

トランプのシャッフルにおける可視化と最適な組み合わせに関する検討 Study on Visualization and Optimal Combination in Card Shuffling

井手 広康[†]
Hiroyasu Ide

奥田 隆史[‡]
Takashi Okuda

1. はじめに

カードゲームにおいて使用される遊戯用カードには、トランプ、花札、UNO、トレーディングカードなどさまざまな種類が存在する。このようなカードを使用したゲーム内において、プレイヤーが次に引くカードが予測できないようデッキ（山札）をシャッフルすることが通常である。このようにシャッフルには“デッキの並びを不規則にすることで積み込み等の不正を防止し、ゲームの公平性を担保する”という目的がある [1]。シャッフルにはさまざまな手法が存在するが、どの手法を用いてどれくらいシャッフルすればデッキの並びが不規則になるのか一般的には知られていない。

しかしシャッフルに関してカードの変動距離を用いて解析した研究 [2]-[7] がいくつか存在する。例えばリフル・シャッフルでは 7 回あたりで急にデッキがよく混ざるということが明らかとなっており、これはマルコフ連鎖におけるカット・オフ現象と呼ばれている。一方日本においてシャッフルに関する文献は僅少であるが、複数のシャッフルを組み合わせたシミュレーション結果から 2 値（白、黒）のビットマップ画像へと可視化し、効果的な組み合わせについて分析した研究 [8] がある。ただしこの研究ではシャッフルの組み合わせの特徴に関して言及されているが、シャッフル結果に関して定量化されていないため、どの組み合わせがもっとも効果的であるのか結論付けられていない。

そこで本研究では先行研究 [8] をベースに主に次の 2 つの変更点を加えた。ひとつ目はシャッフルの可視化に 2 値（白、黒）ではなく 3 値（赤、白、青）を用いた点である。先行研究 [8] で用いられた 2 値ではカードの「偏りやすい位置」しか分からなかったが、3 値を用いることで「偏りにくい位置」についても視覚的に判断することができる。ふたつ目はシミュレーションからデッキの収束率を定義し、シャッフル結果を定量化した点である。これにより複数のシャッフル結果（単独のシャッフルあるいは複数の組み合わせ）の優越を比較することができる。本研究の目的はシャッフル結果の可視化と定量化を行い、どのシャッフルの組み合わせがもっとも効果的であるか検討することである。

以下、2 章では本研究で取り扱う 3 つのシャッフル手法について説明する。3 章ではシャッフル結果の可視化と定量化の手順について説明する。4 章では単独のシャッフル試行について考察する。5 章では効果的な複数のシャッフルを組み合わせについて考察する。6 章では本研究のまとめと今後の課題について述べる。

2. 一般的なシャッフル手法

本章では一般的に広く使用されているシャッフル手法について説明する。まずデッキのカード枚数を n と置き、デッキの上から順に $i = \{1, 2, \dots, n\}$ を与える。次にカード i をデッキの上から $j_i = \{1, 2, \dots, n\}$ 番目に並べ替える操作を置換 π とする。このとき $\pi: \{1, 2, \dots, n\} \rightarrow \{1, 2, \dots, n\}$ は全単射であり、

$$\pi = \begin{pmatrix} 1 & 2 & \cdots & n \\ j_1 & j_2 & \cdots & j_n \end{pmatrix} \quad (1)$$

と表すことができる [9]。

以下の節ではこの表記を用いて 3 つのシャッフル手法（ヒンズー・シャッフル、リフル・シャッフル、ディーラー・シャッフル）について説明する。なお本稿では特に説明がない限り、デッキはジョーカーを省く 52 枚のトランプ ($n = 52$) を表す。またシャッフルの過程においてデッキをいくつかに分割したものをパケットと呼び、任意のパケットのカード枚数を u と置く。

2.1. ヒンズー・シャッフル

ヒンズー・シャッフル (Hindu Shuffle) ではまず右手にデッキを持ち、デッキの上から $u = 10$ 程度のパケットを左手で抜き取り移動させる。次も同様に右手に残ったデッキの上から左手で同程度のパケットを抜き取り、先程抜き取った左手のパケットの上に重ねる。この動作を右手のデッキがなくなるまで繰り返す。例えば $n = 10$ に対するヒンズー・シャッフルは次式の置換を行う ($u = 3$ の場合)。

$$\pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 8 & 9 & 10 & 5 & 6 & 7 & 2 & 3 & 4 & 1 \end{pmatrix} \quad (2)$$

ヒンズー・シャッフルは日本を含め広く東洋で行われるシャッフルであり、「ヒンズー」という名は欧米においてインドの奇術家たちがこのシャッフルをよく用いたことに由来する。しかしヒンズー・シャッフルは短時間で行うことができる反面、ある程度の回数ではカードがよく切り混ざらないという欠点もある [1]。

2.2. リフル・シャッフル

リフル・シャッフル (Riffle Shuffle) ではまずデッキを 2 つのパケットに分割し、右手と左手でそれぞれのパケットを持つ。次にそれぞれのパケットを手で反らしながらカードをリフル（パラパラと落とすこと）させ交互に重ね合わせていく。例えば $n = 10$ に対するリフル・シャッフルは次式の置換を行う ($u = 5$ の場合)。

$$\pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 1 & 3 & 5 & 7 & 9 & 2 & 4 & 6 & 8 & 10 \end{pmatrix} \quad (3)$$

リフル・シャッフルではこのように 2 つに分割したパケットから交互にカードが挿入されるため、デッキの

[†]愛知県立大学 大学院 情報科学研究科, Graduate School of Information Science and Technology, Aichi Prefectural University

[‡]愛知県立大学 情報科学部 情報科学科, Department of Information Science and Technology, Faculty of Information Science and Tehnology, Aichi Prefectural University

中心部に位置するカードほど移動距離が長くなるという性質がある。ただし完全にカードを交互にすること(これをパーフェクト・シャッフルという)は熟練した技術を要するため、多くの場合ある程度のカードが重なった状態でリフルされることになる。

2.3. ディール・シャッフル

ディール・シャッフル (Deal Shuffle) ではまずデッキの上から 1 枚ずつ d 箇所カードを順番に配置していく (2 周目以降はポケットの上に重ねる)。次にこの動作を繰り返すすべてのカードを配り終わったら、 d 個のポケットを任意の順序で重ねて 1 つのデッキにまとめる。例えば $n = 10$ に対するディール・シャッフルは次式の置換を行う ($d = 3$, カードを配布した順序でポケットを重ねた場合)。

$$\pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 4 & 7 & 10 & 3 & 6 & 9 & 2 & 5 & 8 & 1 \end{pmatrix} \quad (4)$$

ディール・シャッフルを行うにはある程度のスペースが必要であり、シャッフルに時間がかかるという欠点がある。またシャッフル結果が $\{n, d, \text{ポケットを重ねる順序}\}$ によって一意に決定されるという性質がある。このうち「ポケットを重ねる順序」に特に決まりはなく、全部で $d!$ 通りの順序が存在する。

3. 可視化手順と定量化手法

2 章で説明した 3 つのシャッフル手法をシミュレーションに実装する。ここでは「単独のシャッフルを m 回繰り返して試行する一連の操作」あるいは「複数のシャッフルを m 回組み合わせる一連の操作」を 1 回のシミュレーションとし、 S 回の独立したシャッフル結果を合算したデータに対して可視化および定量化を行う (1 回ごとにデッキは初期化される)。なおシミュレーションの開発言語には Python を使用した。

以下の節ではシミュレーション結果を可視化するための 3 つの手順と定量化手法について説明する。

3.1. 手順 1: n 次正方行列 P_S の算出

まず S 回のシャッフル結果において「デッキの i 番目のカードが j 番目に移動した累積値」を $t_{i,j}$ ($0 \leq t_{i,j} \leq S$) と置き、1 回のシミュレーションごとに該当する n 箇所すべての $t_{i,j}$ に 1 を加算する。例えば図 1 のようにデッキの 1 番目 ($i = 1$) のカード $\clubsuit A$ が 5 番目 ($j = 5$)

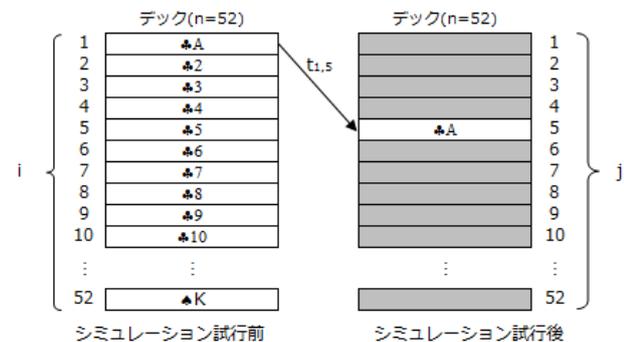


図 1: 1 回のシミュレーションにおける $t_{i,j}$ の加算

へ移動した場合、 $t_{1,5}$ に 1 を加算する。このようにして S 回のシャッフル結果から $t_{i,j}$ を得ることができる。次にデッキ全体の $t_{i,j}$ を n 次正方行列 T_S と置き、

$$T_S = \begin{pmatrix} t_{1,1} & t_{1,2} & t_{1,3} & \dots & t_{1,n} \\ t_{2,1} & t_{2,2} & t_{2,3} & \dots & t_{2,n} \\ t_{3,1} & t_{3,2} & t_{3,3} & \dots & t_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{n,1} & t_{n,2} & t_{n,3} & \dots & t_{n,n} \end{pmatrix} \quad (5)$$

として表す。さらに「デッキの i 番目のカードが j 番目に移動した頻度」は $t_{i,j}/S$ と表すことができ、これを $p_{i,j}$ ($0 \leq p_{i,j} \leq 1$) と置く。ここでデッキ全体の $p_{i,j}$ を n 次正方行列 P_S と置き、

$$P_S = \frac{T_S}{S} = \begin{pmatrix} p_{1,1} & p_{1,2} & p_{1,3} & \dots & p_{1,n} \\ p_{2,1} & p_{2,2} & p_{2,3} & \dots & p_{2,n} \\ p_{3,1} & p_{3,2} & p_{3,3} & \dots & p_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{n,1} & p_{n,2} & p_{n,3} & \dots & p_{n,n} \end{pmatrix} \quad (6)$$

として表す。このようにして S 回のシャッフル結果から n 次正方行列 P_S を算出する。

3.2. 手順 2: 評価基準となる ϵ の算出

あるシャッフル手法が「デッキの並びを不規則にできる」と仮定したとき、上記の P_S においてすべての $p_{i,j}$ は $1/n$ へ収束する (S が十分に大きい場合)。ここでシミュレーションでは Random 関数を用いることでこれを実現することができる (以下これをランダム・シャッフルと呼ぶ)。つまりランダム・シャッフルにおいてすべての $p_{i,j} \in P_\infty$ は均等に $1/n$ となる。これを利用して任意のシャッフル結果 P_∞ に対して非常に小さい正数 ϵ が与えられたとき、 $p_{i,j}$ が $1/n$ に対して ϵ 以上離れる確率からシャッフルの収束率 (後述) を求める。

まず任意のシャッフル結果 P_S の確率モデル Ω において、次のような事象 E を考える。

$$E = \left\{ p_{i,j} \text{ と } \frac{1}{n} \text{ との差が } \epsilon \text{ 以上} \right\} \quad (7)$$

次に確率モデル Ω におけるすべての標本点 ω に対してこの事象 E を集合の表記で表すと

$$E = \left\{ \left| p_{i,j} - \frac{1}{n} \right| \geq \epsilon \text{ となる } \omega \right\} \quad (8)$$

となる。このとき事象 E における確率 $p(E)$ に対して次式が成り立つ。

$$\lim_{S \rightarrow \infty} p(E) = 0 \quad (9)$$

しかし現実には $S \rightarrow \infty$ とすることは不可能であるため、 $S \rightarrow 10^4$ として式 (9) を次式に置き換える。

$$\lim_{S \rightarrow 10^4} p(E) = 0 \quad (10)$$

ここでランダム・シャッフルでは $\epsilon \geq 0.005$ であるとき式 (10) を満たすことがシミュレーションより明らかになった。そのため任意のシャッフル結果 P_{10^4} において $\epsilon = 0.005$ の条件下で式 (10) を満たすとき、これがランダム・シャッフルと相違ない結果であると評価する。

3.3. 手順 3 : 画像への変換と定量化手法

任意のシャッフル結果における $p_{i,j} \in P_{10^4}$ に対して手順 2 の式 (8) を変形させた判別式 (11)~(13) を当てはめ、判別結果からすべての $p_{i,j}$ を次に示す 3 つのパターン $\{1, 0, -1\}$ のいずれかに置換する。

パターン 1 : $1/n$ との差が ϵ 以上である場合

$p_{i,j}$ が次式を満たすとき、これを 1 に置換する。

$$\epsilon \leq p_{i,j} - \frac{1}{n} \quad (11)$$

これはランダム・シャッフルと比較して $p_{i,j}$ が十分に大きい場合である。つまりこの位置にカードが偏りやすい性質があることを示唆している。

パターン 2 : $1/n$ との差が $\pm\epsilon$ 範囲内にある場合

$p_{i,j}$ が次式を満たすとき、これを 0 に置換する。

$$-\epsilon < p_{i,j} - \frac{1}{n} < \epsilon \quad (12)$$

これはランダム・シャッフルと比較して $p_{i,j}$ が許容範囲内にある場合である。つまりこの位置にほぼ $1/n$ の頻度でカードが挿入される性質があることを示唆している。

パターン 3 : $1/n$ との差が $-\epsilon$ 以下である場合

$p_{i,j}$ が次式を満たすとき、これを -1 に置換する。

$$p_{i,j} - \frac{1}{n} \leq -\epsilon \quad (13)$$

これはランダム・シャッフルと比較して $p_{i,j}$ が十分に小さい場合である。つまりこの位置にカードが挿入されにくい (あるいはまったく挿入されない) 性質があることを示唆している。

次に $\{1\} \rightarrow \{\text{赤}\}$, $\{0\} \rightarrow \{\text{白}\}$, $\{-1\} \rightarrow \{\text{青}\}$ へとそれぞれの $p_{i,j}$ をビットマップ化し、 P_{10^4} 全体を $n \times n$ の 3 値ビットマップ画像へと変換する。そのため式 (6) の P_S と同様に、3 値ビットマップ画像の縦軸は「シミュレーション前の位置 i 」、横軸は「シミュレーション後の位置 j 」を表し、左上が $(i, j) = (1, 1)$ となる。

ここで 3 値ビットマップ画像における白ビット数を w と置き、収束率 C を次式として定義する。

$$C := \frac{\text{白ビット数 } w}{\text{全体のビット数 } n^2} \quad (14)$$

このようにシャッフル結果を取束率 C として定量化することで、複数のシャッフル結果を比較することが可能となる。本研究ではシャッフル結果 P_{10^4} において $C = 1.000$ を満たすとき、このシャッフル (単独または組み合わせ) がランダム・シャッフルと相違ない手法であり、デッキがランダムな状態になったと評価する。

4. 単独のシャッフル試行における結果

単独のシャッフル手法を用いて求めたシャッフル結果 P_{10^4} を 3 値ビットマップ画像へと変換し、それぞれの手法がもつ性質について考察する。4.1 節ではヒンズー・シャッフル、4.2 節ではリフル・シャッフル、4.3 節ではディール・シャッフルの単独試行の結果について述べる。

4.1. ヒンズー・シャッフルの単独試行

ヒンズー・シャッフルにおいてデッキから抜き出すカード枚数は正規分布 $N(10, 5/3)$ に従うものとする。このモデルに従いシャッフル結果 P_{10^4} を可視化した 3 値ビットマップ画像を図 2 に示す。図 2(1)-(45) におけるそれぞれのタイトルには「シャッフルの試行回数 m 」と「収束率 C (白ビット数 w)」を表記している (図 3, 図 4 も同様である)。また 3 つのシャッフルの試行回数 m と収束率 C との関係を図 5 に示す。

ヒンズー・シャッフルには 2.1 節で説明したとおり、左手で抜き出すパケットの位置がデッキの中心を対称に入れ替わるという性質がある。そのためシャッフル前にデッキの上部にあったカードはシャッフル後に下部に配置されやすく (またはその逆も然り)、デッキの中心部にあったカードほどその位置はあまり変化しない。このような性質をもつため図 2 を見ると、3 値ビットマップ画像の中心点を $(x, y) = (0, 0)$ としたとき、 m が奇数のときには $y = x$ 、 m が偶数のときには $y = -x$ に対してほぼ線対称となっていることがわかる。しかし m の増加とともにデッキの中心部から徐々に白色に近づき、 $m \simeq 10$ で中心部はほぼ $1/n \pm \epsilon$ へ収束している。さらに m が増加するとデッキの上・下部でも収束箇所が拡大していき、 $m = 45$ においてはじめて $C = 1.000(2704)$ となった。また図 5 を見ても、ヒンズー・シャッフルでは L に比例して C が緩やかな曲線を描いて上昇していることがわかる。

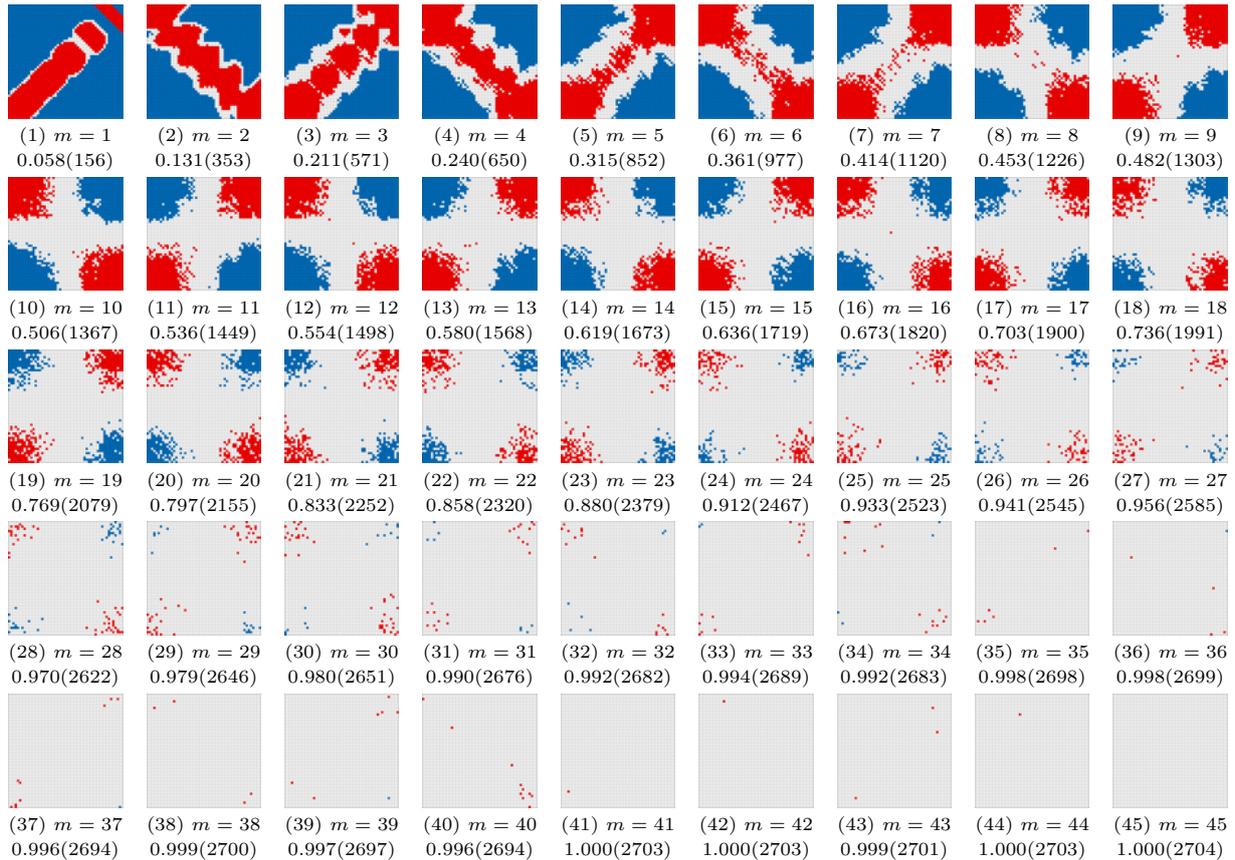
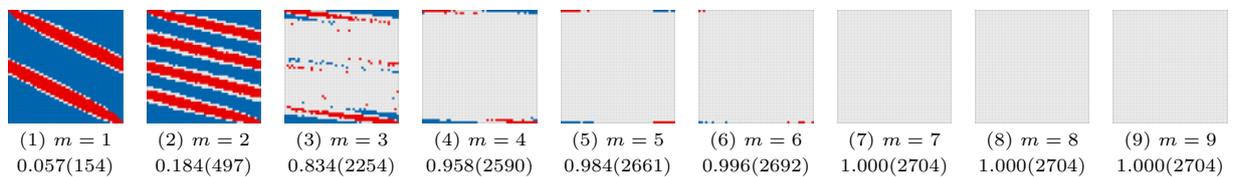
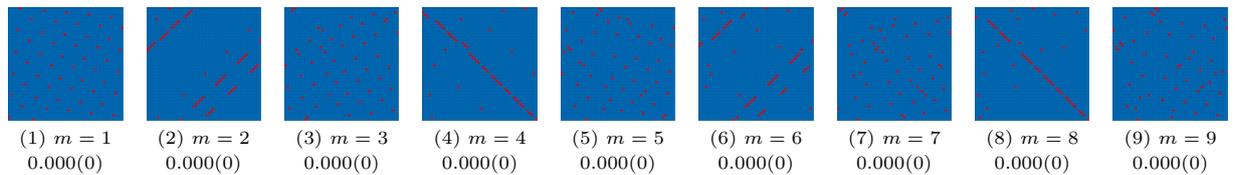
以上の結果から、ヒンズー・シャッフル単独ではシャッフルの効率は決して良くないが、おおよそ $m = 45$ 周辺においてデッキがランダムな状態になるといえる。

4.2. リフル・シャッフルの単独試行

リフル・シャッフルには **Gilbert-Shannon-Reeds** モデル [10][11] (以下これを **GSR** モデルと呼ぶ) を用いる。GSR モデルによるリフル・シャッフルの手順は次のとおりである。まずデッキを二項分布に従い 2 つのパケットに分割し、それぞれのカード枚数を a , b と置く。次に a からは $a/(a+b)$, b からは $b/(a+b)$ の確率でカードをリフルさせる。この操作を 2 つのパケットのカード枚数が 0 になるまで繰り返す。

この GSR モデルに従いシャッフル結果 P_{10^4} を可視化した 3 値ビットマップ画像を図 3 に示す。リフル・シャッフルには 2.2 節で説明したとおり 2 つのパケットから交互にカードが挿入されるという性質がある (GSR モデルでは完全に交互ではない)。そのため図 3(1) のように $m = 1$ では赤色のラインが 2 本現れる (交互にリフルするほどこのラインは細くなる)。これは「デッキの上部から中央部までのカード」と「デッキの中央部から下部までのカード」がそれぞれデッキ全体に広がって移動する傾向があることを示唆している。

さらに図 3(2) のように $m = 2$ ではラインが 4 本現れているため、 $m = 3$ ではラインが 8 本 (2^m) になると推測できる。しかし図 3(3) を見ると中心部の多くのラインは消滅し、 C が急激に上昇していることがわかる (図 5 から明らかである)。これは m の増加にともないそれぞれのラインの重なり合う面積が広くなり、この部分が $1/n \pm \epsilon$ へ収束しているためであると考え

図 2: ヒンズー・シャッフルにおける可視化と定量化 ($m = 1, 2, \dots, 45$)図 3: リフル・シャッフルにおける可視化と定量化 ($m = 1, 2, \dots, 9$)図 4: ディール・シャッフルにおける可視化と定量化 ($m = 1, 2, \dots, 9$)

られる。これ以降も m の増加とともに残っていた上・下部のラインも徐々に消滅していき、 $m = 7$ においてはじめて $C = 1.000(2704)$ となった。

以上の結果から、リフル・シャッフル単独の試行はヒンズー・シャッフルと比較して効率が良く、 $m = 3$ のとき C が急激に上昇し、その後 $m = 7$ においてデッキがランダムな状態となることがわかった。なお“7回の試行でデッキが良く混ざった”というシミュレーション結果は、リフル・シャッフルのカット・オフ現象に関する先行研究 [6][7] とも結果が一致している。

4.3. ディール・シャッフルの単独試行

ディール・シャッフルにおけるパケットの配置箇所 d は先行研究 [8] に合わせて $d = 10$ とする。ここでカードを配置した順番でそれぞれのパケットに $k = \{1, 2, \dots, d\}$ を割り振り、任意のパケットを d_k と表記する。このとき「 d 個のパケットを重ねる順序」についても先行研究 [8] に合わせて $\{k \text{ の奇数順} \} \rightarrow \{k \text{ の偶数順} \}$ と設定した。つまり $d = 10$ の場合においてパケット d_k は $\{d_1 \rightarrow d_3 \rightarrow d_5 \rightarrow d_7 \rightarrow d_9 \rightarrow d_2 \rightarrow d_4 \rightarrow d_6 \rightarrow d_8 \rightarrow d_{10}\}$ という順序で重ね合わせることになる。

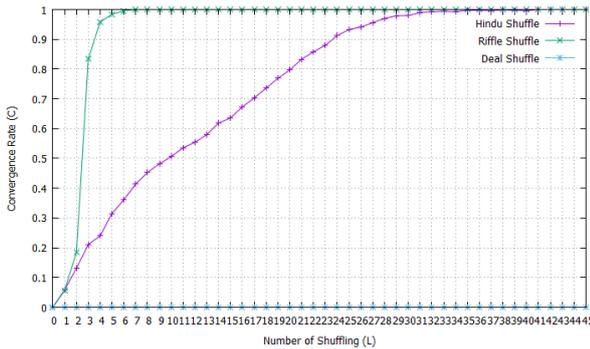


図 5: 各シャッフルの試行回数 m と収束率 C の関係

このモデルに従いシャッフル結果 P_{10^4} を可視化した 3 値ビットマップ画像を図 4 に示す。ディール・シャッフルには 2.3 節で説明したとおり、シャッフルの結果が $\{n, d, \text{パケットを重ねる順序}\}$ によって一意に決定されるという性質がある。そのためディール・シャッフル単独の試行にランダム性はまったくなく ($w = 0$)、図 4 のすべての m において $C = 0.000(0)$ となっていることがわかる (図 5 から明らかである)。また m が 3 つ置きに類似した分散傾向を示しており、 m が 4 の倍数のときにほぼデッキが元の状態 ($y = -x$ のライン) に戻るといった性質も読み取ることができる。

以上の結果から、ディール・シャッフルではカードを一定の規則に従って分散させることができるが、単独の試行にランダム性はまったくなく、他のシャッフルと組み合わせ使用することが望ましいといえる。

5. 複数のシャッフルを組み合わせた結果

4 章では単独のシャッフル試行における結果について考察し、リフル・シャッフルが最短でデッキをランダムな状態にできることが明らかとなった。続いて本章では複数のシャッフルを組み合わせた試行において、もっとも効果的な組み合わせについて考察する。

まずヒンズー・シャッフルを H、リフル・シャッフルを R、ディール・シャッフルを D と表記し、これらの集合を $A = \{H, R, D\}$ と置く。次に m 回のシャッフルを組み合わせたシミュレーション (シャッフルの重複可) において、 $r = \{1, 2, \dots, m\}$ 回目のシャッフルを $L_r \in A$ と置く。このとき m 回の組み合わせ A^m は次式のように直積集合で表すことができる。

$$A^m = \{(L_1, L_2, \dots, L_m) \mid L_1 \in A \wedge L_2 \in A \wedge \dots \wedge L_m \in A\} \quad (15)$$

また $(L_1, L_2) \neq (L_2, L_1)$ であり、二項演算として可換でないことは「シャッフル」という操作の性質からも明らかである。以下この直積集合の表記において、 m 回のシャッフルの組み合わせである (L_1, L_2, \dots, L_m) を $L_1 L_2 \dots L_m$ と略記する。また図の説明において例えば「図 6(2)HR」を「(2)HR」のように略記する。

以下 5.1 節では $A^2 (m = 2)$ 、5.2 節では $A^3 (m = 3)$ 、5.3 節では $A^4 (m = 4)$ 、5.4 節では $A^5 (m = 5)$ による試行でのもっとも効果的なシャッフルの組み合わせについて考察する。

5.1. 2 回の組み合わせ A^2 による試行

$m = 2$ とした場合のシャッフルの組み合わせは A^2 の 9 通りである。すべての組み合わせにおける 3 値ビットマップ画像を図 6 に示す。図 6 を見ると C について (2)HR がもっとも高いことがわかる。ここで H と R の順序に着目した場合、これを逆にした (4)RH の C は (2)HR の 1/3 程度まで低下している。ただし (3)HD と (7)DH、(6)RD と (8)DR のように組み合わせに D を含む場合はいずれも C は低く、順序を逆にしても C に大きな影響は見られない (模様の変化は見られる)。

つまり A^2 において D を含むことは効果的でないということがいえる。またこのような D を含む組み合わせでは、 $L_1 = D$ では水平方向のライン、 $L_m = D$ では垂直方向のラインが現れることが特徴である (この現象は $m \geq 3$ の場合にも当てはまる)。

以上の結果から、 A^2 では (2)HR の組み合わせがもっとも効果的であることがわかった。しかし A^2 のすべての組み合わせにおいてデッキ全体に大きな偏りがあるため、2 回の組み合わせだけではまだシャッフルの効果は低いということがいえる。

5.2. 3 回の組み合わせ A^3 による試行

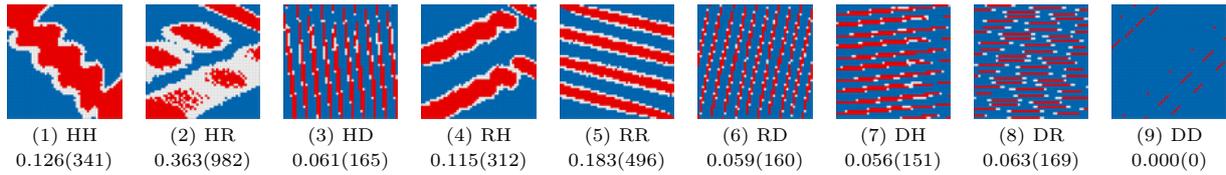
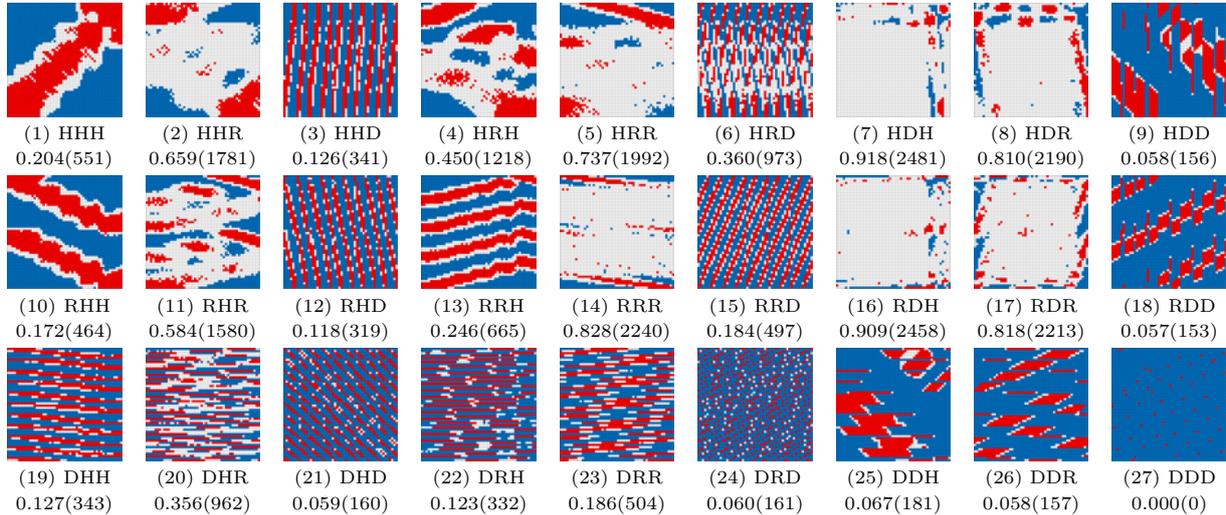
$m = 3$ とした場合のシャッフルの組み合わせは A^3 の 27 通りである。すべての組み合わせにおける 3 値ビットマップ画像を図 7 に示す。図 7 の 1 行目 (1)-(9) は図 6 の先頭に H を加えたもの、図 7 の 2 行目 (10)-(18) は図 6 の先頭に R を加えたもの、図 7 の 3 行目 (19)-(27) は図 6 の先頭に D を加えたものと見なすことができる。そのため基本的には 4 章で説明した H、R、D それぞれの性質が A^2 に対して反映された結果となっている。

しかし (7)HDH、(8)HDR、(16)RDH、(17)RDR のように、 A^2 と比較して格段に C が向上している組み合わせが存在していることがわかる (特に (7)(16) は $C \geq 0.9$ である)。興味深いことに、前節では「 A^2 において D を含むことが効果的でない」ということがわかったが、上記の 4 つの組み合わせには共通して L_2 に D が出現している。ただし L_1 あるいは L_m に D が出現すると、やはりデッキ全体が偏り C は低いままである。特に図 6 と図 7(19)-(27) を比較すると、模様は大きく変わっているが C はほとんど変化していないことがわかる。このように単独のシャッフルでは常に $C = 0.000(0)$ となる D であったが、他のシャッフルで挟み込んで実行すると C が大幅に向上することがわかった。

以上の結果から、 A^3 では (7)HDH、(16)RDH の組み合わせがもっとも効果的であることがわかった。また D を他のシャッフルで挟み込んで実行する組み合わせが効果的であることも明らかとなった。しかし 3 回の組み合わせでもデッキに部分的な偏りが見られるため、まだシャッフル回数が不十分であるといえる。

5.3. 4 回の組み合わせ A^4 による試行

$m = 4$ とした場合のシャッフルの組み合わせは A^4 の 81 通りである。すべての組み合わせにおける 3 値ビットマップ画像を図 8 に示す。図 8 の 1~3 行目 (1)-(27) は図 7 の先頭に H を加えたもの、図 8 の 4~6 行目 (28)-(54) は図 7 の先頭に R を加えたもの、図 8 の

図 6: シャッフルの組み合わせ $A^2(m=2)$ における可視化と定量化 (9 通り)図 7: シャッフルの組み合わせ $A^3(m=3)$ における可視化と定量化 (27 通り)

7~9 行目 (55)-(81) は図 7 の先頭に D を加えたものと見なすことができる。そのため基本的には 4 章で説明した H, R, D それぞれの性質が A^3 に対して反映された結果となっている。

しかし図 8 においてこれらの性質が読み取れないほどに C が格段に向上している組み合わせが存在する。ひとつは (19)-(24), (46)-(51) のように $L_2 = D$ もしくは $\{L_2, L_4\} = D$ の組み合わせであり、前者の組み合わせにおいて特に C が高いことがわかる。もうひとつは (7)(8), (16)(17), (34)(35), (43)(44) のように $L_3 = D$ かつ D が 1 度しか出現しない組み合わせである。つまり A^4 において L_1 と L_4 を省いて D を 1 度だけ含む組み合わせが効果的であるといえる。一方 $L_1 = D$ である組み合わせ (7~9 行目) や $L_4 = D$ である組み合わせ (3, 6, 9 列目) では、一部の組み合わせを省き A^3 と比較して C はほとんど向上していない。

このように A^4 において C が高かった組み合わせは複数存在するが、その中でも $C \geq 0.985$ を満たした組み合わせは (16)HRDH, (19)HDHH, (20)HDHR の 3 通りであった。ここでもやはり L_1 と L_4 を省いて D を 1 度だけ含んでいることがわかる。さらにこれらには共通して「すべて H から始まっている ($L_1 = H$)」, 「すべて D の直後に H が出現している」ということもいえる。これらは A^4 の場合だけに留まらず、前述した A^3 に加えて後述する A^5 にもほぼ当てはまる共通点である。

以上の結果から、 A^4 では (16)HRDH, (19)HDHH, (20)HDHR の組み合わせがもっとも効果的であることがわかった。また L_1 と L_4 を省いて D を 1 度だけ実

行する組み合わせが効果的であり、特に「 $L_1 = H$ かつ D の直後に H が出現する組み合わせ」であるほど C が高くなる傾向にあることも明らかとなった。しかしこの段階でも $C = 1.000$ となる組み合わせは存在せず、デッキを完全にランダムな状態にするためにはまだシャッフル回数が不十分であるといえる。

5.4. 5 回の組み合わせ A^5 による試行

$m=5$ とした場合のシャッフルの組み合わせは A^5 の 243 通りである。すべての組み合わせにおけるシミュレーションの結果、 $C \geq 0.9$ では 112 通り、 $C \geq 0.99$ では 38 通り、 $C \geq 0.999$ では 11 通り、 $C = 1.000$ では 5 通りもの組み合わせが存在することがわかった。そのためここでは紙面の関係上、3 値ビットマップ画像は記載せず $C \geq 0.999$ かつ $w \geq 2700$ を満たした上位 11 通りの組み合わせである (1)HHDHH, (2)HHDHR, (3)HRDHH, (4)HRDHR, (5)HDHHR, (6)HDHRR, (7)HDHDH, (8)HDHDR, (9)RHDHH, (10)RHDHR, (11)RDHDH について考察する。なおこれらの 3 値ビットマップ画像はほぼ白一色である。

ここで本来であれば $C = 1.000(2704)$ である組み合わせのみを取り上げるべきである。しかし本研究のシミュレーションでは $S = \infty$ ではなく $S = 10^4$ としているため、シャッフル結果にある程度の誤差が生じていることも事実である。そのためここでは条件を緩和した上記の 11 通りに対して、 $S = 10^4$ のシミュレーションをさらに 10^3 回試行する ($S \times 10^3$)。この結果において $C = 1.000(2704)$ となった回数 $S'(0 \leq S' \leq 10^3)$ から、これら 11 通りの組み合わせの優越を比較する。

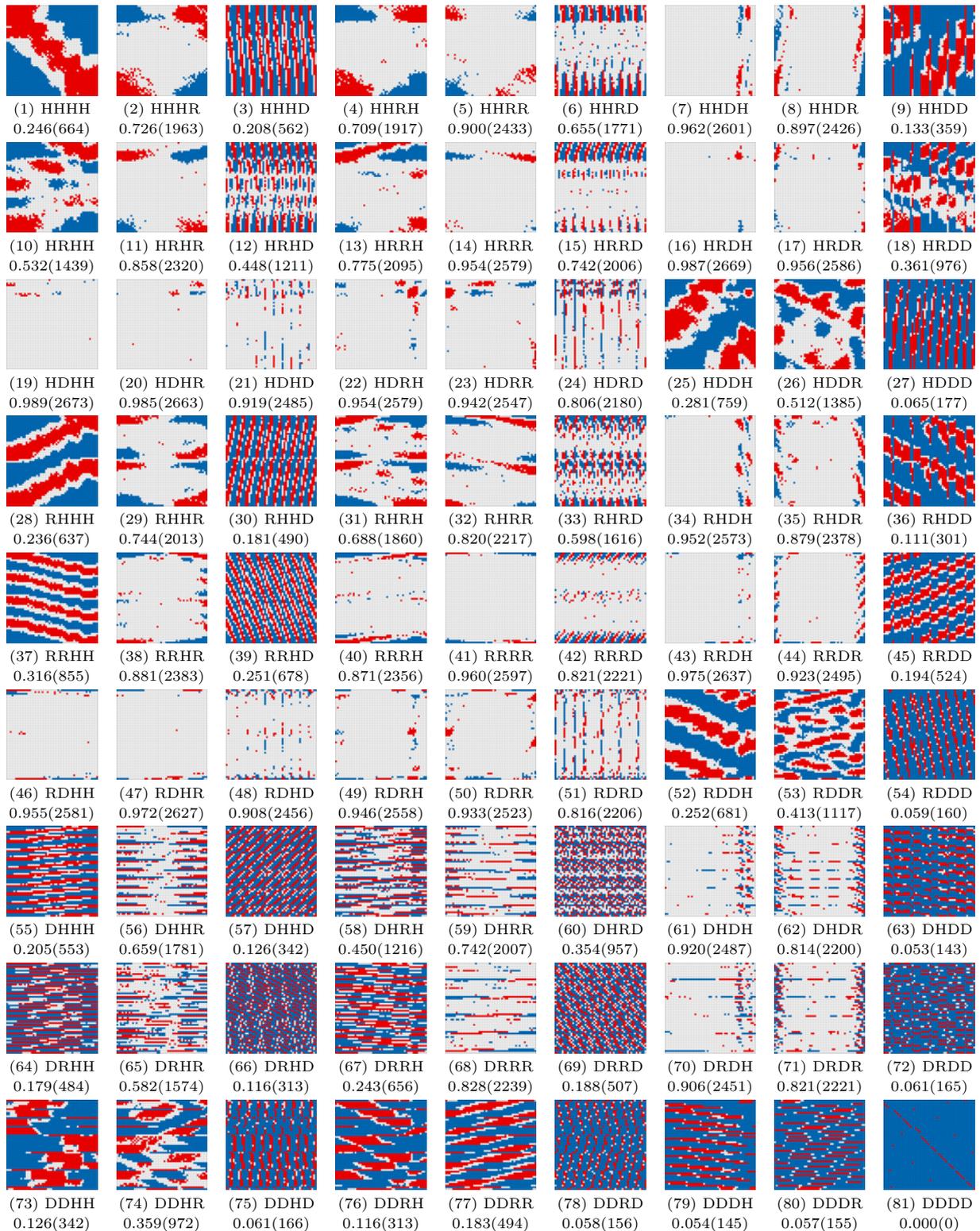
図 8: シャッフルの組み合わせ $A^4(m=4)$ における可視化と定量化 (81 通り)

表 1 に 11 通りの組み合わせにおける「シャッフルの所要時間 s (概算値)」および「 $S = 10^4$ のシミュレーションを 10^3 回試行した結果である w の内訳」を示す。この所要時間 s の算出には先行研究 [8] に従い 1 回の H を 3 秒, R を 7 秒, D を 30 秒と仮定した。まず表 1 から 10^3

回すべての試行において $C \geq 0.999$ かつ $w \geq 2700$ を満たしている組み合わせは (3)HRDHH, (7)HDHDH の 2 通りであることがわかる。このうち (3)HRDHH がもっとも良い結果となり, 65.4% ($S' = 654$) の頻度で $w = 2704$ となっている。ただし (1)HHDHH, (4)HRDHR,

表 1: A^5 における収束率 C の上位 11 通りに対する「シャッフルの所要時間 s [秒]」および「 $S = 10^4$ のシミュレーションをさらに 10^3 回試行した結果である w の内訳 [回]」

A^5	s	w					
		(S')	2704	2703	2702	2701	2700
(1)HHDHH	42	339	388	193	63	14	3
(2)HHDHR	46	15	102	173	231	190	289
(3)HRDHH	46	654	287	47	11	1	0
(4)HRDHR	50	301	346	249	77	22	5
(5)HDHHR	46	328	384	203	57	21	7
(6)HDHRR	50	131	295	286	150	88	50
(7)HDHDH	69	422	357	169	47	5	0
(8)HDHDR	73	45	202	299	261	125	68
(9)RHDHH	46	77	212	278	204	138	91
(10)RHDHR	50	0	6	15	44	95	840
(11)RDHDH	73	253	384	238	100	24	1

(5)HDHHR, (7)HDHDH についても 30~40% 程度で $w = 2704$ を満たしているため, (3)HRDHH には劣るが十分に実用可能な組み合わせであると考えられる。

次にこれらの所要時間 s を見た場合, 最短 42 秒, 最長 73 秒と 30 秒ほどの差が開いており, 「組み合わせに含まれる D の回数」が所要時間に大きく影響していることは明白である。しかし上位 11 通りの組み合わせにはすべて D が 1 回あるいは 2 回含まれているため, どのような組み合わせにしても最短 42 秒はかかる計算になる。これに対してもっとも良い結果となった (3)HRDHH は D を 1 回のみ含む 46 秒である。そのため (3)HRDHH におけるシャッフルの所要時間に関しては十分に許容範囲内であると考えられる。

一方さらに $C \geq 0.995$ まで条件を緩和させた場合, D を 1 度も含まない組み合わせについて HRRRR, HRRRR, RRRRR の 3 通りが存在することがわかった。これらの C については順に 0.996(2694), 0.995(2690), 0.995(2691) となっているため, ほとんど差はない状態である。しかし所要時間 s については順に 27, 31, 31 となっており, その差わずか 4 秒ではあるが HRRRR がもっとも短く, これらの中で唯一 30 秒以内に収まっていることがわかる。このようにデッキに完全なランダム性を求めなければ, D を含まなくともこれらの組み合わせで十分実用可能であると考えられる。

以上の結果から, A^5 の場合では $C = 1.000$ となる組み合わせが多数存在し, 特に (3)HRDHH の組み合わせがもっとも高い確率でデッキをランダムな状態にできることがわかった。またシャッフルの所要時間 s を考慮した場合, C は (3)HRDHH と比較して若干低下するが, HRRRR の組み合わせがもっとも効果的 (およそ (3)HRDHH の 4 割減) であることがわかった。なお A^5 において $C = 1.000$ となる条件を満たしたため, A^6 以降 ($m \geq 6$) の場合について本稿では言及しない。

6. おわりに

本研究では一般的なシャッフル手法であるヒンズー・シャッフル, リフル・シャッフル, ディール・シャッフル

について, シミュレーションによるシャッフル結果から 3 値ビットマップ画像への可視化と定量化を行い, 効果的なシャッフルの組み合わせについて考察した。

その結果, 単独のシャッフル試行においてはリフル・シャッフルがもっとも効果的であり, 最短 7 回のシャッフルでデッキをランダムな状態にできることがわかった。また複数のシャッフルを組み合わせた場合, 2 回の組み合わせでは $HR(C = 0.363)$, 3 回の組み合わせでは $C \geq 0.9$ を満たした HDH, RDH, 4 回の組み合わせでは $C \geq 0.98$ を満たした HRDH, HDHH, HDHR, 5 回の組み合わせでは収束率に重点を置くのであれば HRDHH ($C = 1.000, s = 46$), シャッフルの所要時間に重点を置くのであれば HRRRR ($C = 0.996, s = 27$) がもっとも効果的な組み合わせであることがわかった。なおこのように最短 5 回においてデッキを完全にランダムな状態 ($C = 1.000$) にできたことから, 単独のシャッフルよりも複数のシャッフルを組み合わせた方が効果的であると結論付けることができる。

以上, 本稿ではシミュレーションによるシャッフル結果を可視化し, シャッフルの性質およびシャッフルの効果的な組み合わせについて考察することができた。しかしこれはあくまでシミュレーションによる数値データ上の結果であり, 本研究でのデータと人間がデッキに対して“よく混ざっている”と感じる状態にはある程度の齟齬があると考えられる。そのため“人間のシャッフルに対する感覚”を考慮した効果的なシャッフルの組み合わせについて今後研究していく予定である。

参考文献

- [1] 高木重朗, 『カードマジック辞典 新装版』, 東京堂出版 (2016).
- [2] D,Aldous., P,Diaconis., *Shuffling cards and stopping times*, American Mathematical Monthly, Vol.93, No.5, pp.333-348(1986).
- [3] P,Diaconis., *Group representations in probability and statistics*, Institute of Mathematical Statistics Lecture Notes—Monograph Series, Vol.11(1988)
- [4] B,Mann., *How many times should you shuffle a deck of cards?*, Probability and Stochastics Series, pp.261-289(1995).
- [5] S,Assaf., P,Diaconis., K,Soundararajan., *Riffle shuffles of a deck with repeated cards*, The Annals of Probability(Institute of Mathematical Statistics), Vol.34, No.2, pp.804-819(2006).
- [6] P,Diaconis., *The cutoff phenomenon in finite Markov chains*, Proceedings of the National Academy of Sciences, Vol.93, No.4, pp.1659-1664(1996).
- [7] P,Diaconis., *The cutoff phenomenon for randomized riffle shuffles*, Wiley InterScience, pp.346-374(2007).
- [8] 野瀬彰大, 深川大路, “TCG におけるシャッフル手法に関する計算機実験を用いた考察”, 情報処理学会研究報告ゲーム情報学, Vol.2011-GI-25, No.4, pp.1-8(2011).
- [9] 熊谷隆, “マルコフ連鎖と混合時間—カード・シャッフルの数理—”, <http://www.kurims.kyoto-u.ac.jp/~kenkyubu/kokai-koza/H23-kumagai.pdf> [2017-5-1 参照].
- [10] E,Gilbert., *Theory of shuffling*, Technical memorandum, Bell Laboratories(1955).
- [11] J,Reeds., *Theory of shuffling*, Unpublished manuscript (1976).