Card Shuffling

Peter Luthy

January 26, 2012

1 Does It Matter How You Shuffle Cards?

The focus of the 2008 movie, *21* [6], is on a group of students from MIT who go to casinos to play a card game called Blackjack[1]. In that game, there are two sides: the dealer and the player. In the game, every card has a point value; cards with numbers are worth their face value, jacks, queens, and kings are worth 10. Aces, at the discretion of the player, are worth either 1 or 11 points. In the process of the game, the player and the dealer are dealt cards, and they total their number of points. The goal is to have this total be as close to 21 as possible without exceeding 21. The dealer wins on ties (including when both the dealer and player have totals exceeding 21).

The game proceeds as follows: both the dealer and the player are dealt two cards; the player receives both cards face down but the dealer leaves one card face up and the other face down. The player may then request as many additional cards as he or she would like, one at a time. This process is called hitting, and when a player wants another card, he or she says, "hit me." The dealer must request cards until his or her total is at least 17, at which point he or she can no longer request any cards. After both the dealer and player have finished, they reveal their cards and compare totals. There are other rules to the game, however these are the basics [2, p. 430–431]. The exact rules can also vary slightly from casino to casino. For every dollar bet, a winning player receives 2 dollars (i.e. the payout is at a ratio of 2:1).

In 1956, Baldwin (and coauthors) published a paper, [2], which determined the optimal strategy for one version of Blackjack, assuming the cards were being dealt from a standard 52 card deck and the player had no additional knowledge about the deck. The conclusions of the paper are extremely counterintuitive. For instance, if the dealer's shown card is a 4, 5, or 6, one should never hit if one's total is 12 or more despite 12 being far from 21.

[1] *21* is an adaptation of the 2003 book, *Bringing Down the House*, by Ben Mezrich.

This is because the dealer is quite likely, in such a case, to go over 21. Another important result of the paper is that on a $1 bet, the player should expect to lose $0.006, even if playing optimally.

One key phrase in the preceding paragraph was, "the player had no additional knowledge about the deck." However, if one plays several rounds of Blackjack, one gets additional information: one knows which cards have been discarded and are thus out of play. In particular, after several rounds of betting, suppose that the player knows that the remaining cards in the deck are mostly of value-10 (including jacks, queens, and kings). In such a case, the risk of going over 21 is very high. The player can refuse to take any cards. The dealer, being required to finish with a total of at least 17, does not have this luxury and will be much more likely to lose in such a situation. Thus even knowing the composition of the deck, let alone the order of the cards, can be extremely important.

In the movie mentioned above, the MIT students take advantage of this fact. Their strategy is that one player sits at a Blackjack table making extremely small bets and keeps track of which cards have been discarded. When the deck has a high proportion of value-10 cards, the player signals to a nearby friend who then begins betting vast sums of money. Over the long term, such a strategy is extremely profitable.

The conclusion to be drawn from this example is that information about the deck is important. In the case of Blackjack, one gains an edge even if the knowledge is simply which cards remain (and not the order of the deck). In games like Texas Hold'Em (a type of poker), knowledge of the order of a few cards can give an edge: knowing that the ace of spades and ace of clubs were put into the deck one after another may induce a player to bet stronger or weaker if the ace of spades comes up (assuming they think the dealer did not shuffle well). There are a number of non-standard card games, including the popular *Magic: the Gathering*; in this particular game, one wishes to draw certain cards from the deck together, and so shuffling strategies which do not separate pairs of cards rather well can aid a player greatly. Another motivation from games like *Magic* is that the deck each player uses can vary in size, though typically a deck is between 40 and 60 cards. It is worth noting that there are professional *Magic* tournaments (the top tournaments have prize pools exceeding $200,000 as of 2010); despite the cash prizes at stake, the official tournament rules are extremely vague as to how a player needs to shuffle a deck, stating only that, "Decks must be randomized using some form of riffle and/or mash shuffle at the start of every game and whenever an instruction requires it. Randomization is defined as bringing the deck to a state where no player can have any information regarding the order or

position of cards in any portion of the deck."

As the official tournament rules of *Magic* imply, we shuffle so that the deck is well mixed, i.e. randomized. Our intuition says that shuffling the cards should prevent players from gaining knowledge about the deck. Casinos tend to agree. Casino Blackjack tables use a greater number of decks (generally 8) and many casinos use mechanical shuffling machines to shuffle after every hand.

After this discussion, we can agree that shuffling is an important part of card games. However, we are still left with a number of questions:

1. What do we mean when we say a deck is "well mixed" or "randomized?"
2. How do we measure how mixed or random a deck is?
3. Can we come up with mathematical models of shuffling techniques?
4. Do all shuffling techniques work equally well?

In what follows, we will attempt to answer these questions in detail; to do so, we will connect card shuffling to several major mathematical concepts. It is our hope that, through this discussion, the reader will gain an appreciation for how useful it is in mathematics to have several different points of view. Indeed, this is indirectly one of the main goals of this text.

2 Groups

In mathematics, objects with similar structures frequently come up in different contexts. Sometimes a collection of similar objects will be so common in mathematics that mathematicians will begin to study the properties of these objects in the abstract. This is a vague explanation, but this is likely a factor of any abstract discussion of another abstract discussion. So, let's move to a concrete collection of examples.

2.1 Abstract Groups

When people talk about numbers, they may mean completely different things. Do they mean the number is an integer like 17? A fraction like $\frac{2}{3}$? A real number like π? A complex number like $2 + 2i$? Despite their fundamental differences, these numbers have several similar properties:

1. Each number system is a set.

2. Each has a binary operation. All this means is that one can take two numbers and combine them to get another number. In all four cases, this operation is addition.

3. Each set has an identity element (labeled 0 in all 4 cases), which satisfies $0 + x = x = x + 0$.

4. Each element a has an inverse element, $-a$, so that $a + (-a) = 0$.

5. The operation satisfies the Associative Law: $(a + b) + c = a + (b + c)$

6. The operation satisfies the Commutative Law: $a + b = b + a$.

There are many other examples of objects which satisfy these conditions:

- The set of all real (or rational or complex) numbers except zero with the operation being multiplication. Here the identity element will be 1 and the inverse element for a would be $\frac{1}{a}$.

- The set of all real-valued functions on the unit interval, $[0, 1]$, which do not take zero as a value (for example $1 + x^2$ and 2^x). The operation is multiplication.

- 2×2 matrices with addition, e.g. $\begin{pmatrix} 2 & 2 \\ 3 & 4 \end{pmatrix} + \begin{pmatrix} 2 & -1 \\ 0 & 3 \end{pmatrix} = \begin{pmatrix} 4 & 1 \\ 3 & 7 \end{pmatrix}$. Here the identity element is the matrix with all zeros as entries and to construct the inverse of a matrix, one just changes the sign of every matrix entry.

There are also many examples of objects which have all these properties except they do not satisfy the Commutative Law:

- A geometric example comes from symmetries of regular polygons. Take an equilateral triangle, for example. The triangle has 3 rotation symmetries (rotations by 0°, 120°, 240°) and 3 reflection symmetries (through each of the angle bisectors). The identity element is the rotation by 0°. Each reflection is its own inverse, and each rotation has an inverse (which is itself a rotation). However, it's easy to check (by working with a piece of paper and marking the vertices) that reflections and rotations don't always commute.

- The set of 2×2 matrices under multiplication. The identity element in this case is $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. There is a problem right off the

bat because not every matrix has a multiplicative inverse. For example, the matrix $\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$ has no multiplicative inverse because $\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$ which is not the identity element. However, many matrices do have inverses. One can consider the set of all matrices with multiplicative inverses under the operation of multiplication. To reiterate, the identity element is $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. One can check that all the properties discussed above are satisfied except that it is easy to come up with matrices A and B so that $AB \neq BA$. For example, if you were to pick numbers randomly for all the entries of two matrices, it is extremely likely that they will not commute.

From these examples, one is motivated to make the following definition:

Definition 1. A **Group** is a set equipped with a binary operation with the following properties:

1. The operation obeys the Associative Law.
2. The set contains an identity element with respect to the operation.
3. Every element has an inverse element with respect to the operation.

It is worth noting that the Commutative Law is not listed above. Groups that obey the Commutative Law are called Abelian Groups (after the Norwegian mathematician Niels Henrik Abel). The notion of the group is one of the cornerstones of mathematics; they are present in nearly any mathematical discipline.

2.2 The Symmetric Groups

Card shuffling can be related to groups through a special class of groups called symmetric groups. Rather than start with an abstract definition of the symmetric groups, we will begin with a very simple example. Consider three objects which we will call A, B, and C (short for Apple, Banana, and Carrot). Suppose further we have three boxes labeled 1, 2, and 3. To start with, we'll assume that A is in box 1, B is in box 2, and C is in box 3. A permutation of the objects A, B, C will be a way of putting the objects in different boxes with exactly one element in each box. There are six permutations:

1. The permutation which does not do anything. We will denote this permutation by ε. The permutation ε leaves A in box 1, B in box 2, and C in box 3. This permutation takes the order ABC to ABC.

2. The permutation which moves A to box 2 and B to box 1. We will denote this permutation by $\begin{pmatrix} 1 & 2 \end{pmatrix}$. This permutation takes the order ABC to the order BAC.

3. The permutation which moves A to box 3 and C to box 1. We will denote this permutation by $\begin{pmatrix} 1 & 3 \end{pmatrix}$. This permutation takes the order ABC to the order CBA.

4. The permutation which moves B to box 3 and C to box 2. We will denote this permutation by $\begin{pmatrix} 2 & 3 \end{pmatrix}$. This permutation takes the order ABC to the order ACB.

5. The permutation which moves A to box 2, B to box 3, and C to box 1. We will denote this permutation by $\begin{pmatrix} 1 & 2 & 3 \end{pmatrix}$. This permutation takes the order ABC to the order CAB.

6. The permutation which moves A to box 3, B to box 1, and C to box 2. We will denote this permutation by $\begin{pmatrix} 1 & 3 & 2 \end{pmatrix}$. This permutation takes the order ABC to the order BCA.

Visually, one should look at the notation $\begin{pmatrix} 1 & 3 & 2 \end{pmatrix}$ and read the numbers left to right as follows: "What's in box 1 goes to box 3, what's in box 3 goes into box 2, and what's in box 2 wraps around to box 1."

We define a binary operation on the permutations (for ease of use, we will call the operation multiplication). The multiplication,

$$\begin{pmatrix} 1 & 2 \end{pmatrix} \begin{pmatrix} 1 & 3 & 2 \end{pmatrix}$$

means apply the permutation $\begin{pmatrix} 1 & 3 & 2 \end{pmatrix}$ and then the permutation $\begin{pmatrix} 1 & 2 \end{pmatrix}$. Note that this means you apply the right permutation first and then move to the left (even though we read in the opposite way). In terms of how this changes the order of ABC, the permutation $\begin{pmatrix} 1 & 3 & 2 \end{pmatrix}$ changes the order to BCA so that B is in box 1, C is in box 2, and A is in box 3; then $\begin{pmatrix} 1 & 2 \end{pmatrix}$ interchanges the first and second boxes to produce CBA. So, we obtain

$$\begin{pmatrix} 1 & 2 \end{pmatrix} \begin{pmatrix} 1 & 3 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 3 \end{pmatrix}.$$

It is important to be sure you read the permutations in the correct order since

$$\begin{pmatrix} 1 & 3 & 2 \end{pmatrix} \begin{pmatrix} 1 & 2 \end{pmatrix} = \begin{pmatrix} 2 & 3 \end{pmatrix}.$$

To be completely rigorous, one should check that the operation is associative, but we won't (an extremely interested reader should feel free to check this on his or her own). The permutation ε is the identity element. Every permutation which only permutes two elements is its own inverse, and the two permutations which permute three elements are inverses of one another. So indeed, this set of permutations is indeed a group. This group is called the symmetric group on three elements and is denoted by S_3.

One can view this group as the symmetries of the equilateral triangle described in the previous section. Here one labels the vertices of the triangle as 1, 2, and 3. The permutations, $\begin{pmatrix} 1 & 2 \end{pmatrix}$, $\begin{pmatrix} 1 & 3 \end{pmatrix}$, and $\begin{pmatrix} 2 & 3 \end{pmatrix}$ correspond to reflections through the angle bisector of vertex 3, 2, and 1, respectively; the other permutations are rotations. Which is the rotation by 120 degrees? You should also check that the multiplication in S_3 and the composition of symmetries work in precisely the same way.

With a little imagination, one can construct the group S_4 or, more generally, S_n for $n > 0$ by adding more elements and more boxes, i.e. look at permutations on n elements in essentially the same way as we did for 3 elements. The groups S_n are among the most widely studied groups for a variety of reasons we will not get into, however the group S_{52} is of particular interest to us now since a typical deck of cards has 52 cards. It is worth noting that S_n is not an Abelian group when $n \geq 3$.

The curious reader may wonder if S_4 corresponds to the group of rotation and reflection symmetries of the square just like S_3 corresponded to the symmetry group of the equilateral triangle. This is not the case. To understand why, simply consider the number of elements in each group.

2.3 Card Shuffles, S_{52}, and Computational Issues

A standard deck has 52 cards. Each element of the group S_{52} corresponds to a permutation of 52 objects. Thinking of those objects as cards, each element of S_{52} determines a way to rearrange the cards. So, a permutation in S_{52} gives us a "shuffle." A random element of S_{52} will not give us a rearrangement which corresponds to what we usually think of as a shuffle. For example, the element $\begin{pmatrix} 1 & 52 \end{pmatrix}$ just interchanges the top and bottom cards in the deck; the element $\begin{pmatrix} 1 & 2 & 52 \end{pmatrix}$ moves the top card to the second-to-top position, the second card to the bottom, and the bottom card to the top. These are hardly what we usually think of as shuffling. So each element of S_{52} is just some way of rearranging a deck of cards. However, this is what we will refer to as a general shuffle. Later on, we will discuss more in depth how to relate elements of S_{52} to what we usually think of as "real" shuffles.

One might consider just writing a computer program to analyze various shuffling methods. Since we are generally only interested in the practical applications of the information we determine, we don't even really need any exact information anyway. However, there are some serious drawbacks to studying the group S_{52} with a computer. To begin with, we should determine how many permutations there are in S_n for any $n > 0$. Each permutation assigns exactly one element of n elements to exactly one of n boxes, and there is one permutation for every possible way to put elements in boxes. How many ways are there to put a single element into n boxes? Certainly there are exactly n ways to do that. What about when we put 2 elements into n boxes? We have n places to put the first element but only $n-1$ spots remain for the second element since one box is already occupied. For each of our choices for the first element, there are $n-1$ choices for the second. This means that there are $n \times (n-1)$ ways to place 2 objects into n boxes. Carrying this reasoning forward, there are $n \times (n-1) \times (n-2)$ ways to put 3 elements into n boxes. So, to put n elements in n boxes, there are $n \times (n-1) \times (n-2) \times ... \times 3 \times 2 \times 1$ ways to do so. This number comes up frequently in mathematics and is denoted by "$n!$" or "n factorial." The number $n!$ grows very quickly:

n	$n!$
1	1
2	2
3	6
4	24
5	120
...	...
52	roughly 8×10^{67}
...	...
very large n	roughly $\sqrt{2\pi n} \times \left(\frac{n}{e}\right)^n$

The last row above is called Stirling's approximation. The function n^n grows extraordinarily quickly.

The number 52! is bigger than 8×10^{67}. The fastest super computer on Earth, as of 2008, can perform less than 1.16×10^{15} operations per second. So, to even list the elements of the group S_{52} would take the fastest computer in the world more than approximately 6.8×10^{52} seconds. That's more than 2×10^{45} years, or more than $1,000,000,000,000,000,000,000,000,000,000,000$ times what physicists generally agree is the age of the universe. The number of zeros in that number is more than enough to make everyone agree this is

longer than most people would be willing to wait to figure out if their friend is cheating during a card game. Another counterintuitive conclusion of this analysis is that if one could write down the exact shuffles performed in the collective history of mankind, one would not come even remotely close to all possible shuffles.

While the impossibility of computerizing the problem may be disappointing, there is an upside: human ingenuity is not obsolete. It may seem daunting at this point, but we will produce some rather precise numerical results without having to use a computer.

2.4 Generating Sets

As stated previously, most of what we call mathematical shuffles do not correspond to actual shuffles. So, rather than working with the whole group S_{52}, let us pick a smaller subset of our group elements. We're going to think of this subset as our admissible shuffles. As we will discuss below, our choice in our subset more or less determines our shuffling technique (and there are several different standard techniques). Taking products of these admissible shuffles is like applying one shuffle and then another. Any reasonable shuffling technique should be able to produce an arbitrary shuffle by taking a product of (possibly very many) admissible shuffles. In other words, if we pick an arbitrary permutation σ in S_{52}, we should be able to take some of the permutations from our subset, multiply them together, and get σ as the result.

When products of elements of a subset of a group can produce any element of the group, we say the subset **generates** the group. In what follows, we will give an example of a small subset of the permutations which generates all of S_n.

In S_n, a **transposition** is a permutation which interchanges two positions but leaves everything else fixed. Such an element looks like $\begin{pmatrix} 1 & n-1 \end{pmatrix}$ or $\begin{pmatrix} 2 & 13 \end{pmatrix}$ which are the elements which interchange the cards in the first and $(n-1)$th positions and the cards in the second and thirteenth positions, respectively. We will allow that the identity element also counts as a transposition.

A **cycle** is a more general version of a transposition. It is a permutation which looks like $\begin{pmatrix} 1 & 3 & n & 37 & n-7 \end{pmatrix}$. This permutation moves the first card to the third position, the third card to the last position (or "n^{th} position"), the last card to the thirty-seventh position,the thirty-seventh card to the $(n-7)$th position and the $(n-7)$th card to the top, which is kind of like a loop (hence the name cycle). Cycles are said to be disjoint if they

permute completely different elements, e.g. $\begin{pmatrix} 1 & 2 & 3 & 4 \end{pmatrix}$ and $\begin{pmatrix} 5 & 7 & 9 \end{pmatrix}$ are disjoint cycles.

Theorem 2. *Every permutation can be written as the product of disjoint cycles*

Proof. It is easy to see that disjoint cycles commute with one another. For example,

$$\begin{pmatrix} 1 & n-1 \end{pmatrix}\begin{pmatrix} 2 & 13 \end{pmatrix} = \begin{pmatrix} 2 & 13 \end{pmatrix}\begin{pmatrix} 1 & n-1 \end{pmatrix}.$$

Visually, one could imagine laying n cards side by side; since the two cycles don't include any of the same cards, one can move the cards in one cycle up a little bit and the cards in the other cycle down a little bit; it then doesn't matter whether you move the top cards around and then the bottom cards or the bottom cards first and the top cards second. It is also the case that any permutation can be written as the product of disjoint cycles. This is also easy to check. Because disjoint cycles commute, it suffices to show that each card which is affected by the permutation is part of a cycle. For ease of discussion, suppose we are working with a deck of cards so that the group in question is S_{52}. If a card, say the A♠ is moved by a permutation, then the A♠ takes one card's place, say the 10♡, and another card, say the 4♣, takes the original place of A♠. Likewise, some card must move to occupy the original position of the 4♣ and the 10♡ must move so that the A♠ can move in. Since there are only finitely many cards in the deck, this procedure cannot be continued forever. Thus eventually the cycle must close itself. □

Remark: The above theorem says that cycles generate S_n.

Theorem 3. *Transpositions generate S_n*

Proof. Since every permutation can be written as the product of disjoint cycles, it suffices to prove that every cycle can be written as the product of transpositions. In other words, if we can generate any cycle by taking products of transpositions, we can generate every permutation since the cycles generate S_n. But there is an easy way to decompose cycles into special transpositions, which we demonstrate for a particular cycle:

$$\begin{pmatrix} 1 & 2 & 4 & 37 & n-7 \end{pmatrix} = \begin{pmatrix} 1 & n-7 \end{pmatrix}\begin{pmatrix} 1 & 37 \end{pmatrix}\begin{pmatrix} 1 & 4 \end{pmatrix}\begin{pmatrix} 1 & 2 \end{pmatrix}$$

□

The fact that transpositions generate S_n will be important later on. Also, observe that even though S_n generally has a huge number of permutations, there are only $1 + \frac{n(n-1)}{2}$ transpositions (we have to add one because the identity counts). So if one shuffled the deck by randomly picking a transposition, it should seemingly take a large number of shuffles to produce an arbitrary permutation.

2.5 Shuffling Strategy

At this point, we have established that S_n consists of all the possible ways to shuffle a deck of n cards. Multiplying two elements of S_n corresponds to shuffling the deck according to the first shuffle and then the second shuffle. We have also made the remark that many of the elements of S_n don't correspond to our usual notion of shuffling. So, rather than focusing our attention on all of S_n, let's choose a subset T of S_n which generates S_n. This set T consists of our basic shuffles. In this level of generality T could consist of many elements (S_n is of course technically a generating set) or it could consist of very few elements (the transpositions, for example). This set T will define the basic shuffles in the model of our shuffling strategy.

Examples of shuffling strategies:

1. The set of transpositions generates S_n, so we may choose T to consist of all transpositions. While this shuffling strategy seems like it should shuffle the deck very slowly, its simple nature is helpful as an example to keep in mind.

2. The overhand shuffle is produced by cutting the deck into several different pieces. Both the number and size of each piece can vary. The order of each piece is then reversed while maintaining the order of the cards in each piece. For example, if the deck is split into four pieces A, B, C, and D of sizes 13, 20, 6, and 13, respectively, then the order of the deck after the shuffle would be D, C, B, A; the order of all 13 cards in A would remain the same and likewise for the cards in pieces B, C, and D. The key to seeing that overhand shuffles generate S_n is showing that one can apply a few overhand shuffles to interchange the first and second card. This is done for n cards as follows (it is easiest to understand how this works by working with an actual deck of cards):

    - Perform the overhand shuffle with pieces of sizes 2 and $n-2$ (in that order). This just puts the top two cards on the bottom.

Now perform the overhand shuffle with pieces of sizes $n-2$, 1, and 1, in that order. The deck is now in the same order except the top two cards have been interchanged.

Now that one can interchange the first two cards, one can perform overhand shuffles to interchange the first and third card:

- Interchange the top two cards.
- Put the top card on the bottom.
- Interchange the top two cards.
- Put the bottom card on top.
- Interchange the top two cards.

We know by the proof of Theorem 3 that transpositions of the form $\begin{pmatrix} 1 & k \end{pmatrix}$ for $1 \leq k \leq n$ generate S_n for any n. So, we know that by what we've just done that we can arrange the first three cards in whatever order we want. To switch the first and fourth cards,

- Put the top card on the bottom.
- Put the third card on top.
- Put the bottom card on top.
- Interchange the first and second cards.
- Put the top card on bottom.
- Put the top card in third position.
- Put the bottom card on top.

Now we can get the first four cards in any order. Applying the same method iteratively allows us to put the deck into any order using only repeated overhand shuffles (but it takes a long time).

3. The standard shuffle, called the riffle shuffle is a bit more complicated to describe. This shuffle is described in greater detail in Section 6.1, but essentially involves cutting the deck into two pieces (not necessarily equal in size) and putting one piece in the left hand and one piece in the right hand; then one forms a pile of cards by iteratively dropping a few cards (this could be any number of cards) from one hand, then a few cards from the other hand, and so on until all the cards have been dropped. All the shuffles used to show the overhand shuffles generate S_n are also technically basic riffle shuffles, so the riffle shuffles also generate S_n.

It is important that T generate S_n because we want to know that, after many shuffles, the deck could be in any order. If T did not generate S_n, there would be some σ in S_n which could not be written as a product of elements of T. So, there would be orderings of the deck we would never reach, no matter how many basic shuffles from T we apply.

3 Graphs

3.1 Graphs in the Abstract

Typically, when one thinks of a graph, one imagines a curve representing the value of a particular stock, a pie chart detailing the proportion of profit-loss caused by various sources, or a histogram which describes student grades on an exam. However, mathematically speaking, a graph is much simpler. If, in what follows, you get the idea of something resembling connect-the-dots, you're thinking in the right direction.

A mathematical graph consists of two sets: V, called the vertex set, and E called the edge set. The reason E is called the edge set is because it is made up of one- or two-element (unordered) subsets of V. So, elements e of E can be visualized as lines between the corresponding vertices of e or loops starting and ending at the same vertex if e consists of a single element. One can easily draw a picture to represent a graph. One simply draws points on a piece of paper and labels each point to correspond with the elements of V. Then for each edge in E, draw a line connecting the two elements of V which are the vertices corresponding to that edge. We will also require that one never has two lines going between the same two vertices, and one is allowed no more than one loop which starts and ends at the same vertex. If two vertices in a graph are labeled a and b, we would typically call the edge either ab or ba (note that both of these labels would correspond to the same edge). There will always be at most one edge between vertices, and sometimes vertices will have no edges at all.

Example 4. The Complete Graphs K_4 and K_6

Suppose that V is comprised of four elements and the edge set E is comprised of every pair of distinct vertices. This particular graph is called K_4 and looks like the image shown in Figure 1. If we had defined graphs to disallow loops, this graph would have every possible edge present. The graphs which have every possible edge except for loops are called complete. K_n denotes the complete graph on n vertices. K_6 is shown in Figure 2.

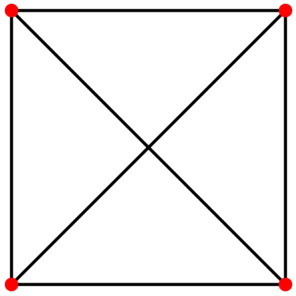


Figure 1: A visualization of K_4, image is public domain (from Wikipedia).

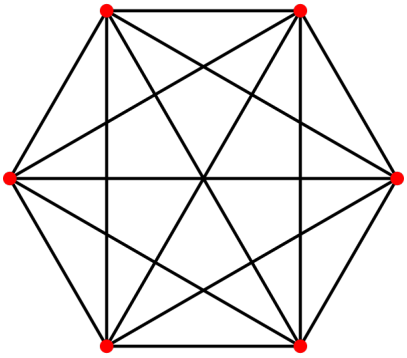


Figure 2: A visualization of K_6, image is public domain (from Wikipedia).

□

Definition 5. A **directed graph** or **digraph** consists of a set of vertices, V, and a directed edge set, E, which corresponds to ordered pairs of vertices — in other words, the edges in E have a direction.

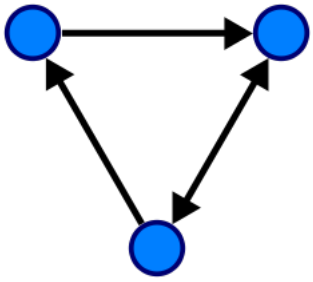
Often, mathematicians do not make an explicit distinction between graphs and digraphs, provided it is clear in context whether the object in question is a graph or digraph. And, truly, every graph can be represented by a digraph — to construct a digraph corresponding to a given graph, let the vertex set of the digraph be the same as the vertex set for the graph. Then

for any edge between two vertices in the graph, produce a pair of edges with opposite directions in the digraph between the same pair of vertices. So, we may at times refer to a digraph as just a graph, assuming it is clear from context that the edges have directions.

In a digraph, an edge between the vertices a and b is written ab or ba, but these labels would correspond to distinct edges in the graph. We will interpret the label ab to mean the edge goes from a to b and ba to mean the edge goes from b to a. When we draw a digraph, we draw an arrow on each edge to indicate which direction it goes in. If there are edges going in each direction, we would either draw two edges pointing in opposite directions or one edge with arrows pointing in both directions.

Example 6. A Directed Graph

Consider the vertex set V comprised of three vertices $\{a, b, c\}$. The graph has four edges: $\{ab, ca, bc, cb\}$. This digraph would look like (without vertex labels)



Exercise: Try labeling the vertices (there is only one possible way).

□

For (di)graphs, one can produce a matrix, called the **adjacency matrix**. For ease of exposition, label all the vertices with a number. In the adjacency matrix, each row corresponds to a vertex and each column corresponds to a vertex. We put a 1 in the entry corresponding to the kth row and mth column if there is an edge going from the kth vertex to the mth vertex. If there is not such an edge, put in a zero. In the preceding figure, if we label the top left vertex 1, the top right vertex 2, and the bottom vertex 3, the corresponding adjacency matrix is

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}.$$

For K_4, the adjacency matrix is

$$\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

This matrix will be useful later on. For now, we shall simply ask the reader to stop and consider some questions.

Exercise: First, what special property do adjacency matrices for graphs have that the adjacency matrices for digraphs do not have? Second, if one multiplies the adjacency matrix by itself, one gets a new matrix, although the entries are now not guaranteed to be one or zero: what do these numbers represent? Third, what is special about adjacency matrices for graphs without loops?

3.2 Walks on Groups

Now that we have the notions of graph, group, and generating set, we can construct a graph which is useful. In general, one would construct digraphs, but for our purposes, ordinary graphs will suffice. After we work through our main result, it will be more clear to the reader how digraphs would enter the picture. Also, for now we will stick to the case where our group is S_n since that's what we're really interested in, but what follows can be applied to any group. Recall the definition of the generating set T from Section 2.5. For the present we'll focus on the subset T of S_n which contains all the transpositions (recall that the identity permutation is in T).

To construct our graph, we will define our vertex set V to consist of every possible ordering of our deck of n cards. So, there will be $n!$ points, and we label all the vertices accordingly, although the particular labeling is not important. We will draw an edge between two vertices if the orderings corresponding to these vertices differ by a transposition. Since the set T generates S_n, the graph produced this way will be **connected**. Intuitively, this means that if one built a physical model of the graph, one could lift the entire graph off the ground by lifting any vertex. The more technical definition is that between any pair of vertices there is a sequence of edges so that consecutive edges share a vertex (this sequence is called a path). Since the identity element is in T, every vertex will have a loop. We will call this the graph of S_n generated by T.

Example 7. The Graph of S_3 Generated by Transpositions

In this case, $T = \{\varepsilon, \begin{pmatrix} 1 & 2 \end{pmatrix}, \begin{pmatrix} 1 & 3 \end{pmatrix}, \begin{pmatrix} 2 & 3 \end{pmatrix}\}$. So, every vertex should have three outgoing edges and one loop. Putting the edges in properly, one produces the graph in Figure 3.

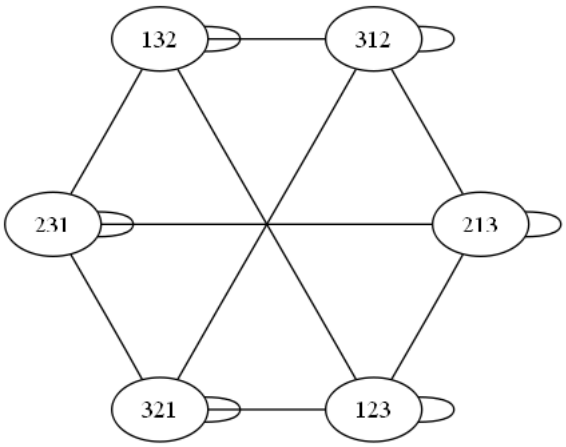


Figure 3: A graph of S_3

□

This graph will provide us with a way to visually interpret how our shuffling technique T shuffles the deck. We begin with the deck in a particular ordering. This corresponds to a particular vertex in our graph. Shuffling the deck once corresponds to picking an element of T which, in turn, corresponds to picking an edge which has our starting vertex as one of its endpoints and moving along that edge. If T contains the identity element, this "move" might actually correspond to staying at the same vertex. Shuffling the deck again corresponds to picking another element of T (which could possibly be the same edge we started with) and thus we end up at at some vertex. In this way, repeated shuffles generate a path through our graph. We will call this path a **walk** on our group. Repeatedly shuffling the deck according to T corresponds to walking along a path in our graph.

In and of itself, this notion may not seem that useful. However, this framework allows us to translate the problem of analyzing shuffles to studying what are called finite Markov chains.

4 Finite Markov Chains

4.1 What is a Finite Markov Chain?

Imagine you are asked to babysit for a toddler who lives in a house with six rooms. This particular toddler does not like staying in the same place for very long; in fact she has numbered the rooms in the house, and her favorite game is to roll a standard six-sided die every minute or so and then run to the room with the corresponding number, re-rolling if the result would have her stay in the room she's in. Since you find this game rather boring, you keep track of the order of the rooms you've been in as you move from room to room.

This is a very simple example of a finite Markov chain. Here chain means something like "chain of events", which corresponds to moving from room to room. Finite refers to the number of **states**: in this case being in a particular room is a state, and there are only a finite number of rooms (six, in fact). **Markov** is a term named after Andrei Markov, a Russian mathematician who lived in the late-nineteenth and early-twentieth centuries. The chain in the example above is Markov because, to figure out which room to go to next, you only need to know which room you are in now rather than all the rooms you've been in before. This chain would not be Markov if, for example, the toddler also would re-roll the die if it would take her back to the room she just left.

To reiterate, a finite Markov chain is a finite collection of states along with a set of rules which dictates how to move from state to state with the property that the rules only require knowing where you are rather than where you've been. These rules might not be deterministic in the sense that you might only know the probability of going from state to state — it may even be possible to stay in the same state with some probability. A completely deterministic set of rules is perfectly acceptable but not particularly interesting for our purposes.

We will now give a more detailed example to motivate our discussion as well as connect the notion of a finite Markov chain to computations involving matrices.

Example 8. A Simple Game of Tag

Suppose that three children, Alex, Ben, and Cecile are playing a game of tag. There are then three states: Alex is it, Ben is it, or Cecile is it. We'll say that the game has advanced one time step every time someone gets tagged. For ease of discussion, assume that all three children are equally good at tag so that whoever is it tags someone else with equal likelihood. This of

course means no one gets stuck being it and everyone should spend about the same amount of time being it over the course of the game. Then we can make the following statements: if Alex is it, then in the next state he will not be it and Cecile will be it with probability 0.5 (i.e. half the time) and Ben will be it with probability 0.5 (again, half the time). We can arrange this information in a matrix. Each row corresponds to who is it and reading the entries in that row tells us the probability that Alex will be it next, Ben will be it next, and Cecile will be it next, assuming that the person associated to that row is currently it. We'll write Alex $\to$ Cecile, to mean the probability of Cecile being it next assuming that Alex is it right now, i.e. the probability that Alex tags Cecile in this round given that Alex is it right now. The matrix then corresponds to

$$\begin{pmatrix} \text{Alex} \to \text{Alex} & \text{Alex} \to \text{Ben} & \text{Alex} \to \text{Cecile} \\ \text{Ben} \to \text{Alex} & \text{Ben} \to \text{Ben} & \text{Ben} \to \text{Cecile} \\ \text{Cecile} \to \text{Alex} & \text{Cecile} \to \text{Ben} & \text{Cecile} \to \text{Cecile} \end{pmatrix}$$

Putting the numbers into the matrix (and labeling the columns and rows for emphasis):

$$\begin{pmatrix} & \text{Alex will be it} & \text{Ben will be it} & \text{Cecile will be it} \\ \text{Alex is it} & 0 & 0.5 & 0.5 \\ \text{Ben is it} & 0.5 & 0 & 0.5 \\ \text{Cecile is it} & 0.5 & 0.5 & 0 \end{pmatrix}$$

Note that each row adds up to 1.

This situation is another example of a finite Markov chain, and this matrix contains all the information we have about the chain. We can think of the different states as vectors. The vector corresponding to the state Alex is it would be $(1, 0, 0)$. The vectors for Ben being it and Cecile being it are $(0, 1, 0)$ and $(0, 0, 1)$, respectively. The vector $(1/3, 1/3, 1/3)$ would correspond to starting the game out by randomly choosing one of the children to be it first. In general, we could pick any vector whose entries add up to 1. The entries of such a vector would correspond to the probability of starting the game of tag out with the respective child being it. In general, such a vector is called a **distribution** on the states of the Markov chain.

Suppose we start the game out with the vector $(0.5, 0.5, 0)$. This means Cecile does not begin being it, and we flip a coin to determine if Alex or Ben starts out being it. What is the probability of each child being it after one round of tag? How do we compute such a thing? The probability that

Alex winds up being it is

$$\begin{aligned} & \text{(probability Alex starts it)} \times \text{(probability Alex tags himself)} \\ + \;& \text{(probability Ben starts it)} \times \text{(probability Ben tags Alex)} \\ + \;& \text{(probability Cecile starts it)} \times \text{(probability Cecile tags Alex)} \\ = \;& (0.5)(0) + (0.5)(0.5) + (0)(0.5) \\ = \;& 0.25 \end{aligned}$$

We would perform a similar calculation to determine that the probability that Ben ends up being it is 0.25 and the probability that Cecile ends up being it is 0.5. The interesting thing to notice here is that all these calculations correspond to multiplying the vector $(0.5, 0.5, 0)$ by our matrix:

$$\begin{pmatrix} 0.5 & 0.5 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0.5 & 0.5 \\ 0.5 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{pmatrix} = \begin{pmatrix} 0.25 & 0.25 & 0.5 \end{pmatrix}$$

This is not a coincidence at all. In fact, if one is very careful, one can see that the probabilities of each person being it after two rounds of tag is given by

$$\begin{aligned} & \begin{pmatrix} 0.5 & 0.5 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0.5 & 0.5 \\ 0.5 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0.5 & 0.5 \\ 0.5 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{pmatrix} \\ = \;& \begin{pmatrix} 0.5 & 0.5 & 0 \end{pmatrix} \begin{pmatrix} 0.5 & 0.25 & 0.25 \\ 0.25 & 0.5 & 0.25 \\ 0.25 & 0.25 & 0.5 \end{pmatrix} \\ = \;& \begin{pmatrix} 0.375 & 0.375 & 0.25 \end{pmatrix} \end{aligned}$$

For n rounds of tag, this is

$$\begin{pmatrix} 0.5 & 0.5 & 0 \end{pmatrix} \underbrace{\begin{pmatrix} 0 & 0.5 & 0.5 \\ 0.5 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{pmatrix} \times \ldots \times \begin{pmatrix} 0 & 0.5 & 0.5 \\ 0.5 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{pmatrix}}_{n \text{ times}}$$

We will not prove this formula explicitly, although we will give some justification for it in the remark below.

□

The entries in the matrix associated with the Markov chain are called transition probabilities. If we look at the entry in the kth row and mth column, we get the probability of transitioning from state k to state m in the next iteration given that we are currently in state k. To avoid writing out the words every time, we will write $p(k, m)$ to denote the entry in the matrix at row k and column m.

Just as we discussed graphs associated to walks on groups, we can construct a (di)graph associated to a Markov chain. Each state corresponds to a vertex. We draw an edge between vertices if the transition probability between those states is nonzero, keeping track of the direction the edge goes in (because the probabilities of going in each direction might be different and, in fact, one might be zero and the other might be nonzero).

If the probability of going from state i to state j is the same as going in the opposite direction (i.e. state j to state i) for any i and j, then the digraph is really just a graph. The matrix associated to a Markov chain with that property is **symmetric** (which will we define in just a second). The **main diagonal** of a matrix corresponds to the diagonal that goes from top-left to bottom-right (i.e. the entries whose row and column numbers are equal). The other diagonal is called the **skew diagonal**. If one reflected the entries of a symmetric matrix about the main diagonal, the matrix would not change, i.e. if each ijth entry were swapped with the jith entry, the matrix would be exactly the same. The matrix in the previous example was symmetric, for instance.

Something extremely special occurs if all the transition probabilities are either zero or some fixed number p. In the previous example, for instance, the probabilities were all either 0 or 0.5. This means one can factor out that fixed number p from the matrix to get a matrix with entries that are either 0 or 1. The matrix thereby produced is actually the adjacency matrix for the digraph! (In section 3.1, we constructed the adjacency matrix for a graph or digraph). At the end of section 3.1, we asked the reader to determine what was special about the adjacency matrices associated to graphs as opposed to digraphs.

Fact 9. *For actual graphs, the adjacency matrix is symmetric.*

The reader was also asked to determine what the entries in the matrix obtained by multiplying the adjacency matrix by itself corresponded to.

Fact 10. *If A is the adjacency matrix of a digraph, then the entry in the ith row and jth column of A^2 is the number of paths of length 2 starting at vertex*

i and ending at vertex j. Likewise, the ijth entry of the matrix obtained by multiplying the adjacency matrix by itself n times gives the number of paths of length n starting at vertex i and ending at vertex j.

Thus, we may somewhat justify the final equation in the previous example. Multiplying the matrix coming from the Markov chain by itself n times corresponds to multiplying the adjacency matrix for the graph by itself n times and multiplying each entry in the result by p^n.

This is what one would intuitively expect: since all the transition probabilities are equal, the probability of walking along a particular path of length 10 is the same as walking along *any* path of length 10, and the probability would be p^{10}. So if we want to know the probability of ending up at state j after 10 iterations given that we start at state i, we just need to count how many paths of length 10 start at i and end at j, and multiply the result by the probability of any single path of length 10, which is p^{10}. This is exactly what the entires of the Markov chain matrix to the tenth power are in this case!

4.2 Random Walks on Groups, Shuffling Techniques

One can always produce a Markov chain corresponding to a graph: the states correspond to vertices; one puts a zero in the Markov matrix in each position for which there is no corresponding edge; and one then assigns a positive probability to each edge and puts those probabilities in the relevant matrix entries.

Given a group G and generating set T, we can construct a graph — recall the vertices correspond to group elements and the edges correspond to elements of T. We assign a probability to each element of T: this means we decide ahead of time what the likelihood of picking any particular element of T is. At present, our choice of T is arbitrary, as is our choice of how likely we are to pick any particular element of T.

Once we have decided upon a probability on T, our graph of G generated by T can be associated with a Markov chain. This Markov chain defines what is called a **random walk on G**, or just a random walk if the group G is clear from context.

Assumption: From here on, we assume the group G is one of the symmetric groups, so that we are considering a random walk on S_n.

The set T along with the probability we assign to elements of T defines

our shuffling technique. The following example shuffles by using transpositions.

Example 11. Shuffling Using Random Transpositions

1. We lay all cards in the deck side by side.
2. With our left hand, we pick a card uniformly at random — this means that picking each card is equally likely. Picking any particular card with our left hand happens with probability $1/n$.
3. With our right hand, we pick a card uniformly at random, independently of our first choice. Note that our two hands might very well pick the same card. Picking any particular card with our right hand happens with probability $1/n$.
4. We swap the positions of the two cards. If both hands picked exactly the same card, the order remains unchanged.

Since we only interchange two cards or do nothing, the set T in this case corresponds to the set of transpositions of S_n. How do we determine the probability of picking any particular transposition? Well, the probability that we the pick A♠ with our left hand and the 10♢ with our right hand is $1/n^2$. However the same shuffle would happen if our right hand picked the A♠ with our right hand and 10♢ with our left. So the probability that we interchange the A♠ and the 10♢ is $2/n^2$. In fact, this is the probability that we pick any particular transposition except the identity (because we chose two distinct cards to begin with). By the same logic as above, the probability that our left and right hands pick the A♠ is $1/n^2$. This would result in the identity transposition. However, we get the identity transposition whenever our hands pick the same card, e.g. they could also both pick 10♢. Since there are n cards, this means we pick the identity with probability $n \times 1/n^2 = 1/n$.

With this, we have developed a model for our shuffling technique: we have a set T with a probability defined on each element. This generates a graph (see the graph in example 7 to see the graph for S_3). Since each edge has an associated probability, we can generate a matrix corresponding to the Markov chain. On S_3, this matrix is

$$\begin{pmatrix} & \text{to } 123 & \text{to } 132 & \text{to } 213 & \text{to } 231 & \text{to } 312 & \text{to } 321 \\ \text{from } 123 & 1/3 & 2/9 & 2/9 & 0 & 0 & 2/9 \\ \text{from } 132 & 2/9 & 1/3 & 0 & 2/9 & 2/9 & 0 \\ \text{from } 213 & 2/9 & 0 & 1/3 & 2/9 & 2/9 & 0 \\ \text{from } 231 & 0 & 2/9 & 2/9 & 1/3 & 0 & 2/9 \\ \text{from } 312 & 0 & 2/9 & 2/9 & 0 & 1/3 & 2/9 \\ \text{from } 321 & 2/9 & 0 & 0 & 2/9 & 2/9 & 1/3 \end{pmatrix}$$

Observe that this matrix to the tenth power is approximately

$$\begin{pmatrix} & \text{to } 123 & \text{to } 132 & \text{to } 213 & \text{to } 231 & \text{to } 312 & \text{to } 321 \\ \text{from } 123 & 0.16668 & 0.16666 & 0.16666 & 0.16666 & 0.16666 & 0.16666 \\ \text{from } 132 & 0.16666 & 0.16668 & 0.16666 & 0.16666 & 0.16666 & 0.16666 \\ \text{from } 213 & 0.16666 & 0.16666 & 0.16668 & 0.16666 & 0.16666 & 0.16666 \\ \text{from } 231 & 0.16666 & 0.16666 & 0.16666 & 0.16668 & 0.16666 & 0.16666 \\ \text{from } 312 & 0.16666 & 0.16666 & 0.16666 & 0.16666 & 0.16668 & 0.16666 \\ \text{from } 321 & 0.16666 & 0.16666 & 0.16666 & 0.16666 & 0.16666 & 0.16668 \end{pmatrix}.$$

The diagonal of this matrix is 0.00002 different from the other entries. This matrix is almost exactly

$$A := \begin{pmatrix} & \text{to } 123 & \text{to } 132 & \text{to } 213 & \text{to } 231 & \text{to } 312 & \text{to } 321 \\ \text{from } 123 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ \text{from } 132 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ \text{from } 213 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ \text{from } 231 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ \text{from } 312 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ \text{from } 321 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \end{pmatrix}.$$

Note that if v is any row vector whose entries sum to 1, then

$$vA = (1/6, 1/6, 1/6, 1/6, 1/6, 1/6).$$

Since A is almost exactly our Markov matrix to the tenth power, no matter what distribution we start out with, we have very nearly equal likelihood of being in any of the 6 orderings of the deck of three cards after ten shuffles. Also, note that $(1/6, 1/6, 1/6, 1/6, 1/6, 1/6)$ times our original Markov matrix is just $(1/6, 1/6, 1/6, 1/6, 1/6, 1/6)$ again.

□

In general, it might be complicated or even computationally infeasible to write down exactly what the elements of T are for more realistic shuffling techniques, e.g. for the standard (riffle) shuffle. As long as we know what probability we pick elements of T with, we know that our shuffling procedure generated by T can be represented by a matrix. This allows us to deduce some fundamental results about about card shuffling in general just by knowing facts about matrices.

5 Important General Definitions and Facts

For the problem of card shuffling, we want to know how long we should shuffle until the deck is well mixed. The definitions and results we present here will describe a mostly qualitative theorem to that effect: for reasonable shuffling strategies, repeatedly shuffling the deck will guarantee the orderings of the deck *eventually* get very close to being uniformly distributed. This means that if we shuffle a deck of n cards many times, the probability of the deck being in any particular ordering gets closer and closer to $1/n!$, which means that all the orderings eventually become equally likely. This is a nice result because it agrees with our intuitive idea of what shuffling should do: if we shuffle the deck, we shouldn't be able to guess what order the deck will be in. In other words, the deck "forgets" its original ordering after shuffling for a while. This is not really a satisfactory result because it does not explain what "eventually" means — and we really want to know how many times we need to shuffle the deck. If we had a shuffling strategy which required a billion shuffles to work, it certainly wouldn't be very useful!

We will now state some general definitions. The definitions are stated in generality not needed for our purposes, but they are stated this way so that the reader can understand the broad setting under which they apply.

Definition 12. Probability Distribution.

A probability distribution on a finite group G is a function defined on G which takes values between 0 and 1, inclusive, and the sum of all possible values is 1. We write this last condition as

$$\sum_{g \in G} P(g) = 1.$$

Definition 13. Random Walk on a Group.

Every probability distribution P on a group G defines a Markov chain. The (s,t)-th entry in the associated matrix is $p(s,t)$, where $p(s,t)$ is given

by

$$p(s,t) = P(ts^{-1}).$$

Here the element ts^{-1} is the product of two elements of the group. One should think of ts^{-1} as an arrow pointing from s to t, so that $P(ts^{-1})$ is the probability of starting at s and ending at t.

Definition 14. Stable Distribution for a Markov Chain.

Let M be the matrix associated to a Markov chain. Here we will think of probability distributions as row-vectors with the probabilities listed in the same order as the rows of M. A stable distribution for M is a distribution π so that

$$\pi M = \pi$$

In that way, the distribution π is fixed by M.

Definition 15. Evolution of a Distribution Under a Markov Chain.

Suppose that we start with a random walk on a group G with matrix M. If we start with the probability distribution Q, thinking about it as a row-vector as above, then the kth evolution of Q, denoted Q_k is given by

$$Q_k = QM^k$$

Recall Example 11. In that example, we noticed that no matter what distribution we started with, the evolution of the Markov chain tended to the uniform distribution $(1/6, ..., 1/6)$ and that the uniform distribution was a stable distribution for the Markov chain.

Under a pair of technical conditions (which we will avoid stating exactly), Q_k for large values of k gets very "close" to a stable distribution. As a matter of fact, the conditions guarantee there is only a single stable distribution. One of the technical conditions guarantees that the states for the Markov chain are not isolated from one another: there is always positive probability of eventually getting from one state to any other state, though it could take several time steps for the transition probability between two states to become positive. If some states were isolated from other states, the situation would be unsatisfactory for the purposes of randomizing the deck: if we started from one particular ordering of the deck, there would be orderings of the deck which would never occur, regardless of how long we tried. This would provide information to the shuffler who implemented this shuffling strategy. When all states are eventually reachable by evolving the Markov chain, the chain is called irreducible.

The second condition is called aperiodicity. Suppose that our chain had the following property: no matter how many times we shuffle, it is completely impossible to return the original ordering in an odd number of shuffles. Such a chain is periodic in some sense because every random path generated by our Markov chain which starts and ends at a particular vertex must have an even length. The same would be true if the length of every path which started and ended at the same vertex was a multiple of 3 (or 4 or 5...). This, too, is unsatisfactory for randomizing the order of the deck: if we kept track of the number of shuffles we had applied, we would know for certain that we would not be in the original order. A Markov chain is called aperiodic if for every state, the lengths of all possible paths which start and end at that state are not all multiples of the same number.

Fact 16. *Every irreducible, aperiodic Markov chain with matrix M has a stable distribution π. Moreover, the evolution QM^k of any starting distribution Q will eventually be approximately π, i.e. $QM^k \approx \pi$ for all k "large enough."*

Definition 17. Difference Between Two Probability Distributions on a Group. Given two probability distributions P and Q defined on a group G, we define the difference $|P - Q|$ as

$$|P - Q| = \frac{1}{2} \sum_{g \in G} |P(g) - Q(g)|.$$

The presence of the $\frac{1}{2}$ is to keep the difference between two distributions between 0 and 1.

Definition 18. When Shuffles are Well mixed.

Suppose that Q_k is the evolution of a shuffling strategy for n cards starting from some initial distribution Q on G in the row-vector sense. We will abuse notation here and also think of Q_k as the function on G so that the value $Q_k(g)$ takes the value corresponding to the g^{th} entry of the vector Q_k. Let U denote the uniform distribution, meaning $U(g) = 1/n!$ for every $g \in S_n$. If we find that after k shuffles that

$$|Q_k - U| \leq \frac{1}{2},$$

regardless of which starting distribution Q we are given, we will say that the deck is well mixed.

Remark: The 1/2 in Definition 18 is fairly arbitrary in that it could be replaced with any positive number strictly smaller than 1.

Using these definitions, one wants to know for a particular strategy how many shuffles to apply before the deck is well mixed so that further shuffling is not particularly necessary. In mathematical terms, is there an integer k_0 so that for all $k > k_0$, $|Q_k - U| < \frac{1}{2}$, no matter which distribution Q we start with? In particular, we don't want k_0 to depend at all on Q. What is the smallest value k_0 we can use?

It turns out that we can always produce a k_0, provided that repeated shuffles do not avoid particular orderings:

Theorem 19. *Suppose that after some number of shuffles, we know that the probability of hitting any particular ordering of the deck is bigger than some fixed number. In math-speak, this means that there exists a positive number c for all $k \geq k_0$ so that*

$$Q_k(g) \geq cU(g) \text{ for all } g \text{ in } G. \tag{1}$$

(We explain condition (1) below). If (1) is satisfied, then

$$|Q_k - U| \leq (1 - c)^{\lfloor k/k_0 \rfloor},$$

where $\lfloor x \rfloor$ means round x down to the nearest integer.

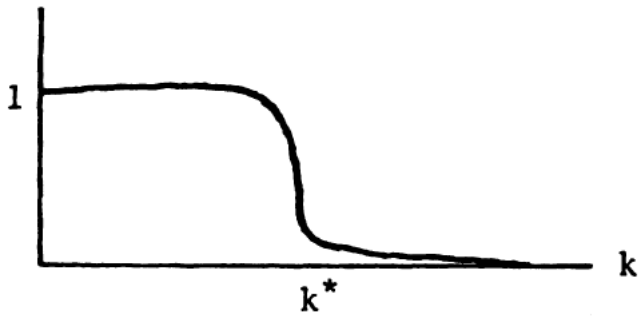
The condition in equation (1) is certainly necessary for any shuffling technique to randomize the deck. Note that $U(g)$ is a constant function. If a shuffling strategy did not satisfy this condition, then there would be some orderings which would be very, very unlikely at certain times. Since this would provide the shuffler with information about the order of the deck, we should be sure our shuffling strategy satisfies (1).

Even though this result seems quantitative, it is really a qualitative result. We generally don't know exactly what c or k_0 are, just that, eventually, the probability of being in any particular ordering becomes uniformly distributed, and the speed with which it becomes uniform is exponential.

Exponential functions *eventually* decay extremely rapidly, which is great news for us, but we still don't know how long we need to wait for the decay to move quickly. Let's define the value k^* to informally be the value of k where the exponential decay first starts to become strong. Refer to Figure 4 below to see what k^* is.

For $k < k^*$, the distribution Q_k is far away from uniform. After k^*, the distribution Q_k gets close to uniform very quickly. The problem with Theorem 19 is that it gives us no way to know what k^* is. Since our main

Figure 4: Graph of $|Q_k - U|$ as a function of k. Figure taken from [3, p. 24].



concern is how many times we need to shuffle the deck, we want to know precisely what k^* is.

This phenomenon is commonly referred to as a "cut-off". If a Markov chain is quite complicated, it may be difficult to guess correctly whether a chain is aperiodic or irreducible. Even if the chain is aperiodic and irreducible, it can be even more difficult to guess at exactly how quickly the chain should decay.

Exercise: In section 2.5, we describe three shuffling strategies. Which shuffling strategy seems intuitively the best, i.e. which cut-off happens fastest? Which is second-best? This is a hard problem to solve, so simply go with your gut as to how fast the strategies should work. The answer is at the end of this chapter.

To avoid some of these difficulties, one typically performs computer simulations of Markov chains to see if and when these cut-offs occur, typically with similar but simpler versions, like shuffling on a deck of 10 cards instead of 52. Once one has a good guess as to what the answer is from the simulations, one tries to do the math to prove the guess is correct.

6 The Standard Shuffle: Riffle Shuffle

In this section, we assume we are dealing with the standard deck of 52 cards. Much of this content comes from Persi Diaconis's book, [3].

6.1 The Riffle Shuffle

The most common shuffling method goes more or less as follows. The shuffler cuts the deck roughly in half, taking one half in each hand. The shuffler then drops a few cards from his or her left hand, a few cards from the right hand, and so on until one hand is empty. The characteristic most commonly associated to the shuffle is the zipper sound it makes. This shuffling technique is called the riffle shuffle.

The first thing we must do is come up with a specific mathematical description for this shuffling technique. We will describe three different ways to view this shuffle. Each of the three ways is "morally" equivalent to the others.

1. How do we cut the deck? To determine the number of cards we cut from the top, flip 52 coins and count the number of heads, then take that many cards off the top to cut the deck into two pieces. It may seem strange that there is a positive probability of having all 52 cards in the deck sitting in the "top half" but this is extremely unlikely, and for most shufflers, the size of the two "halves" are often fairly different. Suppose that our result is k cards in the top half. From here, we think of having 52 boxes lined up and put a card in each of them. We pick k of the boxes (assuming each box is equally likely) and put the top half of the deck in those boxes, keeping them in the same order. Put the remaining $52-k$ cards in the remaining boxes, keeping them in the same order. Stack the cards back up. Note that there are $\binom{52}{k} = \dfrac{52!}{k!(52-k)!}$ ways to put k cards in 52 boxes, so that, given the top half of the deck has k cards, the probability of picking any one of the possible orderings through this method is $\dfrac{1}{\binom{52}{k}}$.

2. Cut the deck into two halves just like before. Suppose that k cards are in the left hand and $52-k$ cards in the right hand. The probability at each step that we drop from the left hand or the right hand will be proportional to the number of cards in the left hand or right hand. Specifically, we decide to drop a card from the the left hand with probability $k/52$ and from the right hand with probability $(52-k)/52$. If a card from the left hand drops, then we do the same thing: drop a card from the left hand now with probability $(k-1)/51$ and from the right hand with probability $(52-k)/51$. Continue this process until all the cards have been dropped.

3. This is the inverse shuffle. For each card in the deck, flip a coin and label the back of the card H or T depending on whether the coin landed heads or tails. Take all the cards labeled H out of the deck, maintaining

them in relative order to one another and put them on top.

We see #1 and #2 are the same because, to begin with, one flips coins in the same way. Assuming that k heads come up, exactly k cards end up in the left hand and $52-k$ cards in the right hand. In #1, one ends up with a sequence of 52 L's and R's: the first is an L if the top card after the shuffle came from the left hand and R if it came from the right. Likewise with the second entry, and so on. Each of these orderings is equally likely. In #2, one ends up with a sequence of L's and R's depending on the order in which the cards dropped. The probability of any particular ordering is always $\dfrac{k!(52-k)!}{52!} = \dfrac{1}{\binom{52}{k}}$, which is the same as the probability we found in #1.

#1 and #3 are not precisely the same, however, they are very similar in the following sense. Observe that every riffle shuffle corresponds to exactly one inverse riffle shuffle which "undoes" the riffle shuffle. In order to get a particular riffle shuffle, we need to get exactly k heads, which occurs with probability $\dfrac{1}{2^{52}}\binom{52}{k}$ and we need to pick a particular way to put the k cards into the 52 boxes which happens with probability $\dfrac{1}{\binom{52}{k}}$. Thus the odds of getting any particular riffle shuffle is $1/2^{52}$. The odds of getting any particular inverse shuffle is the same as the odds of getting any particular sequence of 52 coin flips, or $1/2^{52}$. So the odds of getting any particular riffle shuffle is the same as getting any particular inverse riffle shuffle. The inverse shuffle walk is, in some sense, a lot like usual riffle shuffle walk but *backwards*. If $P(g)$ denotes the probability from the riffle shuffle, then $\widetilde{P}(g) := P(g^{-1})$ is the probability for the inverse shuffle. By carefully considering Definitions 17 and 18 and applying a few tricks, one can see that determining when the inverse shuffles become well mixed is the same as determining when the riffle shuffles become well mixed.

This model of the riffle shuffle is rather accurate for amateur shufflers. Professionals (casino dealers, magicians, and so forth) are not modeled quite as well by this technique since they are able to cut the deck nearly perfectly in half all the time and also tend to drop just a few cards at a time from each hand. In fact, if one can riffle shuffle by cutting the deck perfectly in half and drop exactly one card at a time from each hand, then 8 perfect shuffles will bring the deck back into exactly the original order. Amateurs are actually somewhat better shufflers after a single shuffle, but, after a few shuffles, professionals are much better.

Now that we have a good model of the riffle shuffle, we're well positioned

to find out what k^* is. But first we need some definitions.

Definition 20. Stopping Time

Suppose that we shuffle the deck over and over again. This produces a sequence of elements of S_{52}. A stopping time, $\mathscr{T}$, is a function which takes as input a sequence of shuffles and looks for a particular phenomenon to happen for the first time. The stopping time keeps track of the number of times we shuffle until this happens (and then says, "stop!"). If this phenomenon happens after m shuffles in our sequence, the stopping time outputs m. For example, we might say, "Stop when the deck has an ace on top." If we shuffle 10 times before an ace is at the top of the deck, the stopping time would output 10. If we shuffled the deck once and an ace wound up on top, the stopping time would output 1.

Definition 21. Strong Uniform Time

Suppose we have a stopping time $\mathscr{T}$. $\mathscr{T}$ is a strong uniform time if the outcomes of k shuffles under the requirement that $\mathscr{T} = k$ all are equally likely.

To make the definition a bit more clear, consider the following example.

Example 22. Suppose we shuffle the deck with the following method: simply take the top card of the deck and put it into the deck at random. This corresponds to picking randomly one of the following cycles:

$$\varepsilon, \begin{pmatrix} 2 & 1 \end{pmatrix}, \begin{pmatrix} 3 & 2 & 1 \end{pmatrix}, ..., \begin{pmatrix} n & n-1 & ... & 2 & 1 \end{pmatrix}.$$

In particular, we have the probability distribution P given by

$$P(\varepsilon) = P(\begin{pmatrix} 2 & 1 \end{pmatrix}) = ... = P(\begin{pmatrix} n & n-1 & ... & 2 & 1 \end{pmatrix}) = \frac{1}{n}.$$

What if we define $\mathscr{T}$ to be the first time the original bottom card is shuffled randomly into the deck? This is certainly a stopping time, but is it a strong uniform time? The original bottom card obviously stays on the bottom of the deck until a card is put underneath it, which happens the first time we pick the shuffle

$$\begin{pmatrix} n & n-1 & ... & 2 & 1 \end{pmatrix}.$$

The first time a second card is put underneath the original bottom card, the order of the last two cards is equally: there are exactly two possible shuffles which will move a card below the original bottom card, namely

$$\begin{pmatrix} n-1 & n-2 & ... & 2 & 1 \end{pmatrix} \text{ and } \begin{pmatrix} n & n-1 & ... & 2 & 1 \end{pmatrix},$$

both of which are equally likely. Likewise, if we consider the first time a third card is put below the original bottom card, all possible orderings of the bottom three cards are equal. This is because, before this shuffle, the bottom two cards are in random order, and the probability of the third card being placed above, between, or below these two cards is equally likely. This trend of course continues until the first time the original bottom card reaches the top. At this point, the entire deck is in a random order except the original bottom card is on top. Once we shuffle that card in at random, any ordering of the deck is equally likely (the odds of any particular ordering of the deck occurring is $\frac{1}{n!}$). Thus this is indeed a strong uniform time.

□

The following theorem makes strong uniform times extremely useful:

Theorem 23. *Suppose that $\mathscr{T}$ is a strong uniform time. Let $P(\mathscr{T} > k)$ denote the probability that we stop after k rounds of shuffling. Then*

$$\|Q_k - U\| \leq P(\mathscr{T} > k).$$

The proof of this result is not especially difficult nor is it especially enlightening. The goal now is two-fold. We want to construct a strong uniform time T for the riffle shuffle and compute the probability that $\mathscr{T} > k$ so that we can apply the above theorem. Provided this second task produces a formula, we can guarantee the number of shuffles required. This is a trick which, when known, makes the problem extremely easy.

Let us now describe the strong uniform time. List the 52 cards of the deck as the rows of a matrix. We perform repeated inverse shuffles, which are described in **3** above. At each shuffle, add an additional column to the matrix. Put an H (or T) in the row for each card if in that shuffle, that card was associated with a heads (or tails). If one card is an A♡, then we keep track of whether that card got a heads or a tails. As an example, suppose our deck has 4 cards, A, 2, 3, 4. If the first inverse shuffle had coin flips HHTH, the second had HTTT, and the third was THHT, for the four cards, respectively, then the matrix would be

$$\begin{pmatrix} A & H & H & T \\ 2 & H & T & H \\ 3 & T & T & H \\ 4 & H & T & T \end{pmatrix}.$$

Of course, we can continue adding columns as we perform more shuffles.

The rows produce a way to order the cards. The rest of this paragraph will make sense much more quickly if one experiments with a small stack of cards, say 4. After the first inverse shuffle, all the cards which flipped heads are in the top "half" of the deck and the cards that got tails are in the bottom "half". After two flips, the cards whose second flip was heads sit above those cards whose second flip was tails. But within the group of cards whose second flip came up heads, the cards whose first flip was also heads sit above those whose first flip was tails. In particular, the HH cards sit above the TH cards, which sit above the HT cards, and the TT cards are all on the bottom. For the first three columns, the ordering would be HHH, THH, HTH, TTH, HHT, THT, HTT, TTT.

For a sequence of shuffles, we construct a (very large) matrix A. Let $\mathscr{T}$ be defined on sequences of shuffles so that $\mathscr{T} = k$ when k is the minimum number of columns necessary to make the rows of A distinct (that is, no two rows are exactly the same). For 52 cards, $\mathscr{T} \geq 6$; this is not too hard to see, and you should convince yourself this is true. $\mathscr{T}$ is a stopping time because if $\mathscr{T} = 10$, we are able to determine that $\mathscr{T} = 10$ based on just the first 10 shuffles rather than any future shuffles.

$\mathscr{T}$ is a strong uniform time because, assuming that $\mathscr{T} = k$, one could easily interchange any rows in the matrix A to produce an equally valid sequence of riffle shuffles which still has $\mathscr{T} = k$. Since the sequence in each row determines the position of each card in the deck, this gives us a way to produce any ordering of the deck, and thus there are $n!$ ways of reordering the rows. But any sequence of shuffles which has the same rows as our matrix A are all equally likely, and so the probability of getting a particular outcome X_k, assuming $\mathscr{T} = k$, has probability $1/n!$. Hence $\mathscr{T}$ is a strong uniform time.

Now that we know $\mathscr{T}$ is a strong uniform time, we need to compute the probability that $\mathscr{T}$ is bigger than k. To do that, the following lemma is useful:

Lemma 24. *The Birthday Problem.*

Suppose that there are n people in a room. Assuming that birthdays are independently assigned and the probability of having any particular day of the year is uniformly distributed, the probability that no two of people in the room have the same birthday is

$$L = \left(\frac{364}{365}\right) \times \left(\frac{363}{365}\right) \times \left(\frac{362}{365}\right) \times ... \times \left(\frac{365-(n-2)}{365}\right) \times \left(\frac{365-(n-1)}{365}\right).$$

Hence the probability that at least two people have the same birthday is

$$1 - L.$$

Proof. Put the people in the room in order. There are 365 possibilities for the birthday of the first person. For the second person to have a different birthday, he or she must have one of the 364 other birthdays. Picking the birthday at random, this happens with probability 364/365. Thus the probability that the first two people have distinct birthdays is 364/365. For the third person to have a different birthday, he or she must have one of the 363 birthdays left. Since people's birthdays are independent of one another, this happens with probability $\frac{364}{365}\frac{363}{365}$. Repeating this gives the formula for L. Since the probability of an event not happening is one minus the probability that the event does happen, the probability that two people have the same birthday is $1 - L$. □

The situation we are in with $P(\mathscr{T} > k)$ is similar to the Birthday Problem. Here we think of the cards as people and think of their associated rows as their birthdays. The probability that $\mathscr{T}$ is greater than k is the probability that at least two cards have the same rows of length k: there are 2^k rows of length k (since there are 2 possibilities for each entry, H or T), and thus 2^k birthdays. There are 52 cards. Hence

$$P(\mathscr{T} > k) \leq 1 - \left(1 - \frac{1}{2^k}\right)\left(1 - \frac{2}{2^k}\right)\cdots\left(1 - \frac{50}{2^k}\right)\left(1 - \frac{51}{2^k}\right) := H(k)$$

Computing the right side of this inequality with a computer for various values of k gives (rounding to the nearest thousandth):

k	$H(k)$
8	0.996
9	0.932
10	0.732
11	0.480
12	0.278
13	0.150
14	0.078

Thus 11 shuffles should suffice to randomize a deck of 52 cards. Of course, our chosen level of closeness (1/2) was arbitrary, so the number 11 is only useful in comparing different shuffling techniques using the same level.

7 Other Results and Remarks

Here we will assume the deck size can be any positive integer. In the previous section, we showed that

$$P(\mathscr{T} > k) \leq H(k).$$

Using calculus, this expression on the right can be simplified a bit. For some number $c > 0$, let $k = 2\log_2(n/c)$. Using facts from calculus about what the exponential function is, one can prove that

$$H(k) \approx 1 - e^{-c^2/2} \leq c^2/2.$$

If we pick c to be 1, we see that $2\log_2(n)$ gives us an accurate approximation of the number of shuffles needed to get a deck of n cards well mixed. Technically speaking, this $2\log_2(n)$ is merely an upper bound on the number of shuffles needed to get well mixed. That is to say that it never requires more than $2\log_2(n)$ shuffles to get well mixed, but it may be that fewer shuffles are sufficient. In 1983, mathematician David Aldous, [1], showed that $\frac{3}{2}\log_2(n)$ is the right number of shuffles when n is relatively large. In fact, using the techniques in [8, p. 275], one can produce a table like we did for $H(k)$ to prove that precisely 7 shuffles is enough to get the deck sufficiently randomized when $n = 52$.

This previous computation (which gave us $2\log_2(n)$), while based upon an arbitrary choice of $1/2$ as the "closeness threshold," does give us something independent of this choice: the number of shuffles required *grows logarithmically* with the size of the deck. This is really good: a deck with 10^{10} cards would require something on the order of 100 shuffles which is much smaller than 10^{10}. A deck with 10^{100} cards would require on the order of 1000 shuffles.

There are a number of different shuffling techniques. Using completely different methods, mathematicians have determined how the number of shuffles required to mix the deck grows with the size of the deck. We'll describe some different shuffling techniques and then show how they grow (with relatively large deck sizes):

1. We have already discussed shuffling using random transpositions.

2. The overhand shuffle, as described in the Shuffling Strategies section.

3. This is called the Rudvalis shufle. One chooses with equal probability one of the following:

(a) Do nothing.

(b) Move the top card to the bottom.

(c) Move the bottom card to the top.

(d) Move the card below the top card to the bottom.

(e) Move the card above the bottom card to the top.

It is possible using different techniques to determine the growth of the number of shuffles needed with respect to the size of the deck:

1. For riffle shuffles, the number of shuffles required grows like $\log_2(n)$,
2. For #**1** above, the number of shuffles required grows like $n \log_2(n)$, [4]
3. For #**2** above, the number of shuffles required grows like $n^2 \log_2 n$,[7],[5],
4. For #**3** above, the number of shuffles required grows like $n^3 \log_2(n)$,[9].

This means that, compared to the riffle shuffle, the overhand shuffle is worthless. Roughly 2500 overhand shuffles are required for a normal deck as compared to the riffle shuffle's roughly ten shuffles. The Rudvalis shuffle seems particularly horrible; there are only five moves, and most of your moves are spent undoing each other. The number of shuffles required for this shuffling technique still grows just barely faster than n^3, yet the number of orderings grows like $n!$. This seems quite surprising. We now leave the reader to wonder: how slowly can a shuffling technique satisfying Theorem 19 shuffle the deck?

References

[1] D. Aldous. Random walks on finite groups and rapidly mixing Markov chains. In *Séminaire de probabilités XVII*, pages 243–297. Number 986 in Lecture notes in Mathematics. Springer-Verlag, 1983.

[2] R. Baldwin, W. Cantey, H. Maisel, and J. McDermott. The optimum strategy in blackjack. *Journal of the American Statistical Association*, 51(275):429–439, 1956.

[3] P. Diaconis. *Group Representations in Probability and Statistics*, volume 11 of *Lecture Notes-Monograph Series*. Institute of Mathematical Statistics, 1988.

[4] P. Diaconis and M. Shahshahani. Generating a random permutation with random transpositions. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 57(2):159–179, 1981.

[5] J. Jonasson. The overhand shuffle mixes in $\Theta(n^2 \log n)$ steps. *The Annals of Applied Probability*, 16(1):231–243, 2006.

[6] R. Luketic. 21. Columbia Pictures, 2008.

[7] R. Pemantle. Randomization time for the overhand shuffle. *Journal of Theoretical Probability*, 2(1):37–49, 1989.

[8] L. Saloff-Coste. Random Walks on Finite Groups. In *Probability on Discrete Structures*, volume 110 of *Encyclopedia of Mathematical Sciences*, pages 263–346. Springer, Berlin, 2004.

[9] D. Wilson. Mixing time of the rudvalis shuffle. *Electronic Communications in Probability*, 8:77–85, 2003.

Index