p_1 L p_1 = \sum_{i=0}^{\ell-1} \frac{|G|}{r_1} C_i(p_1, p_1) q^{\ell-1-i} = \sum_{i=0}^{\ell-1} |G| C_i^*(p_1, p_1) q^{\ell-1-i}
$$

and we know each coefficient $|G| C_i^*(p_1, p_1)$ is either 0 or $|G|_{p_1(i+1, \ell-i)}$. A similar simplification happens for each of the other CLNs.

We claim if Alice chooses $p_1 = a_1 a_2 \ldots a_\ell$, the odds that Bob wins are always less than 1.

First note $|G| C_i^*(p_1, p_1) = (q - \ell + i)!$, yielding

$$
\frac{|G|}{r_1} p_1 L p_1 = (q - \ell)! q^{\ell-1} + (q - \ell + 1)! q^{\ell-2} + \cdots + (q - 1)!
$$

Since $\frac{|G|}{r_1} p_1 L p_2 \geq 0$, the numerator is at most the right-hand quantity above.

We now focus on the denominator. To begin, note that Bob’s pattern has at most $\ell - 1$ distinct letters: if all letters are different, then his pattern is equivalent to Alice’s pattern which is prohibited. Thus $|G| C_0^*(p_2, p_2) = |\text{Stab}_G(p_2)| \geq (q - \ell + 1)!$. In addition, for any pattern $p_2$ we have $|G| C_{\ell-1}^*(p_2, p_2) = (q - 1)!$, so we obtain the lower bound:

$$
\frac{|G|}{r_2} p_2 L p_2 \geq |G| C_0^*(p_2, p_2) q^{\ell-1} + |G| C_{\ell-1}^*(p_2, p_2) \geq (q - \ell + 1)! q^{\ell-1} + (q - 1)!
$$

Finally, note that $|G| C_i^*(p_2, p_1)$ is equal to

- 0, if $i = 0$; and
- either 0 or $|G|_{p_1(i+1, \ell)} = (q - \ell + i)!$.

This gives us an upper bound of $(q - \ell + i)!$ on each coefficient $|G| C_i^*(p_2, p_1)$, so

$$
\frac{|G|}{r_2} p_2 L p_1 \leq \sum_{i=1}^{\ell-1} (q - \ell + i)! q^{\ell-1-i} = (q - \ell + 1)! q^{\ell-2} + \cdots + (q - 1)!
$$

Thus, the denominator $\frac{|G|}{r_2} p_2 L p_2 - \frac{|G|}{r_2} p_2 L p_1$ is at least

$$(q - \ell + 1)! q^{\ell-1} - \sum_{i=1}^{\ell-2} (q - \ell + i)! q^{\ell-1-i}.$$
\[
\frac{\sum_{r_1=1}^{\left|G\right|} p_1 \mathcal{L} p_1 - \sum_{r_2=1}^{\left|G\right|} p_2 \mathcal{L} p_2}{\sum_{r_2=1}^{\left|G\right|} p_2 \mathcal{L} p_2 - \sum_{r_1=1}^{\left|G\right|} p_1 \mathcal{L} p_1}
\leq \frac{(q - \ell)!q^{\ell-1} + \sum_{i=1}^{\ell-2}(q - \ell + i)!q^{\ell-1-i}}{(q - \ell + 1)!q^{\ell-1} - \sum_{i=1}^{\ell-2}(q - \ell + i)!q^{\ell-1-i}}
\]

Using the inequality \((q - \ell + i)!q^{\ell-1-i} < (q - \ell + 1)!q^{\ell-2}\), we may bound the right-hand side from above by decreasing the denominator. Namely, the RHS is at most
\[
\frac{(q - \ell)!q^{\ell-1} + (q - \ell + 1)!q^{\ell-1}}{(q - \ell + 1)!q^{\ell-1} - \sum_{i=1}^{\ell-2}(q - \ell + i)!q^{\ell-1-i}} - 1.
\]
The right-hand side is less than 1 for \(\ell < (q + 2)/2\). Thus, Bob’s odds of winning are always less than 1, and he has a disadvantage. □

In particular, this means that Alice has a unique winning strategy—if she doesn’t pick the winning pattern, then Bob can and he will win.

Acknowledgements

We are grateful to the MIT PRIMES-USA program for giving us the opportunity to conduct this research. On behalf of all authors, the corresponding author states that there is no conflict of interest.

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

[1] Isha Agarwal, Matvey Borodin, Aidan Duncan, Kaylee Ji, Tanya Khovanova, Shane Lee, Boyan Litchev, Anshul Rastogi, Garima Rastogi, and Andrew Zhao. From Unequal Chance to a Coin Game Dance: Variants of Penney’s Game, preprint, arXiv:2006.13002.

[2] Elwyn R. Berlekamp, John H. Conway and Richard K. Guy, Winning Ways for your Mathematical Plays, 2nd Edition, Volume 4, AK Peters, 2004, 885.

[3] Stanley Collings. Coin Sequence Probabilities and Paradoxes, Inst. Math A, 18 (1982), 227–232.

[4] Daniel Felix, Optimal Penney Ante Strategy via Correlation Polynomial Identities, Electr. J. Comb., 13 (2006).

[5] Martin Gardner, Mathematical Games, Sci. Am., 231 (1974), no. 4, 120–125.

[6] Martin Gardner, Time Travel and Other Mathematical Bewilderments, W. H. Freeman, 1988.

[7] L.J. Guibas and A.M. Odlyzko. String Overlaps, Pattern Matching, and Non-transitive Games, J. Comb. Theory A, 30 (1981), no. 2, 183–208.
[8] Steve Humble and Yutaka Nishiyama. Humble-Nishiyama Randomness Game — A New Variation on Penney’s Coin Game, *IMA Mathematics Today*. **46** (2010), no. 4, 194–195.

[9] Walter Penney, Problem 95. Penney-Ante, *J. Recreat. Math.*, **2** (1969), 241.

[10] Robert W. Vallin. Sequence game on a roulette wheel, published in *The Mathematics of Very Entertaining Subjects: Research in Recreational Math*, Volume 2, University Press, Princeton, (2017), 286–298.

Sean Li and Tanya Khovanova
Massachusetts Institute of Technology
Cambridge MA
USA
e-mail: seanjli@mit.edu

Tanya Khovanova
e-mail: tanyakh@yahoo.com

Communicated by Ilse Fischer
Received: 8 March 2021.
Accepted: 20 December 2021.