

Research Article

Repetition in Permutation Wordle

Aurora Hiveley*

Department of Mathematics, Rutgers University, Piscataway, New Jersey, USA

(Received: 28 April 2026. Received in revised form: 9 June 2026. Accepted: 9 June 2026. Published online: 11 June 2026.)

© 2026 the author. This is an open-access article under the CC BY (International 4.0) license (www.creativecommons.org/licenses/by/4.0/).

Abstract

In a game of permutation wordle, a player attempts to guess a secret permutation in the fewest number of guesses possible. Previously, Kutin and Smithline [*arXiv:2408.00903 [math.CO]*, (2024)] introduced this game and proposed a strategy called cyclic shift, which they conjecture performs optimally. The present paper continues the investigation of this conjecture and proves it in the sense of “no waste” by considering how information is obtained and, at times, repeated during a game of permutation wordle using an arbitrary strategy. This analysis includes several algorithms to construct a secret permutation which prompts inefficient repetition according to the player’s strategy, as well as proofs of their efficacy.

Keywords: permutation wordle; derangement; cyclic permutation; algorithm.

2020 Mathematics Subject Classification: 05A05, 68W32,

1. Introduction

Consider a variation of the New York Times game, Wordle, where instead of attempting to guess a 5-letter word, the player attempts to guess a permutation on $[n]$. On each turn, the player guesses a permutation and is told which positions of their guess are correct. The process repeats until the player correctly guesses the secret permutation, hoping to do so in the fewest number of guesses possible. This game was first introduced by Kutin and Smithline in 2024, and they proposed a strategy called *cyclic shift* which they conjecture has optimal performance in a game of permutation wordle. Cyclic shift, which we abbreviate CS, generates the k -th guess γ_k in a game of permutation wordle by locking in all correct entries from γ_{k-1} and shifting each incorrect entry one position to the right, skipping over the locked in entries in the process.

The study of secret code guessing games isn’t just trendy in popular culture, but also mathematically rich. Kutin and Smithline were originally inspired by the click-and-drag matching interface on the quiz game website, Sporcle.com, and the question of how well a user could expect to do if they guessed randomly. Li and Zhu [4] studied a similar modification of the classic board game, Mastermind, using Wordle-like feedback. Their game differs from ours in the sense that the secret code is a string in $[c]^n$ where c is the number of colors and n is the length of the code, while ours must be a permutation. Stanley [5] has also considered a permutation guessing game inspired by the Magic Cauldron event in the mobile game Royal Match. Many versions of this game exist in the world both for the curious gamer and the curious mathematician.

Previously, Kutin and Smithline studied the connection between the number of guesses that a game using CS will last and the number of excedances of the secret permutation, which in turn is related to the Eulerian numbers. From there, Hiveley [1] proved optimality of cyclic shift for games ending in exactly three guesses. In this work, a strategy is formalized as a list $S = [s_1, s_2, \dots, s_n]$ where s_i is a permutation of length i for $1 \leq i \leq n$, and the next guess γ_k is generated by considering the set \mathcal{I}_{k-1} of incorrect entries from guess $k-1$ and permuting them according to the permutation $s_{|\mathcal{I}_{k-1}|}$. Note that in this language, CS = $[[1], [2, 1], [2, 3, 1], \dots, [2, 3, \dots, n, 1]]$, and we will often use the list-indexing notation $S[k]$ instead of s_k to refer to the k -th component of a strategy.

In constructing a strategy for permutation wordle, we make a few assumptions about what an efficient game looks like. In a game consisting of guesses $\gamma_1, \gamma_2, \dots, \gamma_r$, we assume for simplicity that γ_1 is always the trivial permutation $[1, 2, \dots, n-1, n]$. We also assume that a player wants to maximize the amount of new information learned on each turn, and a guess which fails to do so is inherently inefficient. For example, say that $1 \in \mathcal{I}_1$, meaning that the entry 1 is incorrect in γ_1 and therefore *not* in the first position in the secret permutation. If the player were to guess the entry 1 in position one later on in the game, say on guess number t , then γ_t cannot possibly be the secret permutation, so it is intrinsically an inefficient guess. The player should guess the entry 1 in some other un-guessed position in an effort to gain new information. In this sense, we require that the strategy components used to permute incorrect entries between guesses are *derangements*,

*E-mail address: aurora.hiveley@rutgers.edu

i.e. lacking fixed points. This guarantees that each incorrect entry is moved to a new spot compared to the previous guess, but how can we be sure that the entry’s new position is *brand new* in the context of the entire game? Our work in this paper will study just that: characterizing repetition in the course of a game, and drawing conclusions about Kutin and Smithline’s conjectured optimality of CS. We will begin with a discussion of repetition within a game of permutation wordle in Section 2, including several algorithms to produce secret permutations that spur such repetition. We will then conclude by examining repetition for games using the strategy CSL, which was previously defined and studied in [1].

2. Repetition in Inductive Strategies

In [1], we previously introduced the idea of an inductively constructed strategy, which has some arbitrary permutation for the n -th component $S[n]$, but the first $n - 1$ components are all rightward cyclic shifting. The motivation for this framework came from an attempt to prove the optimality of CS by induction. However, in [1] it was shown that for any derangement acting as $S[n]$, the average number of guesses needed before \mathcal{J}_k (the set of *correct* entries on guess k) is nonempty and induction takes over is actually constant for a fixed n . This means that in order to prove Kutin and Smithline’s conjecture about the optimality of CS, we will have to pursue an approach that is more complex than simply considering inductively constructed strategies at the “highest level.” Instead, we turn our attention to the relationship between a strategy’s components. First, consider an example game using the strategy $S = [[1], [2, 1], [2, 3, 1], [2, 1, 4, 3], [3, 4, 5, 2, 1]]$ with the secret permutation $p = [4, 1, 5, 2, 3]$. The game will proceed as follows:

$$\begin{array}{ll} \gamma_1 = [1, 2, 3, 4, 5] & \mathcal{I}_1 = \{1, 2, 3, 4, 5\} \\ \gamma_2 = [5, 4, 1, 2, 3] & \mathcal{I}_2 = \{1, 2, 3\} \\ \gamma_3 = [1, 5, 4, 2, 3] & \mathcal{I}_3 = \{1, 2, 3\} \\ \gamma_4 = [4, 1, 5, 2, 3] & \mathcal{I}_4 = \emptyset \end{array}$$

Notice that in this game, the entry 1 is guessed incorrectly in position one in both γ_1 and γ_3 . By the assumptions established in Section 1, this game is suboptimal in the sense that γ_3 . There is no way that γ_3 could have been the secret permutation since we already knew that $p[1] \neq 1$, so the game could not have ended after three guesses, and we wasted an opportunity to learn new information about which digit is in position one and which position the digit 1 occupies in p . So, this strategy can be improved because it fails to perform optimally when the secret permutation is $p = [4, 1, 5, 2, 3]$.

Experimental calculations using Maple indicate that this repetition of incorrect information actually happens for every possible strategy in our construction for at least one possible secret permutation. All strategies, that is, except for CS. We begin by presenting the following lemma. Observe that while we consider only rightward cyclic shifting in the proof, leftward cyclic shifting is handled analogously.

Lemma 2.1. *CS never duplicates incorrect information.*

Proof. This follows directly from Definition 3.1 and Proposition 3.3 in [3]. Assume for the sake of contradiction that CS produces two different guesses in which an incorrect entry is guessed in the same position. Let these two guesses be γ_s and γ_t , and without loss of generality let $s < t$. Let $\gamma_s[i] = \gamma_t[i] = j$ where $i \in \mathcal{I}_s, \mathcal{I}_t$. Since $i \in \mathcal{I}_s$, entry j is shifted once to the right in γ_{s+1} . For $\gamma_t[i] = j$ later on, entry j must have skipped over every open position in the permutation to return to position i . However, this means that there is no open position where entry j is correct, which contradicts either the secret permutation being a permutation at all or the correctness of a previously considered correct position. \square

As mentioned in Section 1, we assume that a strategy *must* be inefficient if incorrect information is duplicated. In other words, if $\gamma_i[m] = \gamma_j[m]$ for $i < j$ where $m \in \mathcal{I}_i$, then the strategy may be improved by simply changing $\gamma_j[m]$. In this scenario, γ_j is guaranteed to be incorrect since position m is incorrect. In this sense, CS certainly satisfies the barest minimum criteria for an effective strategy. And, in fact, any other strategy fails to meet this criteria.

Theorem 2.1. *Any strategy whose components are derangements and which is not cyclic shifting (either rightward or leftward) repeats information for at least one permutation.*

To prove this claim, we present an algorithm which, given the final component $S[n]$ of an *inductive* strategy, constructs a permutation which, when acting as the secret permutation in a game of permutation wordle, causes the strategy S to duplicate incorrect information. We will call these permutations *offending permutations*, denoted ω . We will then extend our algorithm in Section 3 to include any strategy whose components are derangements, thereby proving the full claim.

Displacement

Recall that all games of permutation wordle begin with the trivial guess, $[1, 2, \dots, n]$. In order for the strategy component $S[n]$ to be utilized in the game (without cyclic shifting induction immediately taking over and producing no repeated incorrect entries per Lemma 2.1), we need $\mathcal{J}_1 = \emptyset$, so the secret permutation must be a derangement of $[n]$. Since $\mathcal{J}_1 = \emptyset$, when $S[n]$ generates the next guess we get that $\gamma_2 = S[n]^{-1}$. Since $S[n]$ is a derangement, the inverse is also a derangement, and for the remainder of the algorithm we will use this derangement $S[n]^{-1}$ to construct an offending permutation. The simplest way to force the repetition of bad information in a game of permutation wordle is to impel an entry i back into position i , meaning that for some $t > 1$, we have $\gamma_1[i] = \gamma_t[i] = i$. To do this, we introduce the notion of *displacement*:

Definition 2.1. *The displacement vector $d(\pi)$ of a permutation $\pi \in S_n$ is the list of length n such that, for all $1 \leq i \leq n$, $d(\pi)[i] := \pi[i] - i \pmod n$*

In other words, the list entry $d(\pi)[i]$ is the number of times that the entry i must be shifted rightward before entry i is in position i . For example, the rightward cyclic shifting component $S[n] = [2, 3, \dots, n, 1]$ has a displacement vector of $[1, 1, \dots, 1]$. Similarly, the left shifting component $S[n] = [n, 1, 2, \dots, n-1]$ has a displacement vector $[n-1, n-1, \dots, n-1]$. For a more interesting example, the permutation $[5, 3, 2, 6, 1, 4]$ has a displacement vector $[4, 1, 5, 2, 2, 4]$.

To construct an offending permutation, we will study the displacement vector of $\gamma_2 = S[n]^{-1}$, as this will indicate how many times each entry in γ_2 must be right-shifted before returning to their position in γ_1 . Since $S[n]^{-1}$ is a derangement, the displacement vector must consist of all nonzero numbers as there are no fixed points. From here, we will divide our study into two cases: displacement vectors which contain at least one element e such that $2 \leq e \leq n-2$, and displacement vectors consisting of only $e \in \{1, n-1\}$. We begin by presenting an outline of our construction for a strategy of length 4 to motivate each step of the algorithm before generalizing to larger strategy lengths.

Vectors Containing 2

In the case of $n = 4$, note that $2 = n - 2$, so any permutation whose inverse has at least one displacement vector entry e such that $2 \leq e \leq n - 2$ actually has an entry which is precisely equal to 2. Let position i in γ_2 be an inverse entry with displacement equal to 2. Say that $\gamma_2[i+1] \in \mathcal{J}_2$. (Note, of course, that all of this addition is modulo n , so entries may loop back around to the front of the permutation.) To produce γ_3 , we right shift all incorrect entries, so the entry $\gamma_2[i]$ is shifted past the correct entry in position $i+1$ and becomes $\gamma_3[i+2]$. But since $\gamma_2[i]$ had displacement of 2, $\gamma_2[i] = i+2$, and hence $\gamma_2[i] = \gamma_1[i+2]$ as our first guess is always trivial. Here's the rub, since $\gamma_3[i+2] = \gamma_2[i] = \gamma_1[i+2]$, and $i+2 \notin \mathcal{J}_1$ since $\mathcal{J}_1 = \emptyset$. Hence the same entry was guessed incorrectly in position $i+2$ more than once!

So if $\gamma_2[i+1] \in \mathcal{J}_2$, then we know that an entry will repeat on the third guess, γ_3 . It suffices to construct the rest of the offending permutation ω in a way that will result in a legal game. Certainly the repeated entries must be right shifted at least once more in order to guess the secret permutation and end the game, so for the sake of simplicity say that $\mathcal{J}_2 = \{i+1\}$ and thus all other entries are right shifted to form γ_4 . It remains, then, to verify that each of these entries is in a new position which they have not occupied before so that γ_4 can legally end the game. The details of this construction are expounded in the upcoming proof of Proposition 2.1.

As an example, let $S[4] = [2, 4, 1, 3]$. Then $S[4]^{-1} = \gamma_2 = [3, 1, 4, 2]$, and $d(S[4]^{-1}) = [2, 3, 1, 2]$. Then let $i = 1$ since the element 3 in position one of $S[4]^{-1}$ has displacement of 2. Next, we let $i+1 = 2 \in \mathcal{J}_2$, so the entry 1 in position two is a correct entry. Then $\gamma_3 = [2, 1, 3, 4]$ since we right shift all entries except for the correct entry, 1. Note that entries 3 and 4 are guessed in positions three and four, respectively, but they were already guessed in those positions in γ_1 , so we have duplicated bad information. Lastly, we let $\mathcal{J}_3 = \mathcal{J}_2 = \{2\}$ and right shift all other entries one final time to obtain $\gamma_4 = [4, 1, 2, 3]$. Note that if the secret permutation is $[4, 1, 2, 3]$, then this is a legal game of permutation wordle, and using the inductive strategy with $S[4] = [2, 4, 1, 3]$ caused the duplication of incorrect information between γ_1 and γ_3 .

Note that an offending permutation need not repeat incorrect information in γ_3 , specifically. Especially for larger n which may lead to games lasting a larger number of turns, there are many offending permutations whose first duplicated incorrect entry occurs in γ_4 or later. However, we only need to produce *one* offending permutation in order to prove our claim, and the most straight-forward way to do this is by forcing repetition as soon as possible. We formalize this constructive algorithm as follows, noting that addition of indices happens mod n , so $i+1 = (i \pmod n) + 1$. This algorithm is also implemented in the second of two Maple packages linked in the Appendix as the procedure `construct(p, right)`.

Observe that in Step (1) we required that $(i-1) \notin K$ before adding i to K . Without this condition, if $D = [2, 2, \dots, 2]$, then $K = \{1, \dots, n\}$ and so $\omega = \gamma_2$, which does not produce any repetition as the game ends after two guesses. Then, to ensure that another guess is generated, we must have at least three entries not in K , hence the other condition in Step (1). We conclude our analysis of displacement vectors containing 2 by proving the aforementioned algorithm's efficacy:

Algorithm 1 Offending Permutation Constructor For $2 \in D$

0. Given the n -th strategy component $S[n]$ of an inductive strategy, consider $D = d(S[n]^{-1})$
 1. Initialize $K = \emptyset$. For each $i \in [n]$, if $D[i] = 2$ and $i - 1 \notin K$, add i to K . Stop when $|K| = n - 3$, even if there are remaining 2's to consider.
Special case: If $D[1] = D[n] = 2$ and $D[j] \neq 2$ for all $j \in \{2, \dots, n - 1\}$, then let $i = n$ so that $K = \{n\}$.
 2. For each $i \in K$, let $\omega[i + 1] = S[n]^{-1}[i + 1]$.
 3. For each $j \in \{1, \dots, n\} \setminus K$, shift $S[n]^{-1}[j]$ rightward twice, skipping over the entries locked in during Step (2). In other words, if $L = \{1, \dots, n\} \setminus K$ where $|L| = m$, then $\omega[L[j] + 2] = S[n]^{-1}[L[j]]$ for $1 \leq j \leq m$.
-

Proposition 2.1. *If $S[n]$ is a derangement of length n such that $2 \in D = d(S[n]^{-1})$, then Algorithm 1 produces an offending permutation ω for the inductive strategy with $S[n]$.*

Proof. Let $S[n]^{-1}$ have a displacement vector with at least one 2. Let ω be as defined in Algorithm 1. In a game of permutation wordle where $\gamma_2 = S[n]^{-1}$, say that $\mathcal{J}_2 = K + 1$, i.e., the set K where each element is incremented by 1 (modulo n .) Then $\mathcal{J}_2 \neq \emptyset$ since there is at least one 2 in D , so γ_3 is generated by the strategy component $S[n - |K|]$, which is right-shifting since S is an inductive strategy. Then all incorrect entries shift once rightward in γ_3 . For each index $i \in K$, $i + 1 \in \mathcal{J}_2$, so the entry $\gamma_2[i]$ is right-shifted into position $i + 2$ in γ_3 . Since $D[i] = 2$, we have that $\gamma_2[i] = \gamma_3[i + 2] = i + 2$, and so the entry $i + 2$ is guessed in the same position as in γ_1 , thereby repeating incorrect information since $\mathcal{I}_3 \subset \mathcal{I}_1$.

According to the construction of ω , $\mathcal{J}_3 = \mathcal{J}_2$, so every index not in $\mathcal{J}_2 = K + 1$ will be right-shifted again to form $\gamma_4 = \omega$. Then it suffices to show that γ_4 is a legal end state for the game, i.e. that there are no additional repeated incorrect entries on guess four. Each $i \in K$ ends in position $i + 3$ or $i + 4$ after skipping over one locked entry to go from γ_2 to γ_3 , and skipping over at most one more entry to go from γ_3 to γ_4 . Each element in a position $i \in K$ has only been guessed in position $i + 2$ (on γ_1 and γ_3) or in position i on γ_2 , so $i + 3$ and $i + 4$ must be new positions as long as $i \geq 4$. Note that the $i + 4$ state is not obtainable for a permutation of length 4 since $|\mathcal{J}_2| \leq n - 3 = 1$, so the entry i cannot skip over more than one correct entry.

For any $j \notin K \cup \mathcal{J}_2$, we right shift twice. Then the end position of $\gamma_2[j]$ is either $j + 2$ (if no correct entries are skipped over), $j + 3$ (if one correct entry is skipped over) or $j + 4$ (if two correct entries are skipped over). Notice that since $j \notin K$, the displacement of $\gamma_2[j]$ is not equal to 2, so if $\gamma_2[j]$ ends in position $j + 2$, this is legal. If $\gamma_2[j]$ ends in position $j + 3$, then it must have skipped over a correct entry to go from γ_2 to γ_3 or to go from γ_3 to γ_4 . If $\gamma_2[j]$ skipped over a correct entry after γ_2 , then $j + 1 \in \mathcal{J}_2$, but this means that $j \in K$, which is a contradiction. If $\gamma_2[j]$ ends in position $j + 4$, it must have skipped over one correct entry after each of γ_2 and γ_3 since there are no adjacent entries in \mathcal{J}_2 by construction. But once again we have a contradiction as this would mean that $j + 1 \in \mathcal{J}_2$. So the only problematic possibility is if $\gamma_2[j]$ ends in position $j + 3$ after skipping over a correct entry on γ_3 . However, $\gamma_2[j]$ cannot have previously been guessed in position $j + 3$ since position $j + 2 \in K$ and hence $j + 1$ was an entry with displacement of 2, meaning $\gamma_1[j + 1] = \gamma_3[j + 1] = j + 1$. The only way that $\gamma_2[j]$ could have previously been guessed in position $j + 3$ is if it happened in γ_2 , but we know that $\gamma_2[j]$ was in position j on γ_2 by definition. Hence, $\gamma_2[j]$ ends in a legal position at the end of γ_4 . □

Vectors Not Containing 2

For a strategy of length 4, we note that the displacement vector of $S[n]^{-1}$ can only have 1, 2, and 3 as entries, so all strategies can either be handled by the process described above or by the approach which will be outlined in the next subsection. However, the displacement vector of $S[n]^{-1}$ for a general strategy component of length n will have entries in $\{1, 2, \dots, n - 1\}$, meaning that it is possible for the displacement vector used to construct ω to not have any 2's. The case where all entries are either 1 or $n - 1$ is detailed in the next section, so for now we turn our attention to displacement vectors with some entry $e \in \{3, \dots, n - 2\}$. We further generalize our approach from the previous section as follows:

As an example, consider $S[6] = [3, 4, 6, 5, 1, 2]$. In such a game, $S[6]^{-1} = \gamma_2 = [5, 6, 1, 2, 4, 3]$, and thus $D = [4, 4, 4, 4, 5, 3]$. Then $\mu = 3$ and $\iota = 6$, so $K = \{1, 2\}$. By our construction, $\mathcal{J}_2 = \{1, 2\}$, so $\gamma_3 = [5, 6, 3, 1, 2, 4]$ and we observe that the entry 3 is guessed in the same incorrect position as it was in γ_1 . Allowing $\mathcal{J}_2 = \mathcal{J}_3$ once again, we obtain $\gamma_4 = [5, 6, 4, 3, 1, 2]$, and if γ_4 is the secret permutation, our game ends here.

Proposition 2.2. *If $S[n]$ is a derangement of length n such that $2 \notin D = d(S[n]^{-1})$, then Algorithm 2 produces an offending permutation ω for the inductive strategy with $S[n]$.*

Proof. Let $S[n]^{-1}$ have a displacement vector containing at least one entry in $\{3, \dots, n - 2\}$ and no 2's. Let μ be the minimum non-1 entry in D , and let ω be the permutation constructed as described in the algorithm above. Once again, it suffices to

Algorithm 2 Offending Permutation Constructor For $2 \notin D$

- (0) Given the n -th strategy component $S[n]$ of an inductive strategy, consider $D = d(S[n]^{-1})$. This time, let $\mu = \min_{e \in D \setminus \{1\}}$.
1. Let $\iota = \min\{x \mid D[x] = \mu\}$, and let $K = \{\iota + k \mid 1 \leq k \leq \mu - 1\}$.
 2. For each $i \in K$, let $\omega[i] = S[n]^{-1}[i]$.
 3. For each $j \in [n] \setminus K$, shift $S[n]^{-1}[j]$ rightward twice, skipping over the entries locked in during Step (2). In other words, if $L = [n] \setminus K$ where $|L| = m$, then $\omega[L[j + 2]] = S[n]^{-1}[L[j]]$ for $1 \leq j \leq m$.

Observe that in this case, the entries ι and $\iota - 1$ are each shifted $\mu + 1$ spaces rightward, while all other entries are shifted twice rightward without skipping.

show that an entry is repeated (so the permutation is an offender) and that the final game state $\gamma_4 = \omega$ is legal.

Let ι be a position between 1 and n such that $D[\iota] = \mu$. Since $\mathcal{J}_2 = K$ consists of the $\mu - 1$ entries to the right of position ι (and $\mu \geq 3$ guarantees that $\mathcal{J}_2 \neq \emptyset$) then γ_3 is generated by the right shifting component $S[n - |K|]$ since S is an inductive strategy. Then entry $\gamma_2[\iota] = I$ is shifted μ positions to the right, and since $D[\iota] = \mu$ we have that $\gamma_3[I] = \gamma_1[I] = I$ and thus incorrect information is repeated during the course of a game. Then, to form γ_4 , the entry I is shifted one more position to the right. Recall that $\mathcal{J}_2 = K$ so $|\mathcal{J}_2| = \mu - 1$, and by our construction we have that $\mathcal{J}_3 = \mathcal{J}_2$ and therefore $|\mathcal{I}_3| = n - (\mu - 1)$. Since $3 \leq \mu \leq n - 2$, we then have that $|\mathcal{I}_3| \geq 3$, so the entry I is shifted into a new position in γ_4 as it takes on a third position from $\mathcal{I}_2 = \mathcal{I}_3$.

It remains to show that no other entry from a position $j \in \mathcal{I}_2$ ends at a repeated index. Since \mathcal{J}_2 consists of the $\mu - 1$ entries immediately to the right of the position ι , we have that the entries of \mathcal{I}_2 are the $n - \mu$ entries immediately before the position ι . Observe that any entry j two or more positions to the left of ι will be right shifted twice without skipping over any correct entries. For example, if $j = \iota - 2$ then the entry in position j is right shifted twice into position ι , which is the position immediately left of all entries in K . Since no entry in γ_2 had a displacement of 2, it is impossible for any of these entries to end in their same position in γ_1 , and because $|\mathcal{I}_2| \geq 3$ each of the positions occupied by these entries in γ_3 and γ_4 must be new, hence there is no repetition and the ending state is legal. The only remaining entry to consider is the entry in position $\iota - 1$, since this entry will be right shifted into position ι on γ_3 and then into position $\iota + \mu$ on γ_4 . However, we observe that $\gamma_2[\iota - 1]$ cannot have been guessed in position $\iota + \mu$ previously since the entry $I = \gamma_2[\iota]$ was guessed there in γ_1 and γ_3 , and we know that $\gamma_2[\iota - 1]$ was guessed in position $\iota - 1$ in γ_2 , by definition. And of course, $\iota - 1 \neq \iota + \mu$ since $\mu \leq n - 2$, so every entry ends in a new position, so $\gamma_4 = \omega$ is a legal end state. \square

Displacement Vectors of Derangements

Since the sum of all entries in the displacement vector must be a multiple of n , a displacement vector consisting of only 1's and $(n - 1)$'s, must be either $[1, \dots, 1]$, $[n - 1, \dots, n - 1]$, or it must have an equal number of 1's and $(n - 1)$'s. The two cyclic shifting strategy components are discussed in greater detail in the next subsection, so we will focus only on the latter case. If the displacement vector does have an equal number of 1's and $(n - 1)$'s, then the vector must have the form $[1, n - 1, 1, n - 1, \dots, 1, n - 1]$ or vice versa, otherwise two entries have displacements calculated from the same position, which is not possible. Such a permutation is actually a derangement consisting of $\frac{n}{2}$ 2-cycles. For example, the permutation $[6, 3, 2, 5, 4, 1]$ has a displacement vector $[1, 5, 1, 5, 1, 5]$ and consists of the two cycles $(16)(23)(45)$ when written in cycle notation. Observe also that these permutations are their own inverses, so $\gamma_2 = S[n]$. Then to prove Theorem 2.1, it suffices to show that we can still construct a repeating permutation when $S[n]$ is a derangement of this form.

We begin with an example for $n = 4$. If S is a strategy of length 4 with $S[4] = [2, 1, 4, 3]$, then S will repeat incorrect information when the secret permutation is $[3, 4, 1, 2]$. Additionally, such a game can never terminate since $\gamma_1 = [1, 2, 3, 4]$, and $\mathcal{J}_1 = \emptyset$ so $\gamma_2 = [2, 1, 4, 3]$, but then $\mathcal{J}_2 = \emptyset$, so $\gamma_3 = [1, 2, 3, 4] = \gamma_1$, and so on.

$$\begin{array}{ll}
 \gamma_1 = [1, 2, 3, 4] & \mathcal{J}_1 = \emptyset \\
 \gamma_2 = [2, 1, 4, 3] & \mathcal{J}_2 = \emptyset \\
 \gamma_3 = [1, 2, 3, 4] & \mathcal{J}_3 = \emptyset \\
 \gamma_4 = [2, 1, 4, 3] & \mathcal{J}_4 = \emptyset \\
 \vdots & \vdots
 \end{array}$$

Notice that our guessing game enters an infinite loop wherein $S[4]$ repeatedly produces the same two guesses with no correct entries. Since \mathcal{J}_s will always be empty for any $s \geq 1$, we will never be able to use a different strategy component to produce a different guess, i.e., the number of guesses needed to correctly guess $[3, 4, 1, 2]$ is infinite. A similar argument applies to $[2, 3, 1, 5, 6, 4]$ or $[2, 1, 4, 3, 6, 5]$ when used as $S[6]$, and so on. We generalize this observation as follows:

Theorem 2.2. *Let δ be a derangement of length n with cycle type $[t_1, t_2, \dots, t_k]$. Let $\mu_i := |\{j \mid t_j = t_i\}|$, or in other words μ_i is the multiplicity of the cycle length t_i in the cycle type of δ . If $\mu_i \geq 2$ for all $1 \leq i \leq n$, then any strategy with $S[n] = \delta$ will enter an infinite loop for at least one possible secret permutation ω of length n .*

Proof. Let δ be a derangement with with cycle type $[t_1, t_2, \dots, t_k]$ such that $\mu_i \geq 2$ for all $1 \leq i \leq k$. Let $\delta = c_1 c_2 \dots c_k$ be the partition of δ into cycles where the length of c_i is t_i for each i . For a unique cycle length t_i , let $C_i = \{c_i^1, c_i^2, \dots, c_i^{\mu_i}\}$ be the set of cycles in δ that each have length t_i . Then to construct an offending permutation ω , map the elements of c_i^j to the positions occupied by c_i^{j+1} in δ , in other words $\omega[c_i^j[a]] = c_i^{j+1}[a]$ for any $1 \leq a \leq t_i$. Repeat this mapping for all $1 \leq j \leq \mu_i$ (modulo μ_i , of course, so that $c_i^{\mu_i}$'s elements map to the positions occupied by c_i^1) and for all unique cycle lengths t_i . Consider the example below where δ has length 10 and decomposes into two 2-cycles and two 3-cycles:

$$\delta = [2, 3, 1, 5, 4, 7, 8, 6, 10, 9] = (1, 2, 3)(4, 5)(6, 7)(8, 9, 10) \implies \omega = [8, 9, 10, 6, 7, 4, 5, 1, 2, 3]$$

Observe that a permutation wordle player using the inductive strategy such that $S[n] = \delta$ will enter an infinite loop while attempting to guess ω . The player's first guess is always the trivial permutation $[1, 2, \dots, n]$, and since δ was a derangement, each cycle c_i in δ has length at least 2. Since $\mu_i \geq 2$ for all i , no entries of ω are shared with the trivial permutation. Then $\delta = S[n]$ generates the next guess as follows:

$$\begin{aligned} \gamma_1 &= \underbrace{1, 2, \dots, t_1}_{c_1}, \underbrace{t_1 + 1, \dots, t_1 + t_2}_{c_2}, \dots, \underbrace{n - t_k, \dots, n}_{c_k} \\ \gamma_2 &= \underbrace{2, \dots, t_1, 1}_{c_1}, \underbrace{t_1 + 2, \dots, t_1 + t_2, t_1 + 1}_{c_2}, \dots, \underbrace{n - t_k + 1, \dots, n, n - t_k}_{c_k} \end{aligned}$$

Once again, none of the entries will be correct in γ_2 since each entry of c_1 was mapped to a location within the first t_1 locations, and the same for c_2 , and so on. Since $\mathcal{J}_2 = \emptyset$, we generate γ_3 according to δ once again, but by the same logic as before, $\mathcal{J}_3 = \emptyset$. This process will repeat infinitely without ever locking in any correct entries, so we cannot guess the permutation ω in finitely many guesses using $S[n] = \delta$. □

In the above proof, the length of the strategy S need not be n . That is, the derangement δ may not be the longest component of S , and we can still identify a permutation that will cause this infinite looping later on in a game of permutation wordle. This concept will be studied in greater detail and broader generalization in Section 3. But with this in hand, we will revisit our construction of offending permutations. To review, if a displacement vector consists of only 1's and $(n - 1)$'s, then it must pairwise alternate between these entries. This means that the permutation $S[n]$ must be a derangement comprised of all 2-cycles, which in fact must be adjacent transpositions since each entry is either a distance of one leftward or one rightward away from its original position. Therefore $S[n]^{-1} = S[n]$, and if $\mathcal{J}_2 = \emptyset$ then $\gamma_3 = \gamma_1$. In this case, all information is repeated! The game entering an infinite loop surely guarantees the repetition of incorrect information, so we construct our offending permutation in this way. We have two possibilities using the construction described in the proof above:

Algorithm 3 Offending Permutation Constructor For $D \subset \{1, n - 1\}^n$

1. If $S[n] = [2, 1, 4, 3 \dots, n, n - 1]$, then $S[n] = (1, 2)(3, 4) \dots (n - 1, n)$ when decomposed into cycles. Using the construction from above, we have the offending permutation $\omega = [3, 4, \dots, n - 1, n, 1, 2]$
 2. If $S[n] = [n, 3, 2 \dots, n - 1, n - 2, 1]$, then $S[n] = (1, n)(2, 3) \dots (n - 2, n - 1)$ when decomposed into cycles. Using the construction from above, we have $\omega = [3, 4, \dots, n - 1, n, 1, 2]$
-

Observe that in either case, the same offending permutation is constructed! There is one case where $S[n]$ is a derangement which does not fall into this category, and it is the case where $S[4] = [3, 4, 1, 2]$. This is a derangement consisting of the two 2-cycles $(1, 3)(2, 4)$, but the displacement vector takes the form $D = [2, 2, 2, 2]$. If we employ Algorithm 1, then the secret permutation is constructed by setting $K = \{1\}$. Since $S[n]^{-1} = [3, 4, 1, 2]$, we then have that $\omega[1] = 3$, and we right shift each of the other entries twice to obtain $\gamma_4 = \omega = [3, 1, 2, 4]$. However, the entry 4 ends in position four, so this is not a legal end state for the game since that entry cannot be correct. The issue lies in the fact that an entry with displacement of two was right shifted two positions. In our previous proof and algorithm, this was not an issue, so we must provide an alternative construction for this case. Luckily, we can use a similar approach to the one employed in Algorithm 3 and let $\omega = [2, 1, 4, 3]$. This secret permutation induces the same infinite loop incurred earlier in this section, so although D does not take the form $[1, n - 1, \dots,]$, this method still produces an offending permutation.

Cyclic Shifting Components

There are two possible displacement vectors which we omitted from our analysis thus far: the vectors $[1, 1, \dots, 1]$ and $[n - 1, n - 1, \dots, n - 1]$. Observe that the latter is the case when $S[n]^{-1} = [n, 1, 2, \dots, n - 1]$, which occurs when $S[n] = [2, 3, \dots, n, 1]$. But this is exactly cyclic shift, which never repeats incorrect information by Lemma 2.1. Then we need only concern ourselves with the former case, which results when $S[n]^{-1} = [2, 3, \dots, n, 1]$, i.e. when $S[n] = [n, 1, 2, \dots, n - 1]$. An inductive strategy with such an $S[n]$ left shifts for the n -th component but right shifts for all other components, which has previously been dubbed CSL to denote the *left* shifting in the n -th component [1].

To wholly prove Theorem 2.1, we must also construct at least one offending permutation for CSL. As before, we note that $S[n] = [n, 1, 2, \dots, n - 1]$, so ω must be a derangement so that $S[n]$ is used to generate at least one guess. Then $\gamma_2 = S[n]^{-1} = [2, 3, \dots, n, 1]$, and all entries have displacement of 1, but we must have $\mathcal{J}_2 \neq \emptyset$ in order for right-shifting to take over. For simplicity, let $\mathcal{J}_2 = \{1\}$ so that the leading 2 in γ_2 is the only correct entry. Then $\gamma_3 = [2, 1, 3, \dots, n]$, and so the entries $\{3, \dots, n\}$ are all guessed in the same incorrect positions as they were in γ_1 . Letting $\mathcal{J}_3 = \mathcal{J}_2 = \{1\}$ produces $\gamma_4 = [2, n, 1, 3, \dots, n - 1]$, which is a valid final guess as long as $n \geq 4$. Each entry $j \in \{3, \dots, n - 1\}$ ends in position $j + 1$ after being guessed in position j (in γ_1 and $\gamma - 3$) and $j - 1$ (in γ_2), entry 1 is guessed in position one and position n before ending in position three, and entry n is guessed in position n and position $n - 1$ before ending in position two. So what if $n = 3$? Then $S = [[1], [2, 1], [3, 1, 2]]$, which is the leftward cyclic shifting strategy. By Lemma 2.1, S has no offending permutations, so we have considered all possible non-cyclic shifting strategies.

3. Generalization to All Strategies

In Section 2, we constructed offending permutations for inductive strategies based upon the n -th strategy component, $S[n]$. We will now extend this framework to any strategy, not just inductive ones, in an effort to prove the stronger, overall claim that *any* strategy (other than CS) repeats incorrect information for at least one permutation.

Recall that the components of a strategy S are required to be derangements. Of course, $S[1] = [1]$ for completeness, and $S[2] = [2, 1]$ since this is the only derangement of length 2. Any strategy has these same two components, so for an arbitrary strategy S we will define $\kappa := \min\{k \mid S[k] \neq \text{CS}[k]\}$ and note that $2 \leq \kappa \leq n$. To construct an offending permutation, we want to utilize our previous work with inductive strategies. Observe that the first κ components of S form an inductive strategy of length κ . Then we will construct ω by fixing the permutation entries $\kappa + 1, \kappa + 2, \dots, n$ so that after guessing the trivial permutation for γ_1 , we have that $\mathcal{I}_1 = \{1, 2, \dots, \kappa\}$ and we can then use the construction from Section 2. In other words, the offending permutation ω is made up of a sub-permutation of length κ constructed according to our previous algorithms and a sub-permutation of length $n - \kappa$ consisting exclusively of fixed points $[\kappa + 1, \dots, n]$.

But wait! Say that $\kappa = 3$. Then the only derangement of length three which is not right-cyclic shifting is the permutation $[3, 1, 2]$, which corresponds to left-cyclic shifting as discussed in Section 2. However, there is no algorithm to construct an offending permutation for this case, and in fact the strategy $S = [[1], [2, 1], [3, 1, 2]]$ has no offending permutations at all by Lemma 2.1 since it is cyclic shifting. So we must generalize our offending permutation construction when $S[3] = [3, 1, 2]$, i.e., when inductive cyclic shifting occurs *leftward*. To do this, we must flip our construction so that it accommodates the leftward shifting later in the game. Then we define an analog to displacement: the *left displacement*, as follows:

Definition 3.1. *The left displacement vector $d_\ell(\pi)$ of a permutation $\pi \in S_n$ is the list of length n such that for all $1 \leq i \leq n$, $d_\ell(\pi)[i] := i - \pi[i] \pmod n$.*

Then when an entry in an inductive strategy is correct, we left shift the incorrect entries backwards into positions of repetition, as before. The overall approach outlined in Algorithms 1-3 therefore still holds so long as Step 3 shifts leftward rather than rightward. The only other modification we must make is to the approach used for CSL in Section 2. In this section, for the strategy $\text{CSL} = [[1], [2, 1], \dots, [2, \dots, n - 1, 1], [n, 1, \dots, n - 1]]$, we had the offending permutation $\omega = [2, n, 1, 3, \dots, n - 1]$. The analog for left-cyclic shifting is the strategy CSR defined like so:

$$\text{CSR} = [[1], [2, 1], [3, 1, 2], \dots, [n - 1, 1, \dots, n - 2], [2, \dots, n, 1]]$$

As before, we let $\mathcal{J}_1 = \emptyset$ so that $\gamma_2 = [n, 1, 2, \dots, n - 1]$, and we then let $\mathcal{J}_2 = \{1\}$ so left-shifting all entries in positions $\{2, \dots, n\}$ yields $\gamma_3 = [n, 2, \dots, n - 1, 1]$. Here we see that entries $\{2, \dots, n - 1\}$ are repeated in a familiar manner, and left-shifting the remaining entries yields $\gamma_4 = [n, 3, \dots, n - 1, 1, 2]$, which is a legal end state for the game. Then we have the following (modified) algorithm for an inductive strategy whose final component cyclic shifts in the opposite direction:

With these modifications, Algorithm 5 constructs an offending permutation for a general strategy. This construction is also implemented in the second Maple package linked in the appendix as the procedure `constructgen(s)`.

Algorithm 4 Offending Permutation Constructor For CSL and CSR

1. If $S = \text{CSL} = [[1], [2, 1], \dots, [2, \dots, n-1, 1], [n, 1, \dots, n-1]]$, then $\omega = [2, n, 1, 3, \dots, n-1]$
2. If $S = \text{CSR} = [[1], [2, 1], [3, 1, 2], \dots, [n-1, 1, \dots, n-2], [2, \dots, n, 1]]$, then $\omega = [n, 3, \dots, n-1, 1, 2]$

Algorithm 5 Offending Permutation Constructor For a General Strategy S

1. Check $S[3]$ to determine whether our base strategy is left or right shifting. If $S[3] = [2, 3, 1]$, then our base is right shifting. If $S[3] = [3, 1, 2]$, then our base is left shifting.
2. Let $\kappa = \min\{k \geq 4 \mid S[k] \text{ is not cyclic shifting in the direction from Step 1}\}$
3. If Step 1 was right-cyclic shifting, let $D = d(S[k]^{-1})$ and use casework from Section 2 to construct ω' .
If Step 1 was left-cyclic shifting, let $D = d_\ell(S[k]^{-1})$ and construct ω' using casework from Section 2 (adjusted to left shift on Step 3) and from the modification in Algorithm 4.
4. The offending permutation is $\omega = [\omega'[1], \omega'[2], \dots, \omega'[k], k+1, \dots, n]$

4. Conclusion

In this paper, we presented an overall algorithm that takes an input strategy for a game of permutation wordle and outputs a permutation which, when acting as the secret permutation in a game using that strategy, causes incorrect information to be duplicated. Such duplication is inherently suboptimal, so the fact that this algorithm works for *any* strategy other than cyclic shift leads us to the conclusion that CS is optimal in this sense. Now, it is important to note that any given strategy may have more than one offending permutation, and in fact there is often more than one. Counting and characterizing these permutations for a fixed strategy remains an open area of investigation.

For instance, in [1], the sub-optimality of CSL was studied specifically for games ending in precisely three guesses, and it was proven to be definitively the worst inductive strategy in the sense that it guesses the *fewest* secret permutations in three guesses or fewer. Experimental evidence also suggests that CSL under-performs compared to all other inductive strategies when analyzing offending permutations. The number of offending permutations for the strategy CSL of length $n \geq 4$ conjecturally exceeds the number of offending permutations for any other inductive strategy of the same length, and the number of such permutations is counted by the sequence 4, 35, 244, 1813, 14740, \dots . This sequence does not yet have an entry in Sloane's Online Encyclopedia of Integer Sequences [6] nor does it have a known closed form expression.

Furthermore, our strategies fix each component, $S[k]$, through the course of a game. This standardizes our analysis and experimental computation, but note that it is possible to change $S[k]$ in the middle of a game, particularly if two consecutive guesses each resulted in the same number of incorrect entries. Say, for example, that in a game of length 5 your first guess was the trivial permutation $\gamma_1 = [1, 2, 3, 4, 5]$, but $\mathcal{J}_1 = \emptyset$. Say that you right-shifted to obtain $\gamma_2 = [5, 1, 2, 3, 4]$. If $\mathcal{J}_2 = \emptyset$ again, you may choose to use a different, non-cyclic shifting permutation to permute your five incorrect entries and produce γ_3 . Under our formulation, $S[5]$ is fixed, so this would never happen, but in a real game this isn't forbidden.

Appendix

This paper is supported by two Maple packages, which are linked at <https://aurorahiveley.github.io/wordle.txt> and <https://aurorahiveley.github.io/wordlerep.txt> [2]. The first package was written for [1] and has been modified once from its original published state to fix a bug in an error checking procedure. The second package supports this paper, and it includes examples as well as procedures which (experimentally) verify each of the theorems presented.

Acknowledgement

The author thanks her advisor Dr. Doron Zeilberger for the introduction to the problem and feedback on earlier drafts.

References

- [1] A. Hiveley, Experimenting with permutation wordle, *J. Integer Seq.* **29** (2026) #26.1.7.
- [2] A. Hiveley, *Computational Support To Repetition in Permutation Wordle*, Available at <https://aurorahiveley.github.io/wordlerep.txt>.
- [3] S. Kutin, L. Smithline, Permutation wordle, *arXiv:2408.00903* [math.CO], (2024).
- [4] R. Li, S. Zhu, Playing mastermind with wordle-like feedback, *Amer. Math. Monthly* **131** (2024) 390–399.
- [5] R. Stanley, *Guessing a Permutation in a Way Analogous to Wordle*, MathOverflow, 2025, Available at <https://mathoverflow.net/q/497111>.
- [6] The OEIS Foundation Inc., *The On-Line Encyclopedia of Integer Sequences*, Available at <https://oeis.org>.