A Closer Look at Unravelling Shuffles

Anav Sood

June 2018

1 Introduction

In the following paper, we examine a couple different kinds of shuffles, the overhead shuffle and riffle shuffle, in effort to analyze how well each shuffle randomizes the order of the pre-shuffled deck. One metric of doing so is attempting to come up with an effective guessing strategy for the order of the deck has been shuffled once. The more cards you guess correct in expectation, the worse the shuffle is. Below I comment on what I was able to do in this paper with the time given and what there is still to accomplish. All the results below are my own research.

For the overhead shuffle we offer a guessing strategy that we conjecture is optimal with respect to expectation. We find a closed form of its expected value and variance. We also make some discussion on a slight variation of the strategy. Future work includes proving the strategy is or isn't optimal (I conjecture it is). Also I made some effort to discuss generalizing the strategy to n-shuffles but made minimal progress. There is certainly more work to be done here.

For the riffle shuffle we offer what we believe to be the true optimal guessing strategy. The problem of proving that McGrath's guessing strategy is optimal was provided as the eighth open problem in Diaconis' Mathematical Developments from the Analysis of Riffle Shuffling (see [1]). McGraths's strategy is described in [2]. I believe that McGrath's method is not optimal, but close to it when the number of cards N is large. I provide a computation heavy guessing strategy which I believe to be optimal, and a method for computing its expectation. The most natural course of future work would involve comparing the strategy to McGrath's and figuring out when the two line up (which should be after the card topping the second pile of the cut is revealed), and also proving that there is a case where McGrath's strategy is sub-optimal in comparison to the one provided. There is also a need to actually compute the given expectation. Additionally the strategy can be naturally extended to n-shuffles, but there is some work in writing out the details.

2 The Overhead Shuffle

The overhead shuffle is a common, simple shuffling technique involving grabbing a stack of cards from the top of the deck and placing the stack, in order, at the bottom. As one can imagine, since the stacks are left in order, the shuffle is not an ideal randomizing agent. We can define the shuffle as follows.

Definition 2.1. Suppose we have a N card deck where we label the bottom card as the 1st card and the top card as the Nth card. Consider $X_1, \ldots, X_{N-1}$ to be $N-1$ i.i.d Bernoulli random variables with parameter $\theta \in [0,1]$, so $\mathbf{P}(X_i = 1) = \theta$ and $\mathbf{P}(X_i = 0) = 1-\theta$. Let $S = \sum_{i=1}^{N-1} X_i$.

Suppose $i_1 > \cdots > i_S$ are indices such that $X_{i_k} = 1$. We take all the cards with label higher than i_1 and make it our first (lowest) stack, we take all the cards with label higher than i_2 and make it our second stack, so on and so forth. The remaining cards are made into our last stack.

Experiments shows that a reasonable θ value which mimics human performance is 0.2. Note if $\theta = 1$ the deck is simply put in reverse order and if $\theta = 0$ no changes occur. These edge cases are deterministic, so for our analyses we will assume $\theta \in (0, 1)$. We provide an example of a overhead shuffle with a 10-card deck below.

10			2
	$X_9 = 0$		
9			1
	$X_8 = 1$		
8			5
	$X_7 = 0$		
7			4
	$X_6 = 0$		
6		$\rightarrow$	3
	$X_5 = 1$		
5			8
	$X_4 = 0$		
4			7
	$X_3 = 0$		
3			6
	$X_2 = 1$		
2			10
	$X_1 = 0$		
1			9

2.1 Developing Guessing Strategies

Now, to get an understanding of how well such a shuffle randomizes our deck, we try to answer the following question. Suppose a friend shows us an ordered deck of cards and performs one overhead shuffle on the deck. He wants us to guess the resulting order of the deck card by card from the top of the deck to bottom. Upon our guess of any particular card, he reveals that card to us. Note that the better the deck is randomized, the less information this reveal should give to us. What is the optimal guessing strategy? We first offer a greedy approach, but highlight what steps in the greedy approach are optimal as well.

As a start to answering this question, we focus our attention on guessing the first card. It is obvious that should $X_k = 1$, then card k is the top of an in-order stack. The first card will be the top of the final stack, thus if we adopt the indices $i_1 > \cdots > i_S$ from the definition above i_S will be the label of the top card. Suppose T is the label of the top card. Then it is easy to see that T follows a geometric distribution with finite support. In particular

$$P(T = i) = \begin{cases} \theta & \text{if } i = 1 \\ \theta(1-\theta)^{i-1} & \text{if } 1 < i < N \\ (1-\theta)^{N-1} & \text{if } i = N \end{cases}$$

Since $\theta \in (0, 1)$ we always have $\theta > \theta(1-\theta)^i$. For reasonable values of θ and N, we will also have that $\theta > (1-\theta)^{N-1}$. This implies it is most likely that our top card is our bottom card, in which case we guess so accordingly. This is clearly the optimal decision. Regardless of whether or not we guess right, the top card of the deck is revealed to us and it is thus automatically the top of the stack. If the label of the card is $k > 1$, then it is obviously optimal to guess $k-1$, $k-2$, ..., 1 for the next cards, as this is determined to be the case. Once we're at the point where we've exhausted guessing cards in the stack, we must guess the next card, which will be the top of the second to last stack. In doing so, we find ourselves repeating the above process with a smaller deck. To stay mathematically true to the greedy nature of our approach, we must then start considering that it might be the case $\theta < (1-\theta)^{\alpha}$ for some small $\alpha > 0$. Recognizing that in practice, it may be hard to guess our friend's θ value and also difficult to compute α if we have not done so beforehand, we first offer a simplified strategy where we ignore this possibility. Thus if indeed T is our top card, the most likely guess for the next top of a stack is $T+1$ for the reasons outlined above. This leads to the following guessing strategy:

Strategy 2.2. Begin by guessing the bottom card of the un-shuffled deck as the top card of the shuffled deck. Then, if possible once the top card of the un-shuffled deck is revealed, guess the remaining cards in the stack of the said card. Suppose the last card you guessed was k. Guess $k+1$ for the next card. Once this card is revealed, guess the remaining cards in it's stack. Continue this process until you have exhausted the deck.

Because the analysis of this guessing strategy will be encompassed in the more robust one, we leave it for later.

Now suppose we want to account for the fact that $\theta < (1-\theta)^{\alpha}$ for some small $\alpha > 0$. Here we are recognizing that it may be more likely for the top card of the un-shuffled deck to be the top of the next stack we are guessing than the next card in order. To understand intuitively why we may take this into account, one can check that the top card of the un-shuffled deck will always be the top of the stack, and thus it is reasonable that the greedy algorithm demands that we make an effort to search for it. To accommodate this we offer the following robust greedy strategy:

Strategy 2.3. In the case that $\theta < (1-\theta)^{N-1}$ guess the top card of the un-shuffled deck as the top card of the shuffled deck, otherwise guess the bottom card of the un-shuffled deck as the top card of the shuffled deck. Then, if possible once the top card of the un-shuffled deck is revealed, guess the remaining cards in the stack of the said card. Now suppose k cards have been guessed. If it is the case that $\theta < (1-\theta)^{N-k}$, then guess the top card of the un-shuffled deck as the next card, otherwise guess the $k+1$st card. Once this card is revealed, guess the remaining cards in it's stack. Continue this process until you have exhausted the deck.

2.2 Probabilistic Analysis

We will now analyze the above strategy in expectation and variance. Suppose we let C represent the number of cards we guess correctly and $S = \sum_{i=1}^{N-1} X_i$ to be as above. We note that while guessing, we are guaranteed to guess any card correctly that is not the top of a stack. Since every top of the stack, excluding the top most card of the un-shuffled deck, has label k such that $X_k = 1$, we find that the total number of stacks is $S+1$. Now, letting R be the number of tops of stacks we do guess correctly, we have that

$$C = N + R - (S+1)$$

We note that our new strategy is very similar to our old strategy, but at integer time $\tau(\theta)$, rather than guess the $\tau(\theta)$th card as the top of our next stack, we would rather guess the top of the un-shuffled deck. In the case that $\tau(\theta) = 1$, we only ever guess the top of the un-shuffled deck, so $R = 1$ as the top of the un-shuffled deck is the only stack we will guess correctly. In the case that $\tau(\theta) = 2$ we see have a chance at correctly guessing the bottom card of the unshuffled deck as the top of the stack iff $X_1 = 1$. We will also correctly guess the top card of the un-shuffled deck as the top of a stack iff it is not the case that $X_1 = \cdots = X_{N-1} = 0$. Thus we will have

$$R = X_1 + 1 - \prod_{i=1}^{N-1}(1 - X_i)$$

In cases where $\tau(\theta) = k$ where $k \in \{3, \ldots, N-1\}$, then we follow the logic above, and now recognize for card i where $i \in \{2, \ldots, N-1\}$ that we correctly guess i to be the top of a stack iff $X_i = X_{i-1} = 1$. Thus we have

$$R = X_1 + 1 - \prod_{i=k-1}^{N-1}(1 - X_i) + \sum_{i=2}^{k-1} X_i X_{i-1}$$

And finally in the case that $\tau(\theta) = N$ we are simply making no effort to purposefully guess the top of the un-shuffled deck as the top of our stack. In this case we guess the top of the un-shuffled deck as the top of a stack iff $X_{N-1} = 1$, so

$$R = X_1 + X_N + \sum_{i=2}^{N-1} X_i X_{i-1}$$

From this we find

$$R = \begin{cases} 1 & \text{if } \tau(\theta) = 1 \\ X_1 + 1 - \prod_{i=1}^{N-1}(1 - X_i) & \text{if } \tau(\theta) = 2 \\ X_1 + 1 - \prod_{i=\tau(\theta)-1}^{N-1}(1 - X_i) + \sum_{i=2}^{\tau(\theta)-1} X_i X_{i-1} & \text{if } 2 < \tau(\theta) \leq N \end{cases}$$

Using this we have that

$$C = \begin{cases} N - \sum_{i=1}^{N-1} X_i & \text{if } \tau(\theta) = 1 \\ N - \sum_{i=2}^{N-1} X_i - \prod_{i=1}^{N-1}(1 - X_i) & \text{if } \tau(\theta) = 2 \\ N - \sum_{i=2}^{N-1} X_i - \prod_{i=\tau(\theta)-1}^{N-1}(1 - X_i) + \sum_{i=2}^{\tau(\theta)-1} X_i X_{i-1} & \text{if } 2 < \tau(\theta) \leq N \end{cases}$$

Making it easy to compute our expected value, which we provide below

$$\mathbf{E}(C) = \begin{cases} N - (N-1)\theta & \text{if } \tau(\theta) = 1 \\ N - (N-2)\theta + (\tau(\theta) - 2)\theta^2 - (1-\theta)^{N-\tau(\theta)+1} & \text{if } 2 \leq \tau(\theta) \leq N \end{cases}$$

In an effort to find the variance, we first must find C^2 which requires lengthy computation. The computation is quite involved, so we detail it to show some of the tricks used in casework below. Note since X_i carries all it's weight on 0 and 1 we can use the identities $X_i^2 = X_i$, $(1 - X_i)^2 = (1 - X_i)$, and $(1 - X_i)X_i = 0$.

Case One. Suppose $\tau(\theta) = 1$ so we have that $C = N - \sum_{i=1}^{N-1} X_i$, so

$$\begin{aligned}
C^2 &= N^2 - 2N \sum_{i=1}^{N-1} X_i + (\sum_{i=1}^{N-1} X_i)^2 \\
&= N^2 - 2N \sum_{i=1}^{N-1} X_i + 2 \sum_{1 \le i < j \le N-1} X_i X_j + \sum_{i=1}^{N-1} X_i \\
&= N^2 + 2 \sum_{1 \le i < j \le N-1} X_i X_j + (1 - 2N) \sum_{i=1}^{N-1} X_i
\end{aligned}$$

Case Two. Suppose $\tau(\theta) = 2$ so we have that $C = N - \sum_{i=2}^{N-1} X_i - \prod_{i=1}^{N-1}(1 - X_i)$, so

$$\begin{aligned}
C^2 &= N^2 + (\sum_{i=2}^{N-1} X_i)^2 + \prod_{i=1}^{N-1}(1 - X_i)^2 + 2\Big(- N \sum_{i=2}^{N-1} X_i - N \prod_{i=1}^{N-1}(1 - X_i) + \sum_{i=2}^{N-1} \Big(X_i \prod_{j=1}^{N-1}(1 - X_j)\Big)\Big) \\
&= N^2 + 2 \sum_{2 \le i < j \le N-1} X_i X_j + (1 - 2N)(\sum_{i=2}^{N-1} X_i + \prod_{i=1}^{N-1}(1 - X_i))
\end{aligned}$$

Case Three. Suppose $2 < \tau(\theta) \le N$ so we have that $C = N - \sum_{i=2}^{N-1} X_i - \prod_{i=\tau(\theta)-1}^{N-1}(1 - X_i) + \sum_{i=2}^{\tau(\theta)-1} X_i X_{i-1}$, so

$$
\begin{aligned}
C^2 &= N^2 + 2\sum_{2\le i<j\le N-1} X_iX_j + \sum_{i=2}^{N-1} X_i + \prod_{i=\tau(\theta)-1}^{N-1}(1-X_i) + 2\sum_{2\le i<j\le \tau(\theta)-1} X_iX_{i-1}X_jX_{j-1} + \sum_{i=2}^{\tau(\theta)-1} X_iX_{i-1} \\
&+ 2(-N\sum_{i=2}^{N-1} X_i - N\prod_{\tau(\theta)-1}^{N-1}(1-X_i) + N\sum_{i=2}^{\tau(\theta)-1} X_iX_{i-1} + \sum_{i=2}^{N-1}\left(X_i\prod_{j=\tau(\theta)-1}^{N-1}(1-X_j)\right) - \sum_{i=2}^{N-1}\sum_{j=2}^{\tau(\theta)-1} X_iX_jX_{j-1} \\
&- \sum_{i=2}^{\tau(\theta)-1}\left(X_iX_{i-1}\prod_{j=\tau(\theta)-1}^{N-1}(1-X_j)\right)) \\
&= N^2 + 2\sum_{2\le i<j\le N-1} X_iX_j + 2\sum_{2\le i<j\le \tau(\theta)-1} X_iX_{i-1}X_jX_{j-1} + (1-2N)(\sum_{i=2}^{N-1} X_i + \prod_{i=\tau(\theta)-1}^{N-1}(1-X_i)) \\
&(1+2N)\sum_{i=2}^{\tau(\theta)-1} X_iX_{i-1} + 2\sum_{i=2}^{\tau(\theta)-2}\left(X_i(1-X_{i-1})\prod_{j=\tau(\theta)-1}^{N-1}(1-X_j)\right) - 2\sum_{i=2}^{N-1}\sum_{j=2}^{\tau(\theta)-1} X_iX_jX_{j-1} \\
&= N^2 + 2\sum_{2\le i<j\le N-1} X_iX_j + 2\sum_{i=2}^{\tau(\theta)-2} X_{i-1}X_iX_{i+1} + 2\sum_{2<i+1<j\le \tau(\theta)-1} X_iX_{i-1}X_jX_{j-1} \\
&+ (1-2N)(\sum_{i=2}^{N-1} X_i + \prod_{i=\tau(\theta)-1}^{N-1}(1-X_i)) + (1+2N)\sum_{i=2}^{\tau(\theta)-1} X_iX_{i-1} \\
&+ 2\sum_{i=2}^{\tau(\theta)-2}\left(X_i(1-X_{i-1})\prod_{\tau(\theta)-1}^{N-1}(1-X_j)\right) - 2\sum_{i=2}^{N-1}\sum_{j=2}^{\tau(\theta)-1} X_iX_jX_{j-1}
\end{aligned}
$$

From this, with some pretty careful and painful computation, we find that

$$
\mathbf{E}(C^2) = \begin{cases} N^2 + (1-2N)(N-1)\theta + (N-1)(N-2)\theta^2 & \text{if } \tau(\theta) = 1 \\ N^2 + (1-2N)(N-2)\theta + (N-2)(N-3)\theta^2 + (1-2N)(1-\theta)^{N-1} & \text{if } \tau(\theta) = 2 \\ N^2 + (1-2N)(N-2)\theta + ((N-2)(N-3) + (2N-3)(\tau(\theta)-2))\theta^2 & \text{if } 3 \le \tau(\theta) \le N \\ 2((3\tau(\theta)-7) - (\tau(\theta)-2)(N-2))\theta^3 + (\tau(\theta)-3)(\tau(\theta)-4)\theta^4 & \\ +((1-2N) + 2(\tau(\theta)-3)\theta + 2(3-\tau(\theta))\theta^2)(1-\theta)^{N-\tau(\theta)+1} & \end{cases}
$$

This gives us the following result

Theorem 2.4. For Strategy 2.3, if C represents the number of cards guessed correctly, then

$$
\mathbf{E}(C) = \begin{cases} N - (N-1)\theta & \text{if } \tau(\theta) = 1 \\ N - (N-2)\theta + (\tau(\theta)-2)\theta^2 - (1-\theta)^{N-\tau(\theta)+1} & \text{if } 2 \le \tau(\theta) \le N \end{cases}
$$

$$
\mathbf{E}(C^2) = \begin{cases} N^2 + (1-2N)(N-1)\theta + (N-1)(N-2)\theta^2 & \text{if } \tau(\theta) = 1 \\ N^2 + (1-2N)(N-2)\theta + (N-2)(N-3)\theta^2 + (1-2N)(1-\theta)^{N-1} & \text{if } \tau(\theta) = 2 \\ N^2 + (1-2N)(N-2)\theta + ((N-2)(N-3) + (2N-3)(\tau(\theta)-2))\theta^2 & \text{if } 3 \le \tau(\theta) \le N \\ 2((3\tau(\theta)-7) - (\tau(\theta)-2)(N-2))\theta^3 + (\tau(\theta)-3)(\tau(\theta)-4)\theta^4 & \\ +((1-2N) + 2(\tau(\theta)-3)\theta + 2(3-\tau(\theta))\theta^2)(1-\theta)^{N-\tau(\theta)+1} & \end{cases}
$$

where $\tau(\theta)$ is the time we choose to guess the top card of the deck rather instead of the card with the label of the card we are currently guessing.

2.3 Finding $\tau(\theta)$

Now it is important to discuss how to find $\tau(\theta)$. The greedy algorithm dictates that $\tau(\theta)$ is set so that if $k \geq \tau(\theta)$, then $\theta < (1-\theta)^{N-k}$. Solving $\theta = (1-\theta)^{N-k}$ for k gives $k = \frac{\log \theta}{\log 1-\theta}$. Thus a reasonable greedy time is given by

$$\tau(\theta) = \max(\min(\left\lceil N - \frac{\log(\theta)}{\log(1-\theta)} \right\rceil, N), 1)$$

We can also ask, rather than take a greedy approach, what if we simply tried to maximize our expectation condition in terms of $\tau(\theta)$? This amounts to maximizing the function $x\theta^2 - (1-\theta)^{N-x+1}$ with respect to x. The first order condition gives

$$0 = \theta^2 + \log(1-\theta)(1-\theta)^{N-x+1}$$

Solving the above for x yields

$$x = N - 2\frac{\log(\theta)}{\log(1-\theta)} + \frac{\log[(\theta-1)\log(1-\theta)]}{\log(1-\theta)}$$

we can thus define an optimal stopping time $\alpha(\theta)$ as follows

$$\alpha(\theta) = \max(\min(\text{round}\left(N - 2\frac{\log(\theta)}{\log(1-\theta)} + \frac{\log[(\theta-1)\log(1-\theta)]}{\log(1-\theta)}\right), N), 1)$$

Naturally we seek to compare the two stopping times, and it is easiest to do so before we do any sort of rounding. We see the difference between the two is

$$D(\theta) = \frac{\log[(\theta-1)\log(1-\theta)] - \log\theta}{\log(1-\theta)}$$

By examining it's first derivative

$$\frac{d}{d\theta}D(\theta) = \frac{\theta\left(\log[(\theta-1)\log(1-\theta)] - \log\theta - 1\right) + \log(1-\theta)}{\theta(1-\theta)\log^2(1-\theta)}$$

We see that the function is increasing on the interval $(0,1)$. Additionally

$$\lim_{\theta\to 0} D(\theta) = \frac{1}{2}, \lim_{\theta\to 1} D(\theta) = 1$$

so we can be certain that pre-rounding, our two values are within 0.5 to 1 of each other. In practice we're even better off. $D(\theta)$ has an incredibly sharp peak singularity at 1, so it's is very hard to get a difference close to 1. For example, $D(0.9999) < 0.76$. Practical θ values have a $D(\theta)$ value closer to around 0.5. Additionally, depending on N, many $\alpha(\theta)$ and $\tau(\theta)$ values are rounded down to N for high θ and rounded up to 1 for low θ. As the data below will show, many θ values make for uninteresting results. Here we provide $\tau(\theta)$ for some values of θ

θ	$\tau(\theta)$	θ	$\tau(\theta)$
0.05	1	0.05	1
0.10	31	0.06	7
0.15	41	0.07	16
0.20	45	0.08	22
0.25	48	0.09	27
0.30	49	0.10	31
0.35	50	0.11	34
0.40	51	0.12	36
0.45	51	0.13	38
0.50	51	0.14	39
0.55	52	0.15	41
0.60	52	0.16	42
0.65	52	0.17	43
0.70	52	0.18	44
0.75	52	0.19	45
0.80	52	0.20	45
0.85	52	0.21	46
0.90	52	0.22	47
0.95	52	0.23	47

I also tested the greedy $\tau(\theta)$ for all θ from $(0, 1)$ incrementing by 0.0001, and excitingly is was the optimal time to stop for every single value (given our expression for expectation). This leads to the conjecture we provide in our final section. Additionally for all 10000 values of θ tested, $\alpha(\theta)$ and $\tau(\theta)$ differ only 94 times. In each case $\alpha(\theta) - \tau(\theta) = 1$, and in most cases $\alpha(\theta)$ is very close to rounding to the right value, but is slightly too large. Note that this is a great indication that $\tau(\theta)$ is the best time, and also captures valuable information about the shape of our expectation function.

2.4 Results on Expectation and Variance

Here we provide a big table of results of expectation and variance where we use $N = 52$ and the greedy stopping time.

θ	$\mathbf{E}(C)$	$\mathrm{Var}(C)$
0.01	51.4492790258	0.533712249501
0.02	50.9229455124	1.01019430643
0.03	50.4122906983	1.44518752516
0.04	49.9154417363	1.83077270391
0.05	49.4315054748	2.18918007992
0.06	48.9599394339	2.5064927682
0.07	48.5003866943	2.78724941749
0.08	48.0525910927	3.05069977784
0.09	47.6163855167	3.26768052166
0.1	47.1915229098	3.45094969776
0.11	46.7779528474	3.61583035561
0.12	46.3757834475	3.77928872752
0.13	45.9845805732	3.89946084838
0.14	45.6041462305	4.0428208458
0.15	45.2352582429	4.08362084304
0.16	44.8770829678	4.16122213665
0.17	44.5297395881	4.2105201199
0.18	44.1931804496	4.23272740027
0.19	43.8669979811	4.22952217903
0.2	43.55222784	4.30097552313
0.21	43.2483609101	4.25485970136
0.22	42.9539443115	4.29722791755
0.23	42.6720776199	4.21385528033
0.24	42.3993000714	4.23035982358
0.25	42.1376953125	4.11739253998
0.26	41.8876993376	4.11125664574
0.27	41.6460928407	4.10000919472
0.28	41.41606144	3.95447013818
0.29	41.19858319	3.92543231512
0.3	40.9899	3.89352999
0.31	40.79002879	3.85939878832
0.32	40.600768	3.69332083098
0.33	40.426437	3.64767423923
0.34	40.261304	3.60176616518
0.35	40.105375	3.55613923437
0.36	39.958656	3.51130248806
0.37	39.821153	3.46773140639
0.38	39.692872	3.42586793082
0.39	39.5808	3.26931696
0.4	39.48	3.2256
0.41	39.3888	3.18495216
0.42	39.3072	3.14770176
0.43	39.2352	3.11414256
0.44	39.1728	3.08453376
0.45	39.12	3.0591
0.46	39.0768	3.03803136
0.47	39.0432	3.02148336
0.48	39.0192	3.00957696
0.49	39.0048	3.00239856
0.5	39.0	3.0

θ	$\mathbf{E}(C)$	$\mathrm{Var}(C)$
0.51	39.015	3.00249852
0.52	39.04	3.00997632
0.53	39.075	3.02238012
0.54	39.12	3.03962112
0.55	39.175	3.061575
0.56	39.24	3.08808192
0.57	39.315	3.11894652
0.58	39.4	3.15393792
0.59	39.495	3.19278972
0.6	39.6	3.2352
0.61	39.715	3.28083132
0.62	39.84	3.32931072
0.63	39.975	3.38022972
0.64	40.12	3.43314432
0.65	40.275	3.487575
0.66	40.44	3.54300672
0.67	40.615	3.59888892
0.68	40.8	3.65463552
0.69	40.995	3.70962492
0.7	41.2	3.7632
0.71	41.415	3.81466812
0.72	41.64	3.86330112
0.73	41.875	3.90833532
0.74	42.12	3.94897152
0.75	42.375	3.984375
0.76	42.64	4.01367552
0.77	42.915	4.03596732
0.78	43.2	4.05030912
0.79	43.495	4.05572412
0.8	43.8	4.0512
0.81	44.115	4.03568892
0.82	44.44	4.00810752
0.83	44.775	3.96733692
0.84	45.12	3.91222272
0.85	45.475	3.841575
0.86	45.84	3.75416832
0.87	46.215	3.64874172
0.88	46.6	3.52399872
0.89	46.995	3.37860732
0.9	47.4	3.2112
0.91	47.815	3.02037372
0.92	48.24	2.80468992
0.93	48.675	2.56267452
0.94	49.12	2.29281792
0.95	49.575	1.993575
0.96	50.04	1.66336512
0.97	50.515	1.30057212
0.98	51.0	0.90354432
0.99	51.495	0.470594519999

2.5 Optimally and Generalizing to N-shuffles

First we provide the following conjecture

Conjecture 2.5. Strategy 2.3 is the optimal guessing strategy in terms of maximizing expectation for one overhead shuffle.

The motivation for this conjecture comes from the fact that guessing cards simply comes down to guessing what cards make the top of a stack. It seems very natural that the above strategy maximizes the expectation of the number of tops of stacks you get, especially considering that it always outperforms or does equivalently well as the solution given by maximizing our first order conditions. In fact, we should expect that the expectation of the next top of a stack plays a huge role in maximizing the function we've provided on integer support. How to prove this I am not yet sure.

As a quick note about n-overhead shuffles, I tried but wasn't able to discern anything too meaningful. It appears to be a much more complex problem, and I believe exponentially harder to guess. A good way to model it is using vectors of Bernoulli parameters $\mathbf{X}_k = (X_k^{(1)}, \ldots, X_k^{(n)})$ to model the shuffle. It appears that any pairings of 1s, despite what slot they are in in the vector will make a new stack. The order of these stacks in the final deck is quite confusing, and the greedy approach of guessing the top of the next stack doesn't appear to be too fruitful or intelligent. The best I can say is guess a card, continue guessing the remaining cards in its stack until you are incorrect, and then move on to the next stack. The more vectors of the form $(0, \ldots, 0)$, the more cards you'll end up getting right. Definitely more work to be done here.

3 The Riffle Shuffle

The riffle shuffle is likely the most common shuffle and thus the most thoroughly mathematically examined. It is defined as follows.

Definition 3.1. Suppose we have a N card deck. We cut the deck at card c with probability

$$\mathbf{P}(\text{cut } = c) = \frac{\binom{N}{c}}{2^N}$$

We now have two stacks, one of size A and one of size B. We drop cards one at a time from each stack until we exhaust the deck. The probability of dropping from a stack is proportional to its size. In particular

$$\mathbf{P}(\text{drop } A) = \frac{A}{A+B}$$

,

$$\mathbf{P}(\text{drop } B) = \frac{B}{A+B}$$

3.1 Developing an Optimal Guessing Strategy

Suppose at some-point during our shuffle we have dropped r cards and there are k cards in the pile containing the top most card of the un-shuffled deck. We label this event $S_{r,k}$. From the way we cut the deck it is easy to see that

$$\mathbf{P}(S_{0,k}) = \begin{cases} 0 & \text{if } k = 0 \\ \frac{\binom{N}{k}}{2^N} & \text{if } k \in \{1, \ldots, N-1\} \\ \frac{1}{2^{N-1}} & \text{if } k = N \end{cases}$$

From this point it is pretty easy to compute $\mathbf{P}(S_{r,k})$ in the following fashion. In the case that $r > 0$ and $k \in \{0, \ldots N - r\}$

$$\mathbf{P}(S_{r,k}) = \mathbf{P}(S_{r-1,k+1})\frac{k+1}{N-r+1} + \mathbf{P}(S_{r-1,k})(1 - \frac{k}{N-r+1})$$

and obviously $\mathbf{P}(S_{r,k}) = 0$ if $k > N - r$. Now suppose we our guessing the order of our shuffled deck. We know the deck is composed of two interwoven stacks, and we know the top of one stack is the top card of the un-shuffled deck. We don't know what the top of the second stack is, but we do have some idea of what it could be. In contrast to the overhead shuffle, if we labelled our deck with the top card being 1 and the bottom card being N, then we have from the cut that if N cards have yet to be dropped ($r = 0$) then for $i \in \{2, \ldots N\}$

$$\mathbf{P}(\text{card } i \text{ of un-shuffled deck is top of other pile}) = \frac{\binom{N}{i-1}}{2^N}$$

As we're guessing, if some card from the second pile comes up, we immediately know that it's the top card of the second pile. In the case that we have yet to see the top card of the second pile, we have to adjust our distribution based on how many cards have been dropped. Suppose there are still r cards remaining for us to guess and we have yet to see a card from the second pile. Then for $i \in \{2 + r, \ldots N\}$ we have that

$$\mathbf{P}(\text{card } i \text{ of un-shuffled deck is top of other pile} | r \text{ cards remaining to guess}) = \frac{\binom{N}{i-1}}{2^N} / \sum_{k=N-r+1}^{N} \frac{\binom{N}{k-1}}{2^N}$$

Now we start our guessing argument. Suppose we let I_i be the indicator that we guessed the card i correctly. Then we seek to find a guessing strategy that maximizes the expectation of the sum $C = \sum_{i=1}^{N} I_i$. It suffices then to maximize the expectation of each individual $\mathbf{E}(I_i) = \mathbf{P}(\text{guess card } i \text{ correctly})$.

The obvious first step is to guess the top card as our first card. This results in $\mathbf{E}(I_1) = \frac{1}{2}$, and one can easily see there is no better option. Suppose now we are guessing the jth card of the shuffled deck. If we are guessing the jth card, there will be $r = N - j + 1$ cards remaining to be guessed. Thus we are in some state $S_{N-j+1,k}$. We examine some cases.

Case One. The first case we examine is when there are no cards in the other pile, so we have yet to discover what the top card of the second pile is. In this case we have two options. Guess the next card we expect from the first pile, or make a stab at the top card in the second pile. Note we are thus currently in the state $S_{N-j+1,j-1}$. The chance that the jth card goes to the pile with the top card (imagine reverse shuffling) in it is exactly $\mathbf{P}(S_{N-j,j} | S_{N-j+1,j-1})$. In this case we know what the jth card is and can guess it w,.p.1. In fact it is exactly the jth card of the un-shuffled deck. We also must consider the possibility that the jth card is the top of the next pile, which is $\mathbf{P}(S_{N-j,j-1} | S_{N-j+1,j-1})$. However in this case, since there are no cards in the other pile, we aren't completely sure what card to guess. We do know that

$$\mathbf{P}(\text{card } i \text{ of un-shuffled deck is top of other pile}|r \text{ cards remaining to guess}) = \frac{\binom{N}{i-1}}{2^N} / \sum_{k=N-r+1}^{N} \frac{\binom{N}{k-1}}{2^N}$$

Thus in the chance we guess un-shuffled card j our probability of being correct is

$$\mathbf{P}(S_{N-j,j}|S_{N-j+1,j-1}) + \mathbf{P}(S_{N-j,j-1}|S_{N-j+1,j-1})\left(\frac{\binom{N}{j-1}}{2^N} / \sum_{k=j}^{N} \frac{\binom{N}{k-1}}{2^N}\right)$$

Otherwise we pick un-shuffled card i such that

$$i = \underset{i \in \{j+1,\ldots,N\}}{\arg\max}\, \mathbf{P}(S_{N-j,j-1}|S_{N-j+1,j-1})\left(\frac{\binom{N}{i-1}}{2^N} / \sum_{k=j}^{N} \frac{\binom{N}{k-1}}{2^N}\right)$$

in which case the probability of being correct is

$$\max_{i \in \{j+1,\ldots,N\}} \mathbf{P}(S_{N-j,j-1}|S_{N-j+1,j-1})\left(\frac{\binom{N}{i-1}}{2^N} / \sum_{k=j}^{N} \frac{\binom{N}{k-1}}{2^N}\right)$$

To make our final decision we either pick i or j from above, which ever one gives us the more likely outcome of being correct.

Case Two. The second case we examine is when there are indeed cards in the pile not containing the top card of the un-shuffled deck. In this case we know the card we're guessing can only be two things, the next card in either pile. At this point we are in some state $S_{N-j+1,m}$ where $0 \leq m < j-1$. We thus simply make a decision regarding which pile the card goes to. If we make this decision optimally we will choose the pile such that the probability of us guessing the card right is

$$\max(\mathbf{P}(S_{N-j,m}|S_{N-j+1,m}), \mathbf{P}(S_{N-j,m+1}|S_{N-j+1,m}))$$

With these two cases, we can write the expectation of guessing the card j where $j > 1$ as

$$\begin{aligned}\mathbf{E}(I_j) = \max\Bigg(&\mathbf{P}(S_{N-j,j}|S_{N-j+1,j-1}) + \mathbf{P}(S_{N-j,j-1}|S_{N-j+1,j-1})\left(\frac{\binom{N}{j-1}}{2^N} / \sum_{k=j}^{N} \frac{\binom{N}{k-1}}{2^N}\right),\\ &\max_{i \in \{j+1,\ldots,N\}} \mathbf{P}(S_{N-j,j-1}|S_{N-j+1,j-1})\left(\frac{\binom{N}{i-1}}{2^N} / \sum_{k=j}^{N} \frac{\binom{N}{k-1}}{2^N}\right)\Bigg)\mathbf{P}(S_{N-j+1,j-1})\\ &+ \sum_{m=0}^{j-2} \max(\mathbf{P}(S_{N-j,m}|S_{N-j+1,m}), \mathbf{P}(S_{N-j,m+1}|S_{N-j+1,m}))\mathbf{P}(S_{N-j+1,m})\end{aligned}$$

Recognizing that $P(A|B)P(B) = P(B|A)P(A)$, we can further simplify the above

$$\mathbf{E}(I_j) = \max\left(\mathbf{P}(S_{N-j,j}) + \frac{1}{j}\mathbf{P}(S_{N-j,j-1})\Big(\frac{\binom{N}{j-1}}{2^N}/\sum_{k=j}^{N}\frac{\binom{N}{k-1}}{2^N}\Big), \frac{1}{j}\mathbf{P}(S_{N-j,j-1})\max_{i\in\{j+1,\ldots,N\}}\left(\frac{\binom{N}{i-1}}{2^N}/\sum_{k=j}^{N}\frac{\binom{N}{k-1}}{2^N}\right)\right)$$
$$+\sum_{m=0}^{j-2}\max((1-\frac{m}{j})\mathbf{P}(S_{N-j,m}), \frac{m+1}{j}\mathbf{P}(S_{N-j,m+1}))$$

Meaning we can write our optimal expectation from our guessing strategy as

$$\mathbf{E}(C) = \frac{1}{2} + \sum_{j=2}^{N}\Bigg(\max\left(\mathbf{P}(S_{N-j,j}) + \frac{1}{j}\mathbf{P}(S_{N-j,j-1})\Big(\frac{\binom{N}{j-1}}{2^N}/\sum_{k=j}^{N}\frac{\binom{N}{k-1}}{2^N}\Big),\right.$$
$$\left.\frac{1}{j}\mathbf{P}(S_{N-j,j-1})\max_{i\in\{j+1,\ldots,N\}}\left(\frac{\binom{N}{i-1}}{2^N}/\sum_{k=j}^{N}\frac{\binom{N}{k-1}}{2^N}\right)\right) + \sum_{m=0}^{j-2}\max((1-\frac{m}{j})\mathbf{P}(S_{N-j,m}), \frac{m+1}{j}\mathbf{P}(S_{N-j,m+1}))\Bigg)$$

This is the expectation of the following esoteric guessing strategy

Strategy 3.2. For the first card guess the top card of the un-shuffled deck. After, if the top of the pile not containing the top card of the un-shuffled deck has yet to be revealed, determine whether or not it is to your advantage to guess that it is coming up next. If it is guess it to the best of your ability, if not guess the next card in the pile containing the top card. Once the top of both piles is revealed, simply guess whichever card is the next for the more likely pile to be added to.

Theorem 3.3. Strategy 3.2 is the optimal guessing strategy with respect to expectation for one riffle shuffle. It has expectation

$$\frac{1}{2} + \sum_{j=2}^{N}\Bigg(\max\left(\mathbf{P}(S_{N-j,j}) + \frac{1}{j}\mathbf{P}(S_{N-j,j-1})\Big(\frac{\binom{N}{j-1}}{2^N}/\sum_{k=j}^{N}\frac{\binom{N}{k-1}}{2^N}\Big),\right.$$
$$\left.\frac{1}{j}\mathbf{P}(S_{N-j,j-1})\max_{i\in\{j+1,\ldots,N\}}\left(\frac{\binom{N}{i-1}}{2^N}/\sum_{k=j}^{N}\frac{\binom{N}{k-1}}{2^N}\right)\right) + \sum_{m=0}^{j-2}\max((1-\frac{m}{j})\mathbf{P}(S_{N-j,m}), \frac{m+1}{j}\mathbf{P}(S_{N-j,m+1}))\Bigg)$$

Proof. See computations above. Optimality is obvious by construction. □

To generalize to n-riffle shuffles one can simply optimize the above computational approach to maximize the expectation of each card. Note that a n-fold riffle-shuffle is equivalent to splitting up our deck into multiple stacks and dropping cards from multiple stacks accordingly rather than just two, so the computations carry over naturally. I believe the most important course of further action is to compare this result to McGrath's proposal, which is certainly a more practical but likely sub-optimal approach.

References

[1] Mathematical Developments from the Analysis of Riffle-Shuffling. In A. Ivanov, A. Fuanou & M. Liebeck (eds.), Groups, Combinatorics and Geometry, pp.73-97, World Scientific, N.J. (2003)

[2] Trailing the Dovetail Shuffle to its Lair. P. Diaconis D. Bayer, Ann. Appl. Prob., 2(2):294-313.