VEXILLARY PERMUTATIONS

MAX GROSS AND NICOLE MARSAGLIA

Advisors: Sara Billey, Brendan Pawlowski, Jair Taylor
University of Washington

Abstract. This paper examines the permutation matrix of a vexillary permutation and its reduced expressions. The Vexillary Metropolis Algorithm is created to generate random vexillary permutations and notable properties of this algorithm are discussed. Reduced expressions are generated in a similar fashion as in Angel, Holroyd, Romik and Virág's paper "Random Sorting Networks," by employing the Hook Walk Algorithm and inverse Edelman-Green process.

1. Introduction

1.1. Preliminaries.

Definition 1.1. A *permutation* is a bijection from a set S of size n to itself. One can represent a permutation $\rho : S \to S$ by its one line notation $[\rho(1), \ldots, \rho(n)]$. The set of permutations on S forms a group under composition. The identity permutation is $[1, 2, 3, \ldots, n]$.

One way to graphically show a permutation is in its matrix form. The matrix will be of size $n \times n$. The permutation $\rho : S \to S$ has 1's in positions $(i, \rho(i))$ and 0's everywhere else. Figure 1 shows the matrix for the permutation $\rho = [6, 3, 5, 2, 4, 1]$.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Definition 1.2. For $1 \leq s \leq n-1$, define the *simple transposition* τ_s as the permutation $[1, 2, \ldots, s+1, s, \ldots, n]$.

Example 1.3. Given the permutation $\rho = [\rho(1), \rho(2), \rho(3), \rho(4), \rho(5)]$, the product $\rho\tau_2 = [\rho(1), \rho(3), \rho(2), \rho(4), \rho(5)]$.

Definition 1.4. If ρ is a permutation, then a *reduced expression* for ρ is any minimal length product of simple simple transpositions $\tau_{a_1} \ldots \tau_{a_k} = \rho$.

Example 1.5. One reduced expression for the permutation $[1, 3, 4, 2]$ is $\tau_2\tau_3$. A reduced expression for $[5, 1, 3, 2, 4]$ is $\tau_4\tau_3\tau_2\tau_1\tau_3$.

Date: 8/8/2012.
University of Washington 2012 REU.

There are many different reduced expressions for the same permutation. The braid relation states that $\tau_i\tau_{i+1}\tau_i = \tau_{i+1}\tau_i\tau_{i+1}$. The commutation relation states that $\tau_i\tau_j = \tau_j\tau_i$ if $|i-j| > 1$. Using the braid and commutation relations, every possible reduced expression for a permutation can be found given a single reduced expression.

Definition 1.6. A *reduced word* is the list of indices of a reduced expression of a permutation.

Definition 1.7. For a permutation, an *inversion* is a pair, i, j, when $1 \leq i < j \leq n$ and $\rho(i) > \rho(j)$.

Example 1.8. $\rho = [5,3,2,4,1]$ has 8 total inversions $(5,3),(5,2),(5,4),(5,1),(3,2),(3,1),(2,1)$ and $(4,1)$. The total number of inversions in a permutation is equal to the length of any reduced expression.

Definition 1.9. A permutation has a *descent* when for some $1 \leq i \leq n$, $\rho(i+1){<}\rho(i)$. The other entries are then called *ascents*.

1.2. **Insertion and Recording Tableaux.**

Definition 1.10. A *partition*, $\lambda = (\lambda_1 \geq \lambda_1 \geq \ldots \geq \lambda_n)$, of a positive integer n is a nonincreasing sequence of positive integers.

Definition 1.11. A *partition diagram* of a partition $\lambda = [\lambda_1, \ldots, \lambda_r]$ is an array of left justified boxes where the first row has λ_1 boxes, the second λ_2 boxes, and so on until the last row has λ_r boxes.

Definition 1.12. A *tableau* is a partition diagram where each cell has been assigned a positive integer such that every row and column of the diagram contain increasing sequences of numbers. The top-left cell is denoted as (1,1) and (1,2) is to its right. [1]

Definition 1.13. Given a partition diagram with n cells, a *standard tableau* is a tableau that uses the numbers $1, \ldots, n$ exactly once.

In their 1987 paper entitled "Balanced Tableaux," authors Edelman and Greene discuss the ways in which tableau and standard tableau can be used in regard to permutations and their reduced expressions. They define an insertion tableau as a visual representation of the reduced word of a permutation. Each element of the reduced word is placed into a cell under the guidelines of a few rules. Starting with the first element of the reduced word in cell (1, 1), a new cell is created with the inclusion of each new element. With i_j being the element of the reduced word to be added and y representing the elements already entered into the partition, an insertion tableau is constructed by the following three rules:

(1) If every number in row 1 is smaller than i_j, append i_j to the end of the row.
(2) If i_j appears in row 1, insert $i_j + 1$ into the next row.
(3) Otherwise find the smallest $y \geq i_j$ in row 1, replace y by i_j, and insert y into the next row by the above rules.

Example 1.14. Two possible reduced expressions for the reverse permutation $[4,3,2,1]$ are $\tau_3\tau_2\tau_1\tau_2\tau_3\tau_2$ and $\tau_3\tau_2\tau_3\tau_1\tau_2\tau_3$. The insertion tableaux for the reduced expressions are built in the following ways:

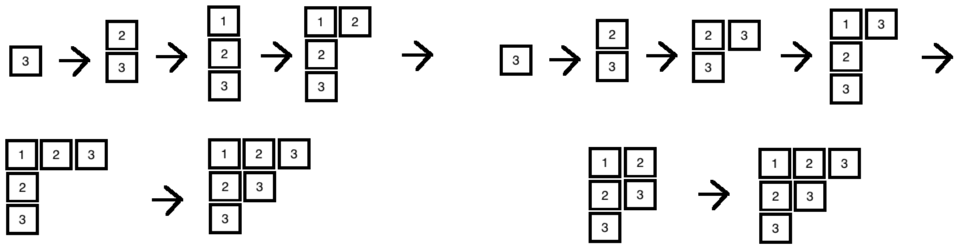


Figure 1. The process of creating the insertion tableau for the reduced expressions $\tau_3\tau_2\tau_1\tau_2\tau_3\tau_2$ and $\tau_3\tau_2\tau_3\tau_1\tau_2\tau_3$ respectively.

Edelman and Greene define recording tableaux, as standard tableaux which keep track of the order in which cells of the insertion tableaux are created. In the recording tableaux, the cell $(1, 1)$ is always labeled with a 1, since it is always the first cell created. The second cell that is created in the insertion tableau is labeled with a 2, the third cell with a 3 and so on until the recording tableau for a permutation has the same shape as its corresponding insertion tableau. The Edelman-Greene process is the construction of an insertion tableau and a recording tableau from a given reduced expression.

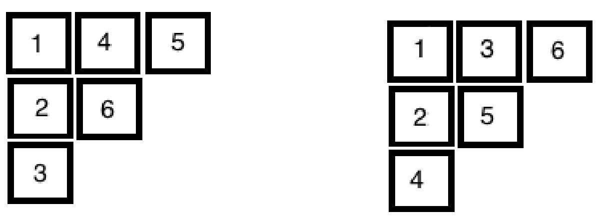


Figure 2. The recording tableau for the reduced expressions $\tau_3\tau_2\tau_1\tau_2\tau_3\tau_2$ and $\tau_3\tau_2\tau_3\tau_1\tau_2\tau_3$ respectively.

Richard Stanley notes in his paper "On the Number of Reduced Decompositions of Elements of Coxeter Groups" that there is a bijection between reduced expressions for the reverse permutation $\rho = [n, n-1, n-2, ..., 2, 1]$ and the staircase shape for insertion and recording tableaux where the first row has n cells, the second row has $n-1$ cells, the third row has $n-2$ cells and the final row has 1 cell. In the case of the reverse permutation, there will only be one insertion tableau, which is of a staircase shape.

Edelman and Greene generalize this for all permutations. Given $P(\rho)$ is an insertion tableau for ρ, then there is a bijection between the reduced expression ρ and $\{(P, Q) : P \in P(\rho)$ where Q is standard and the shape of Q is the same as the shape of $P\}$. That is, Edelman and Greene prove that each reduced expression has only one pair of insertion and recording tableau. Denote the Edelman-Greene process as the building of the pair of insertion and recording tableau from a reduced expression.

1.3. **Generating Random Reduced Expressions.** There are two steps to generate a reduced expression for a permutation uniformly at random. First, the Hook Walk Algorithm builds a random recording tableau of a given shape. Second, the inverse Edelman-Greene process uses an insertion tableau and a recording tableau to build a reduced expression for a given permutation.

The first step of this process is the Hook Walk Algorithm. Given the shape of an insertion tableau for a permutation, create a corresponding standard tableau using the following steps:

(1) Choose a cell in the partition uniformly at random.
(2) Choose a cell in the hook of the chosen cell uniformly at random.
    - The hook of a cell in the partition is defined as all of the cells in the same row as the chosen cell but to the right of it and all of the cells in the same column as the chosen cell but below it.
(3) Continue step 2 until the chosen cell is a corner cell.
(4) Label the corner cell with the number of cells in the partition.
(5) Remove the labeled cell from the partition and repeat steps 1 through 4 until the final cell is labeled with a 1.

Example 1.15. To implement the Hook Walk Algorithm and yield a random recording tableau, one must begin with a permutation and one of its reduced expressions. The permutation $[4, 3, 5, 2, 1]$ and its reduced expression $\tau_2, \tau_1, \tau_3, \tau_2, \tau_3, \tau_4, \tau_3, \tau_1$, has the insertion tableau shown in Figure 4.

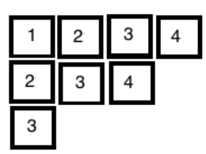


Figure 3

The shape of the insertion tableau is used to perform the Hook Walk Algorithm and generate a random recording tableau, as shown in Figure 5.

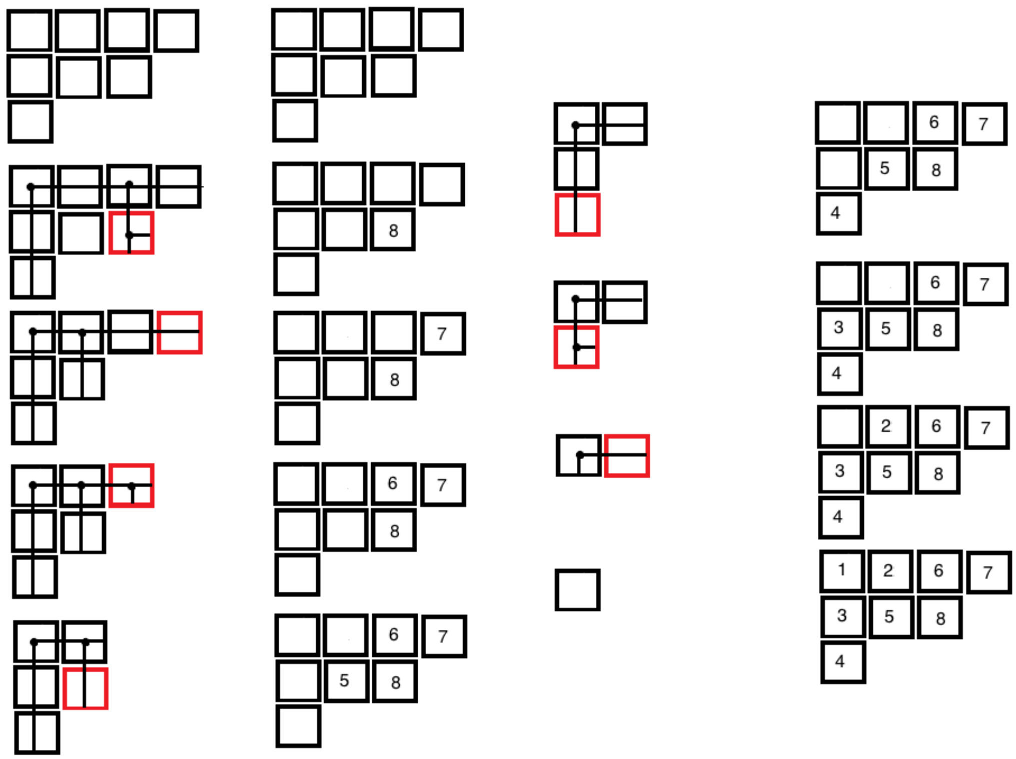


FIGURE 4

The Hook Walk Algorithm attains a recording tableau, denoted as Q, of the same shape as the given insertion tableau, denoted as P, uniformly at random. Reversing the Edelman-Greene process yields a reduced expression chosen uniformly at random from the set of all reduced expressions with P as the insertion tableau. The Edelman-Greene Process builds a set (P, Q) given a reduced expression, so the reverse Edelman-Greene Process builds a reduced expression given a set (P, Q).

Example 1.16. Applying the reverse Edelman-Greene bijection to Example 1.15, one can produce a unique reduced expression for a given permutation from a pair (P, Q).

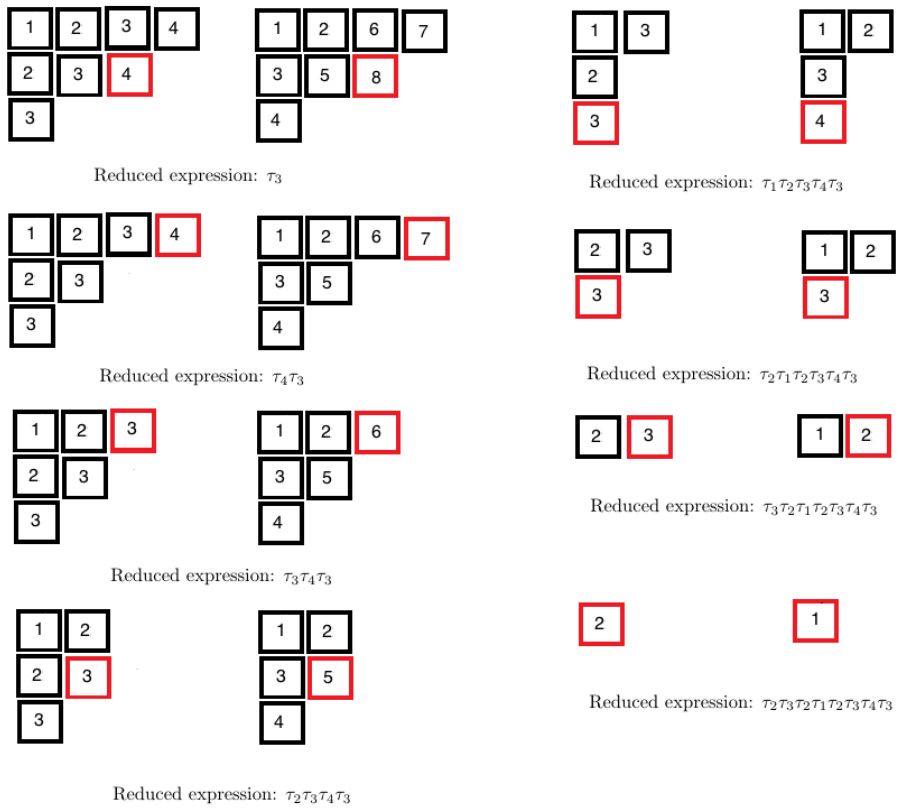


FIGURE 5. Working backwards from the insertion and recording tableaux, the reverse Edelman-Greene Process produces a reduced expression.

1.4. **Angel, Holroyd, Romik and Virág's Findings.** In 2006 Omer Angel, Alexander E. Holroyd, Dan Romik and Bálint Virág authored a paper entitled "Random Sorting Networks." One of their primary findings was that the permutation matrix of the reverse permutation after a certain amount of time through the reduced expression had several important properties. They came to these discoveries by generating an insertion tableau for a reverse permutation and then using the Hook Walk Algorithm and the inverse Edelman-Green bijection to choose a random reduced expression for the reverse permutation. Using this random reduced expression for the reverse permutation, Angel, Holroyd, Romik and Virág were able to come to some notable conclusions.

Denote time as t. At $t = \frac{1}{2}$, the permutation matrix provides the visual of the permutation when $\frac{1}{2}$ of the steps of the reduced expression have been completed. Likewise, at $t = \frac{1}{3}$, the permutation matrix provides the visual of the permutation when $\frac{1}{3}$ of the steps of the reduced expression have been completed. Angel, Holroyd, Romik and Virág proved that as n approaches infinity the 1's in the permutation matrix will be within certain octagons (outlined in the figure below) with probability 1. More interestingly, they conjectured that at $t = \frac{1}{2}$, the points in the permutation matrix for the reverse permutation form the projection of a circle.

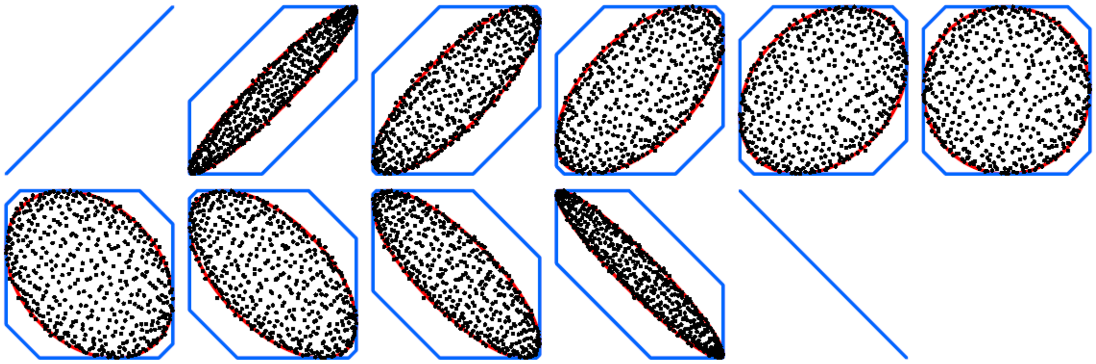


FIGURE 6. Shows the progression of the reverse permutation throughout a reduced expression

These findings are quite fascinating since it is appears to be unexpected that the basic geometry of a sphere would arise. Angel, Holroyd, Romik and Virág only examined the permutation matrices of various reverse permutations. This leaves open questions as to whether other types of permutations would produce similarly interesting results.

1.5. **Vexillary Permutations.**

Definition 1.17. Given permutations ρ and ρ', for $i, j \in \rho'$, if for all $i \in \rho'$, $i \in \rho$ and for $\rho(i)<\rho(j)$ then $\rho'(i)<\rho'(j)$ for all i, j, then ρ' is a *subpermutation* of ρ.

Definition 1.18. In a permutation, if there are four elements i, j, k, l such that $i<j<k<l$ and $\rho(j)<\rho(i)<\rho(l)<\rho(k)$, then the permutation is *nonvexillary*. Permutations that are 2143-avoiding are called *vexillary*.

Example 1.19. The permutation $[1, 5, 6, 4, 2, 3, 7]$ is vexillary because it is 2143-avoiding. The permutations $[1, 5, 4, 2, 3, 7, 6]$ and $[4, 6, 1, 2, 7, 3, 5]$ are nonvexillary because they are not 2143-avoiding. They contain the subpermutations $(5, 4, 7, 6)$ and $(4, 1, 7, 5)$ respectively, which fail the vexillary conditions.

An important distinction can be made between the insertion tableaux of vexillary and nonvexillary permutations. A vexillary permutation has one and only one insertion tableau. That is, all of the reduced expressions for a vexillary permutation will produce the same insertion tableau. A nonvexillary permutation, in contrast, will always have more than one insertion tableau, meaning that different reduced expressions for a nonvexillary permutation may produce different insertion tableaux.

Example 1.20. An example of two different insertion tableaux for the reduced expressions of the same nonvexillary permutation $[5, 2, 1, 4, 3]$.

Figure 7. Insertion tableaux for the reduced expressions $\tau_3\tau_4\tau_3\tau_1\tau_2\tau_1$ and $\tau_4\tau_3\tau_2\tau_1\tau_2\tau_4$ respecitvely

The question of how a permuation matrix of a permutation transforms over the time of a reduced expression has only been investigated for reverse permutations. A fast method for choosing a random reduced expression is only known for vexillary permutations. Therefore this paper will examine how a permutation matrix of a vexillary permutation transforms over the time of a reduced expression. This distinction between vexillary and nonvexillary permutations leads to the question of how the permutation matrix of a vexillary permutation transforms over time of a reduced expression.

2. Vexillary Metropolis Algorithm

The goal of this paper is to examine what the permutation matrix of a vexillary permutation looks like over time of a random reduced expression. The first step of this process is to choose a vexillary permutation unfiformly at random. There is a method of doing this which employs a bijection between 1234-avoiding permutations and 2143-avoiding permutations. Instead of using this method, this paper will use the Vexillary Metropolis Algorithm to randomly choose a vexillary permutation.

2.1. **The Vexillary Metropolis Algorithm.** Beginning with an identity permutation of length n, the Vexillary Metropolis Algorithm (VMA) generates a random vexillary permutation after a chosen number of simple simple transpositions. The steps of the VMA are as follows:

(1) Randomly select an element α_x from the permutation such that $1 \leq \alpha_x < n$.
(2) Swap α_x with the neighboring element to its right.
   - $\alpha_1\alpha_2 \ldots \alpha_x\alpha_{x+1} \ldots \alpha_{n-1}\alpha_n \longrightarrow \alpha_1\alpha_2 \ldots \alpha_{x+1}\alpha_x \ldots \alpha_{n-1}\alpha_n$.
(3) If this new permutation is vexillary, record it as a successful simple transposition and repeat steps (1) and (2) for the given number of simple transpositions. However, if this new permutation is nonvexillary, undo the swap and do not record it as a successful simple transposition. Repeat steps (1) and (2) for the given number of simple transpositions.

Essentially, the VMA randomly chooses an element in the permutation and swaps it with the element to its right. If this new permutation remains vexillary, then a new element is chosen and swapped. If the new permutation is nonvexillary, then the swap is undone and a new element is chosen. The following example shows the Vexillary Metropolis Algorithm for a permutation of length 6 after 2 simple transpositions.

Example 2.1.

$$\rho_0 = [1, 2, 3, 4, 5, 6] \tag{1}$$

Begin with the identity permutation of length 6.

$$\rho_1 = [1, 3, 2, 4, 5, 6] \tag{2}$$

α_x was randomly chosen as 2, so the 2 and 3 were swapped. The permutation remains vexillary after the swap, so it is recorded as one successful simple transposition.

$$\rho_2 = [1, 3, 2, 4, 6, 5] \tag{3}$$

α_x was randomly chosen as 5, so the 5 and 6 were swapped. The permutation no longer is vexillary, due to the nonvexillary subpermutation (3,2,6,5). The swap is undone and is not recorded as a successful simple transposition, so the permutation again is ρ_1.

$$\rho_2 = [3, 1, 2, 4, 5, 6] \tag{4}$$

α_x was randomly chosen as 1, so the 1 and 3 were swapped. The permutation remains vexillary after the swap, so it is recorded as the second simple transposition.

According to Example 2.1, one possible vexillary permutation of length 6 after 2 simple transpositions generated using the Vexillary Metropolis Algorithm is $\rho = [3, 1, 2, 4, 5, 6]$. It is important to note the very strict restrictions that the VMA places on each simple transposition. Beginning with the identity, any element can be transposed with its neighbor to the right and the permutation will always remain vexillary. However, the options of which elements in the permutation can be transposed on the second simple transposition is much less. In fact, there are only two elements which can be transposed in order for the permutation to remain vexillary after the second simple transposition. For the second simple transposition in Example 2.1, if the 1, 2, or 3 are chosen as α_x, the permutation will remain vexillary, yet if any other element is chosen as α_x, the permutation will become nonvexillary. The most important feature of the VMA is the fact that once a swap causes a permutation to become nonvexillary, the swap is undone and is not recorded as a simple transposition. This inability to create nonvexillary permutations creates some interesting properties that will be discussed in detail later.

Proposition 2.2. *If, after a simple transposition, there exists more than one nonvexillary subpermutations, the only simple transposition that can be performed is switching back the transposed elements.*

Proof. Suppose ρ is a vexillary permutation that is one simple transposition away from having multiple nonvexillary subpermutations. Thus, ρ has more than one inversion within the permutation such that after a simple transposition another inversion is created that makes the permutation nonvexillary. Clearly, none of the other inversions can be transposed to make ρ vexillary again because the unchanged inversions composing a different nonvexillary subpermutation would still remain. Hence, the only option is to switch the originial elements back, applying any other simple transposition would not make the permutation vexillary. □

2.2. **The Limitations of the VMA.** At any given number of simple transpositions, the permutations that the VMA is able to generate is limited. In other words, with a set number of simple transpositions, the VMA cannot generate every possible permutation of a cetain length. There is an even/odd distinction which must be made when dealing with the VMA. After an even number of simple transpositions on a permutation of length n, the possible vexillary permutations that may be generated are rooted in a pool of only about half of all possible length n vexillary permutations. Similarly, after an odd number of simple transpositions on a permutation of length n, the possible vexillary permutations that may be generated are rooted in a pool of only about half of all possible length n vexillary permutations.

Vexillary permutations of length $n = 4$ tend to be of little interest, since $\rho = [2, 1, 4, 3]$ is the only possible nonvexillary permutation when $n = 4$. However, the property of a simple transposition restricting the possible vexillary permutations that the VMA can generate to about half of all possible vexillary permutations of a certain length is easily seen at this length. In the following diagram, each row represents a simple transposition.

0)	[1,2,3,4]
1)	[2,1,3,4]; [1,3,2,4]; [1,2,4,3]
2)	[1,2,3,4]; [2,3,1,4]; [3,1,2,4]; [1,3,4,2]; [1,4,2,3]
3)	[2,1,3,4]; [1,3,2,4]; [1,2,4,3]; [3,2,1,4]; [2,3,4,1]; [3,1,4,2]; [1,4,3,2]; [4,1,3,2]
4)	[1,2,3,4]; [2,3,1,4]; [3,1,2,4]; [1,3,4,2]; [1,4,2,3]; [3,2,4,1]; [2,4,3,1]; [3,4,1,2]; [4,1,3,2]; [4,2,1,3]
5)	[2,1,3,4]; [1,3,2,4]; [1,2,4,3]; [3,2,1,4]; [2,3,4,1]; [3,1,4,2]; [1,4,3,2]; [4,1,2,3]; [3,4,2,1]; [4,2,3,1]; [2,4,1,3]; [4,3,1,2]
6)	[1,2,3,4]; [2,3,1,4]; [3,1,2,4]; [1,3,4,2]; [1,4,2,3]; [3,2,4,1]; [2,4,3,1]; [3,4,1,2]; [4,1,3,2]; [4,2,1,3]; [4,3,2,1]

This diagram displays the $n = 4$ permutations that are able to be generated after each simple transposition using the Vexillary Metropolis Algorithm. The VMA always begins with the identity, which is labeled here as configuration 0. After one simple transposition, the only possible permutations that can be generated using the VMA are $[2, 1, 3, 4]; [1, 3, 2, 4]$; and $[1, 2, 4, 3]$, which result from when $\alpha_x = 1$, $\alpha_x = 2$ and $\alpha_x = 3$ respectively. These three permutations are the only permutations that can possibly be generated using the VMA on $n = 4$ permutations after the first simple transposition. Using similar logic, after the second simple transposition, the only possible permutations that can be generated using the VMA are those in row number 2. Notice that row 2 contains the permutation $[1, 2, 3, 4]$, which appeared in row 0, but does not contain any of the permutations that appeared in row 1. This trend continues as the permutations that appear in row 3 contain all of those in row 1, yet none in row 2. The permutations that appear in row 4 contain all of those in row 0 and row 2 but none in row 1 or row 3. It quickly becomes evident that some permutations are able to be generated using the VMA only after an even number of simple transpositions while others are only able to be generated under the VMA after an odd number of simple transpositions.

As the diagram shows, 5 simple transpositions using the VMA on an $n = 4$ permutation can yield a total of 12 possible permutations. Similarly, 6 simple transpositions using the VMA on an $n = 4$ permutation can yield a total of 11 possible permutations. If the diagram continued to show row 7, it would be identical to row 5. Row 8 would be identical to row 6. Since there are 23 $n = 4$ vexillary permutations, it is clear that 12 of them are able to be generated after an odd number of simple transposition and 11 of them are able to be generated after an even number of simple transpositions. This pattern of some permutations appearing only after an odd number of simple transpositions and others only after and even number is not only true for permutations of length 4. Using the VMA for permutations of any length n, some of the vexillary permutations of length n are able to be generated after an even number of simple transpositions while the rest are able to be generated after an odd number of simple transpositions.

The purpose of the Vexillary Metropolis Algorithm is to generate a random vexillary permutation. Since the number of simple transpositions chosen has a great effect on the permutations that can be generated, an extra step is necessary after choosing the number of simple transpositions in order to preserve randomness. After choosing the number of simple transpositions, randomly select a 0 or a 1. If a 0 is selected, apply the VMA with the chosen number of simple transpositions. If a 1 is selected, apply the VMA with one more than the chosen number of simple transpositions. This guarantees that there is an equally likely chance of having an even or odd number of simple transpositions. In turn, this translates to an equally likely chance of generating a vexillary permutation from those that are only able to be generated after an even number of simple transpositions as from those that are only able to be generated after an odd number of simple transpositions.

2.3. **Vexillary Permutations That are One simple transposition from Being Nonvexillary.** A simple transposition on a vexillary permutation can create a nonvexillary permutation, as has been shown in examples. Likewise, a simple transposition on a nonvexillary permutation can create a vexillary permutation. This is seen most simply through the $n = 4$ nonvexillary permutation $\rho_0 = [2, 1, 4, 3]$. Let $\alpha_x = 2$. This simple transposition produces $\rho_1 = [1, 2, 4, 3]$, which is certainly a vexillary permutation.

Using the Vexillary Metropolis Algorithm, a simple transposition will never be executed on a nonvexillary permutation, since any simple transposition which generates a nonvexillary permutation will be undone. Therefore, any vexillary permutation that can be produced after a simple transposition on a nonvexillary permutation will not be able to be produced in this manner using the VMA. The VMA requires that such vexillary permutations which arise after a simple transposition on a nonvexillary permutation must be produced through a series of simple transpositions which always avoids the creation of a nonvexillary permutation. This naturally leads to the question as to whether the VMA is able to generate every possible vexillary permutation of a certain length, since it seems as though the vexillary permutations which arise after a simple transposition of a nonvexillary permutation may not always be able to be produced.

Proposition 2.3. *There are three possible vexillary permutations that can be generated from a simple transposition of a nonvexillary permutation*

Proof. For $\alpha_i \in [1, \ldots, n]$ and $\alpha_1 < \alpha_2 < \alpha_3 < \alpha_4$ the VMA can generate the following three nonvexillary permutations before switching the the transposed elements back:

$$\ldots \alpha_2 \alpha_1 \ldots \alpha_4 \ldots \alpha_3$$

$$\ldots \alpha_2 \ldots \alpha_1 \alpha_4 \ldots \alpha_3$$

$$\ldots \alpha_2 \ldots \alpha_1 \ldots \alpha_4 \alpha_3$$

By implementing a simple transposition, there are three possible vexillary permutations that can be generated from the above nonvexillary permutations, namely:

(1) $\alpha_1 \alpha_2 \alpha_4 \alpha_3$
(2) $\alpha_2 \alpha_1 \alpha_3 \alpha_4$
(3) $\alpha_2 \alpha_4 \alpha_1 \alpha_3$

$\square$

Proposition 2.4. *Every vexillary permutation of a given length n can be generated from the VMA.*

Proof. In order to generate all possible vexillary permutations of a given length n, one must be able to generate the three permutations from Proposition 2.3 above without utilizing a nonvexillary permutation in the process.
From Proposition 2.3:

(1) $\alpha_1\alpha_2\alpha_4\alpha_3$
(2) $\alpha_2\alpha_1\alpha_3\alpha_4$
(3) $\alpha_2\alpha_4\alpha_1\alpha_3$

To generate (1) and (2), this can be done through induction. Let ω be a vexillary permutation. Let $m(\omega) = \#\{\text{inversions in } \omega\}$. For $\omega = [1, 2, \ldots, n]$, $m(\omega) = 0$. To show that every vexillary permutation can be obtained from the identity by only performing simple transpositions on some ω, $m(\omega)$ needs to 1) decrease by one and 2) remain vexillary.

By induction:
Base Case: For $m(\omega) = 1$, clearly a simple transposition can be performed that keeps ω vexillary and decreases the number of inversions of ω by one. This is because it is impossible for a permutation with one inversion to be nonvexillary.
Now suppose that the two conditions for $m(\omega)$ holds for $1 \le m(\omega) < k$.
The goal is to show the conditions hold for $m(\omega) = k$. Since a simple transposition can always be performed on a descent an inversion will be undone, preventing the creation of a nonvexillary permutation. Hence, $m(\omega) = k$ holds under the two conditions.
Hence, through a series of simple transpositions on descents, it is proven that a vexillary permutation with an arbitrary number of inversions can be attained from the identity. Thus, every vexillary permutation, including the first two from Proposition 2.3, is generated through a simple transpositions and without utilizing a nonvexillary permutation in the process.
The process for (3) must be handled slightly differently, a problem arises when transposing the descent in 2413 becasue it will create the nonvexillary pattern 2143. But this problem can be easily remedied.
Let $\omega = [\ldots\alpha_2\ldots\alpha_4\alpha_1\ldots\alpha_3\ldots]$ and $m(\omega) = k$ such that $\alpha_4\alpha_1$ is the last descent. Then transposing α_1 past α_3 will increase the number of inversions by $l \in \mathbb{N}$, so $m(\omega) = k + l$. But then transposing α_4 past α_1 would decrease the number of inversions by $l - 1$, making $m(\omega) = k - 1$. A visual representation of this process follows:

$$\ldots\alpha_2\ldots\alpha_4\alpha_1\ldots\alpha_3\ldots \to \ldots\alpha_2\ldots\alpha_4\ldots\alpha_3\alpha_1\ldots \to \ldots\alpha_2\ldots\alpha_3\alpha_1\alpha_4\ldots$$

The result follows from induction similar to (1) and (2).

□

Proposition 2.5. *For a permutation of length n, performing a simple transposition can yield a maximum of $\frac{(n-2)(n-3)}{2}$ nonvexillary subpermutations.*

Proof. By virtue of the 2143 pattern, a nonvexilliary permutation must have two inversions with the latter having both elements larger than the former. So, to maximize the number of possible nonvexillary subpermutations after imposing a simple transposition on a vexillary permutation, it's clear that a permutation of length n must have the form $(n-2, n-3, \ldots, 1, n-1, n)$ or $(1, 2, n, n-1, \ldots, 3)$. Thus, transposing element $n-1$ with n or 1 with 2 respectively, will create the second inversion and a nonvexillary permutation with the most nonvexillary subpermutations.

Namely, the number of nonvexillary subpermutations is the following sum:

$$\sum_{2}^{n-2} n-1 = 1+2+3+\cdots+n-3 = \frac{(n-2)(n-3)}{2}$$

□

2.4. **Probability Distribution.** It is important to remember that the goal of the Vexillary Metropolis Algorithm is to generate a random vexillary permutation. As the number of chosen simple transpositions on a length n permutation approached infinity, it was expected that the probability of generating each different vexillary permutation would approach equality. That is, with a large number of simple transpositions, the distribution of vexillary permutations would approach a uniform random distribution. It quickly became evident, however, that no matter how many simple transpositions are performed, the probability distribution will never approach a uniform random distribution.

There will never be a uniform random distribution using the VMA because of two forces which compete to determine the likelihood that a vexillary permutation will be generated. The first of these forces is the minimum number of simple transpositions that must be applied to the permutation in order to arrive at the identity. This corresponds to the length of the reduced expression of the permutation. The second force is the minimum number of simple transpositions that must be applied to the permutation in order to arrive at a nonvexillary permutation. In general, with a small number of chosen simple transpositions, the first force wins out and with a large number of chosen simple transpositions, the second force wins out. That is, with a small number of chosen simple transpositions, the vexillary permutations that take the least number of simple transpositions to return to the identity are more likely to be generated using the VMA than vexillary permutations that take more simple transpositions to return to the identity. Likewise, with a large number of chosen simple transpositions, vexillary permutations which are more than one simple transposition from being nonvexillary are more likely to be generated using the VMA than vexillary permutations which are just one simple transposition from being nonvexillary.

These two forces can be shown graphically. Both bar charts in Figure 8 and Figure 9 display 100,000 $n = 5$ vexillary permutations generated using the Vexilary Metropolis Algorithm. The only difference in the two charts is that Figure 8 shows 100,000 $n = 5$ vexillary permutations after 12 simple transpositions while Figure 9 shows them after 100 simple transpositions. Neither charts display a uniform random distribution, yet for different reasons.

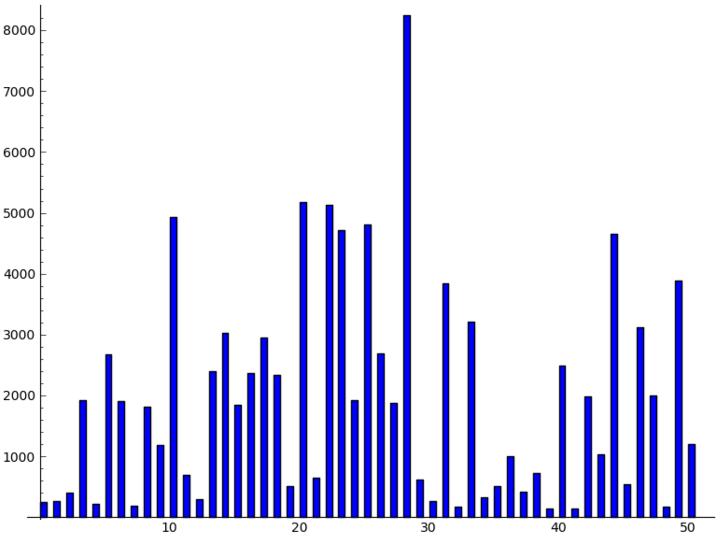


FIGURE 8. 12 simple transpositions

With only 12 simple transpositions, the vexillary permutations that take the least number of simple transpositions to return to the identity were generated more often than vexillary permutations that take more simple transpositions to return to the identity. The data of the first graph shows that $\rho_1 = [1, 2, 3, 4, 5]$ was generated 8.25% of the time and $\rho_2 = [3, 1, 2, 4, 5]$ was generated 4.66% of the time. Both ρ_1 and ρ_2 have reduced expressions of small length, with lengths 0 and 2 respectively. The data shows that $\rho_3 = [3, 4, 1, 2, 5]$ was generated 3.02% of the time and $\rho_4 = [5, 1, 4, 2, 3]$ was generated 1.84% of the time. Both ρ_3 and ρ_4 have reduced expressions of medium length, with lengths 4 and 6 respectively. The data shows that $\rho_5 = [5, 4, 2, 1, 3]$ was generated 0.33% of the time and $\rho_6 = [5, 4, 3, 2, 1]$ was generated 0.17% of the time. Both ρ_5 and ρ_6 have reduced expressions of large length, with lengths 8 and 10 respectively. As shown, the permutations with shorter reduced expressions are more likely to be generated than those with smaller reduced expressions for a small number of simple transpositions. Other data entries follow the same trend.

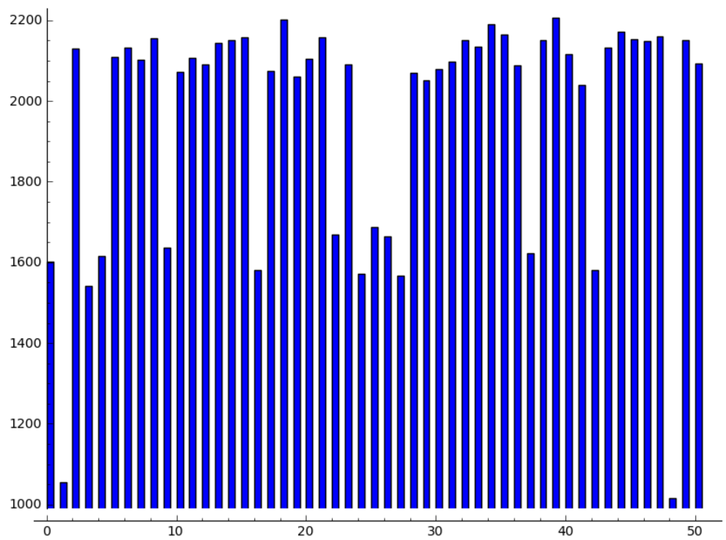


FIGURE 9. 100 simple transpositions

While the chart for 100 simple transpositions certainly closer than the chart for 12 simple transpositions, there is still not a uniform random distribution. With 100 simple transpositions, the vexillary permutations which are more than one simple transposition from being nonvexillary are more likely to be generated than vexillary permutations which are just one simple transposition from being nonvexillary. As the data for this graph shows, $\rho_1 = [2,5,1,3,4]$ and $\rho_2 = [2,3,5,1,4]$ were both generated 2.20% of the time. Both of these permutations are more than one simple transposition away from becoming nonvexillary. In fact, ρ_1 needs 3 simple transpositions to arrive at a nonvexillary subpermutation and ρ_2 needs 2 simple transpositions to become nonvexillary. The data of the graph also shows that $\rho_3 = [2,3,5,1,4]$ was generated 1.01% of the time and $\rho_4 = [2,5,1,3,4]$ was generated 1.02% of the time. Both ρ_3 and ρ_4 are one simple transposition away from becoming nonvexillary, as they are one simple transposition away from the subpermutations $(3,2,5,4)$ and $(2,1,4,3)$ respectively. As shown the permutations that are one simple transposition away from becoming nonvexillary being less likely to be generated continues throughout the data. Other data entries follow the same trend.

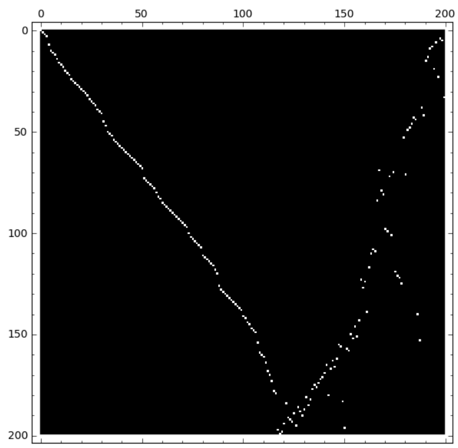
In general, it is diffiicult to define exactly the number of simple transpositions that are needed for the second force to ultimately win out in terms of which permutations are most likely to be generated. As the vexillary permutation with the longest reduced expression, the reverse permutation provides some rough insight as to where the second force will ultimately be more telling as to what the VMA will result in. The number of simple transpositions that a permutation is from becoming nonvexillary tends to become the most important factor in terms of the likelihood of generating a certain vexillary permutation using the VMA at some point shortly after the number of simple transpositions needed to get to the reverse permutation of the given length from the identity. In other words, the length of the reduced expression for the reverse permutation of a set length is the minimum number of simple transpositions for the second force to be more important than the first.

It is also very important to note that these patterns of the probability of generating a certain vexillary permutation using the VMA hold true for both an even and odd number of simple transpositions.

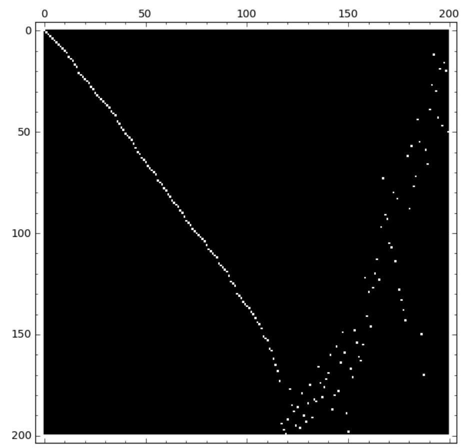
3. Vexillary Permutation Matrices

Using the Hook Walk Algorithm to generate a random reduced expression of a VMA produced vexillary permutation and examing its permutation matrix over time, it can be seen that the graphs have asymptotic bounds.

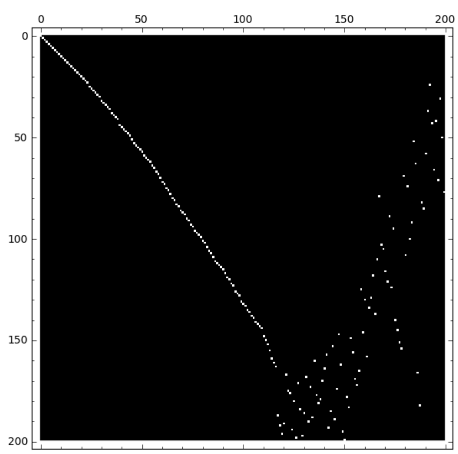
3.1. **VMA Produced Permutations.** Figure 10 depicts the change over time of the permutation matix generated of a length 200 vexillary permutation that underwent one million simple transpositions in the VMA.



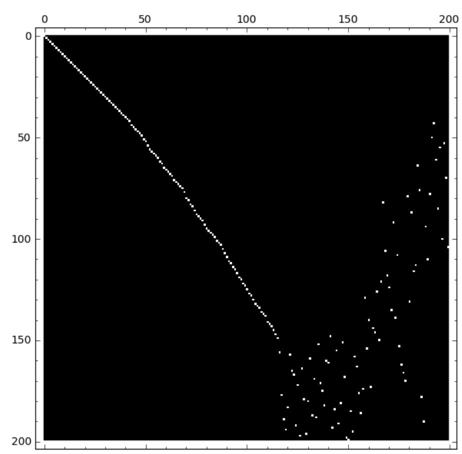
(A) $t = 1$



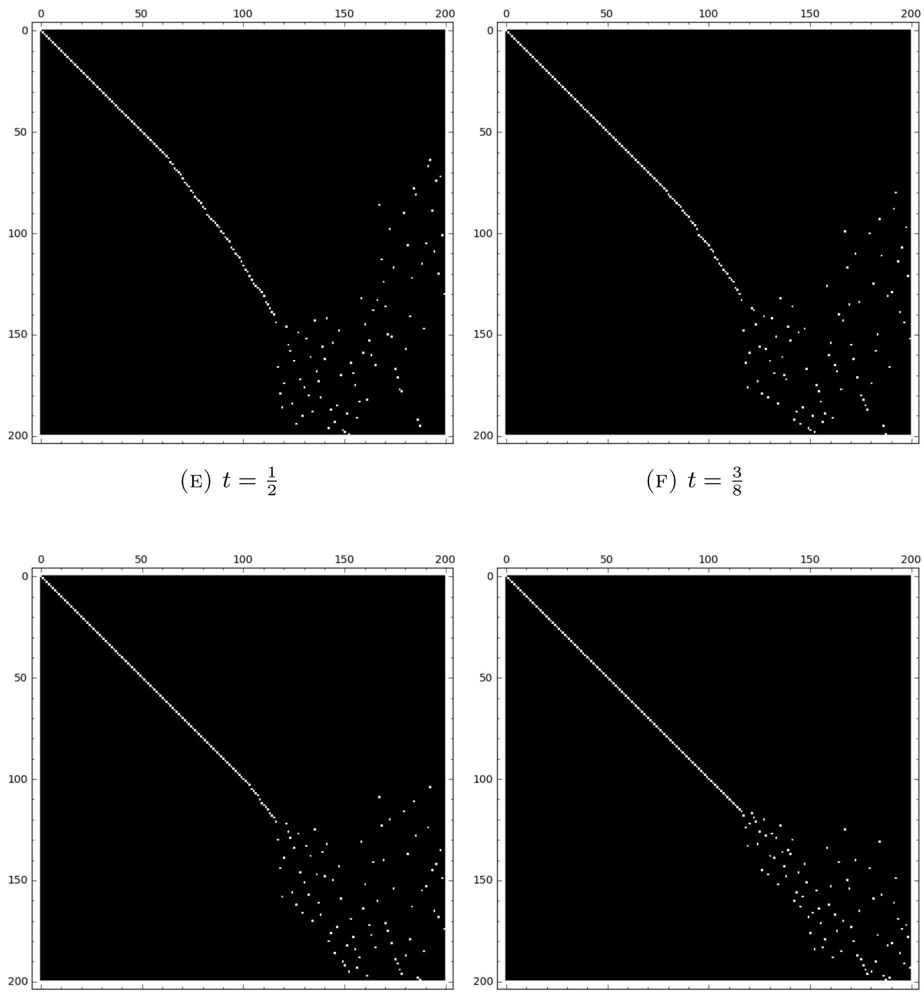
(B) $t = \frac{7}{8}$



(C) $t = \frac{3}{4}$



(D) $t = \frac{5}{8}$



(E) $t = \frac{1}{2}$ (F) $t = \frac{3}{8}$

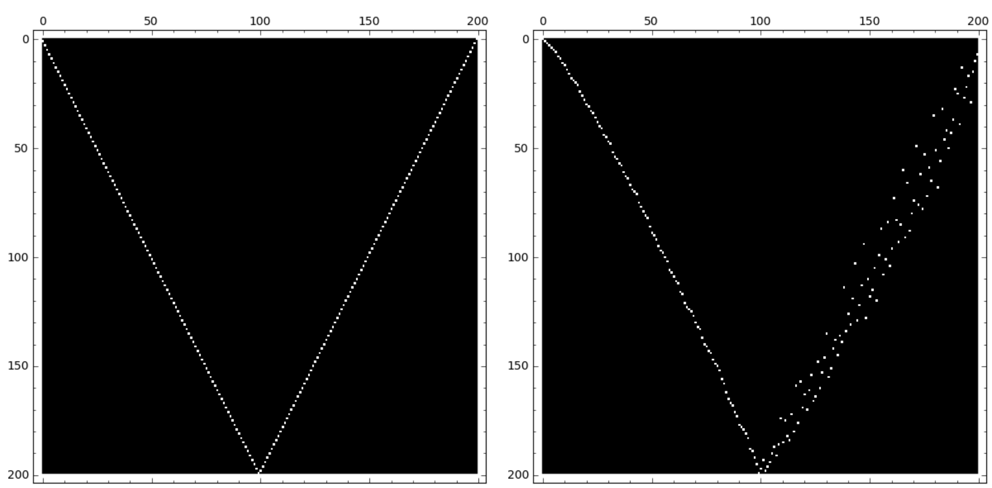
(G) $t = \frac{1}{4}$ (H) $t = \frac{1}{8}$

FIGURE 10

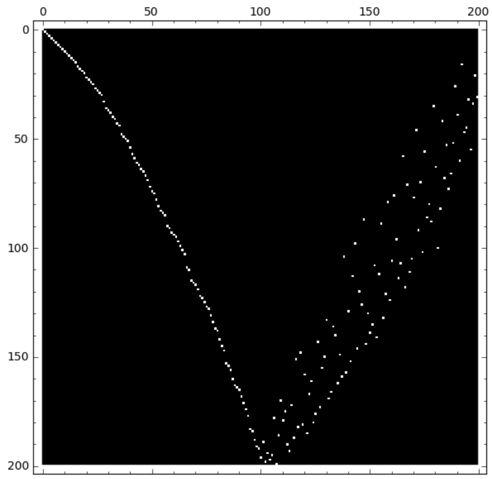
Though a portion of its permutation matrix over time is similar to the permutation matrices produced by Angel, Holroyd, Romik and Virág, there is something to be said about a vexillary permutation matrix at $t = 1$. Since vexillary is such a strict restriction on the order of a permutation, it is not surprising that this restriction has implications on its permutation matrix. For every vexillary permutation matrix there is a distinctive V-shape that occurs at time $t = 1$. This is due to the descending and ascending nature of the $2, 1, 4, 3$ vexillary pattern. The left arm of the V being made up of strictly ascending elements of the permutation, while the right arm of the V is made up of blocks of descending elements that can have nonadjacent elements increasing, as well as being made up of numerically larger elements that are essentially outlying points, having been knocked out of the blocks. These outlying points, to avoid creating a nonvexillary pattern, are in ascending order.

The point of the V can occur at any $k \in \rho$ when the ascending elements connect with the descending blocks of elements. When the permutation matrix descends through time the left arm of the V, bound by the line at time $t = 0$, begins to slowly curve and bend to that line. Also, conjecturing for the right arm of the V, the transformation of the permutation matrix is similar to the transformations put forth by Angel and Co., resembling elliptical and spherical shapes with outlying points. This is no surprise since the right arm of the V is composed largely of decreasing blocks, where as the reverse permutation is strictly decreasing.

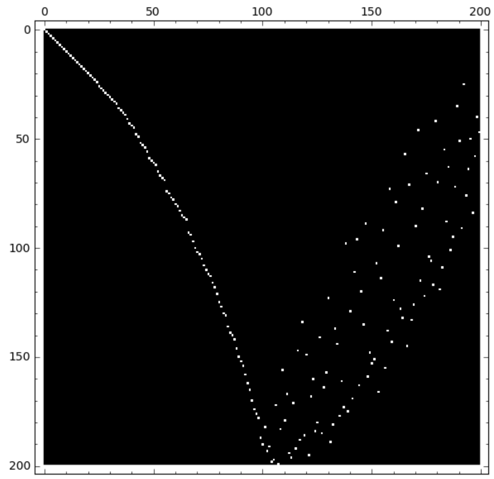
3.2. **Special Type of Vexillary Permutations.** An interesting vexillary permutation to examine is the one in which each element is a simple transposition away from an element it neighbors in the identity. For a given length n, this is the permutation $\rho = [n, 1, (n-1), 2, (n-2), 3 \ldots (\frac{n}{2}+1), \frac{n}{2}]$. When $n = 10$, this special type of vexillary permutation is $[10, 1, 9, 2, 8, 3, 7, 4, 6, 5]$.



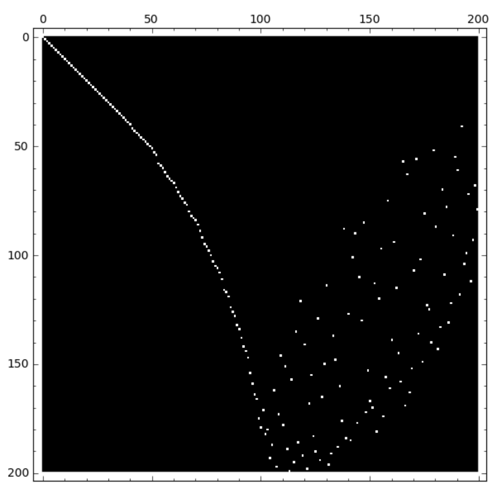
(A) $t = 1$ (B) $t = \frac{7}{8}$



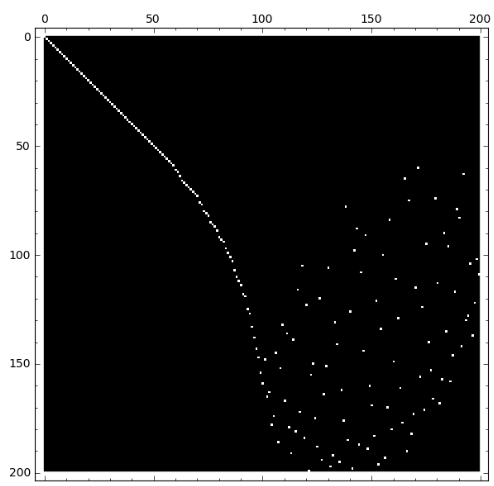
(c) $t = \frac{3}{4}$



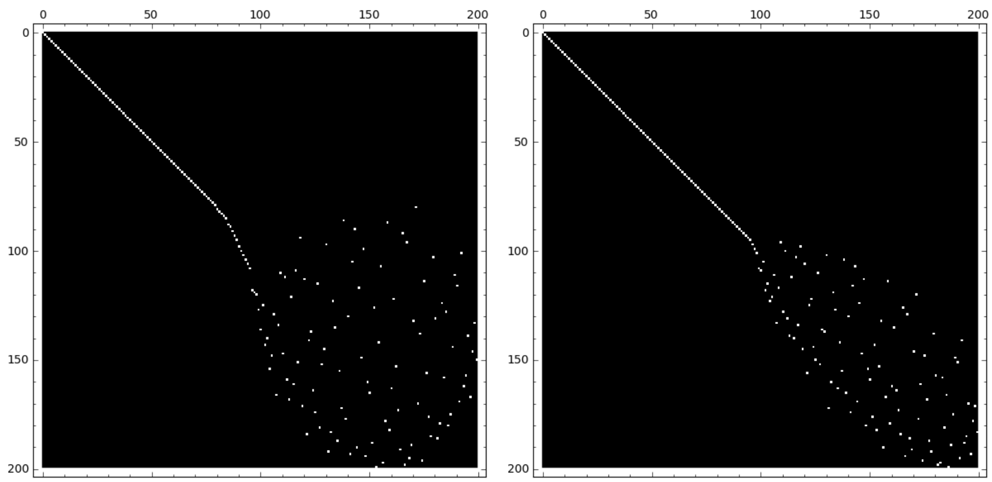
(d) $t = \frac{5}{8}$



(e) $t = \frac{1}{2}$



(f) $t = \frac{3}{8}$



(G) $t = \frac{1}{4}$

(H) $t = \frac{1}{8}$

Figure 11

The above visual shows the progression of this special type of permutation with length $n = 1000$ as the time of the reduced expression changes. At $t = 1$, the permutation matrix for the special type of vexillary permutation looks exactly like a V-shape. The values from 1 to $\frac{n}{2}$ are all in increasing order, which creates a downward sloping line on the left half of the permutation matrix. The values from $\frac{n}{2}$ to n are all in decreasing order, which creates an upward sloping line on the right half of the permutation matrix. As time decreases from $t = 1$ to $t = \frac{3}{4}$ the left portion follows the same pattern that was seen earlier in the randomized vexillary permutation. The right portion takes on an elliptic shape. At half time, the trend of the left portion of the matrix continues in the same manner as the previous case, yet the right portion takes on an elliptic shape with a greater radius than before on both axes. At $t = \frac{1}{4}$, the left portion of the graph continues as expected while the right portion takes on a spherical shape.

The progression of the permutation matrix for this special case of the vexillary permutation is quite interesting in its relation to the findings of Angel, Holroyd, Romik and Virág. The right portion of the matrix certainly seems to follow a similar progression as the authors found for the progression of the reverse permutation. The shapes that the right portion of the matrix have octagon bounds, as it grows from linear at $t = 0$ to spherical at $t = \frac{1}{4}$ to elliptical at $t = \frac{1}{2}$ and $t = \frac{3}{4}$ until finally becoming linear again at $t = 1$. Unlike the authors' findings, however, the right portion of the matrix of this special type of vexillary permutation takes on a spherical shape at $t = \frac{1}{4}$, whereas the reverse permutation appeared as a spherical shape at $t = \frac{1}{2}$.

4. The VMA Algorithm

```
def VMA(n,  transpositions):
    "generate a random vexillary of length n by the number of simple transpositions"

k = 1
sequence = []
```

```
while k <= n:
   sequence.append(k)
   k = k+1

r =1
while r <= transpositions:
   a = sequence.index(choice(sequence))
   while a == len(sequence)-1:
       a = sequence.index(choice(sequence))
   b = a+1
   r += 1

   sequence[a] , sequence[b] = sequence[b] , sequence[a]
   stop = 0


   for i in xrange(0,n):
      for j in xrange(i+1, n):
         if sequence[i] > sequence[j]:
           for h in xrange(j+1,n):
             if sequence[h] > sequence[i]+1:
               for l in xrange(h+1,n):
                 if not stop and sequence[l] > sequence[i] and sequence[l] < sequence[h]:
                     sequence[a] , sequence[b] = sequence[b] , sequence[a]
                     r -= 1
                     stop = 1
return sequence
```

References

[1] Omer Angel, Alexander E. Holroyd, Dan Romik, Bálint Virág. *Random Sorting Networks*, Advances in Mathematics, 2006.
[2] Paul Edelman, Curtis Greene. *Balanced Tableaux*, Advances in Mathematics, 1987.
[3] Curtis Greene, Albert Nijenhuis, Herbert S. Wilf. *A Probabilistic Proof of a Formula for the Number of Young Tableaux of a Given Shape*, Advances in Mathematics, 1979.
[4] Richard P. Stanley *On the Number of Reduced Decompositions of Elements of Coxeter Groups*, European Journal of Combinatorics, 1984.

University of Maryland
E-mail address: maxgross17@gmail.com

University of Oregon
E-mail address: marsagli@uoregon.edu