G-03

トランプのシャッフルシミュレーションにおける無作為性の評価
Evaluating randomness of playing cards shuffles by simulations

西川 和希†　　今井 敏行‡
Kazuki Nishikawa　　Toshiyuki Imai

1. はじめに

コンピュータ上でトランプ 54 枚を対象としたカードの束においてシャッフルを行い，シャッフルが無作為化の点で効果的であるかを評価する．本研究におけるトランプの無作為化とは，次に引くカードを予測できない状態にすることだとする．

2. シャッフルの再現方法と乱数による並び替え

メジャーなシャッフルである“ヒンズー”，“ディール”，“ファロー”をプログラムとして再現する．ここで，54 枚のカードを 0 から 53 までの整数とみなし，0、1、2、…51、52、53 と順に並んだ状態を初期状態としている．これら整数の並び替えによってシャッフルを再現する．また，本研究では乱数によるカードの並び替えを実装し，シャッフルにより無作為化が行われているか判断するための比較対象とする．

2.1 各シャッフルの特徴とプログラムによる再現

ヒンズーは山札を左右任意の手で支え，もう片方の手で任意の枚数を抜き出す．この操作を山札がなくなるまで繰り返す方法であり，最も一般的なシャッフル方法である．

プログラムによる再現において，先行研究[1]を参考にし，抜き出す枚数を平均 10，標準偏差 5/3 の正規分布に従うものとした．

ディールは山札の上から 1 枚ずつ並べていき，小さな束を複数作り最終的にこの束を 1 つにまとめる方法である．初期配置において隣り合ったカードが散らばる特徴があるが，並び替えが規則的であるため，シャッフル後の配置は予測可能である．

プログラムによる再現において，山札の上から順番にカードを取り出していき，6 枚ずつ 9 つに分けた束をまとめる方法で実装した．

ファローは山札をおおよそ半分に分け，2 つの山札の片方をもう片方に嚙み合わせ，カードが互い違いに入れ込む方法である．しかし，カードを嚙合わせる操作は難度が高く，慣れないユーザーにとってはカードを落としてしまうなど，失敗しやすいシャッフルである．

プログラムによる再現において，先行研究[2]を参考にした．2 つの山札を互いに入れ込み 1 つにする操作において，入れ込むカードがどちらの山札から選ばれるかを，2 項分布を元に決定している．

2.2 乱数による並び替え

シャッフルによる無作為化を評価するにあたり，乱数による並び替えを実装する．使用する乱数は，メルセンヌツイスタである．厳密には疑似乱数であるが，本研究においては乱数として扱って問題ないと考える．

†和歌山大学大学院，Wakayama Graduate School
‡ 和歌山大学，Wakayama University

3. 無作為性を評価する 2 つの評価方法

無作為であることの確認は難しい．完全な乱数ならばこうなるはずという基準はいくつも設定でき，その基準を満たさなければ乱数ではないとは言えるものの，基準を満たせば乱数であるとは言えないからである．

2.1 で説明したシャッフルを試行した結果，トランプの無作為化がどれだけ進んでいるか“前後の差を比較”“カードの移動配置を記録”の 2 つにより評価を行う．

3.1.1 前後の差を比較

シャッフルと乱数による並び替えにおいて，並び替え後における並びの前後のカードの差を比較する．カードに見立てた数字は 0 から 53 までの整数であり，とりうる値の最大値は$53 - 0 = 53$、最小値は$0 - 53 = -53$である．ただし，同じ数字は存在しないため，前後の差が 0 になることはない．

3.1.2 基準の設定と許容誤差の決定

乱数による並び替えを 1 万回試行し，-53から 53 までの値の出現回数を記録した．X 軸を前後の差，Y 軸を出現回数としてグラフで表現した結果，図 1 のように$y = 9990 - 185x$を左右に反転させたグラフと，ほぼ一致することが分かった．

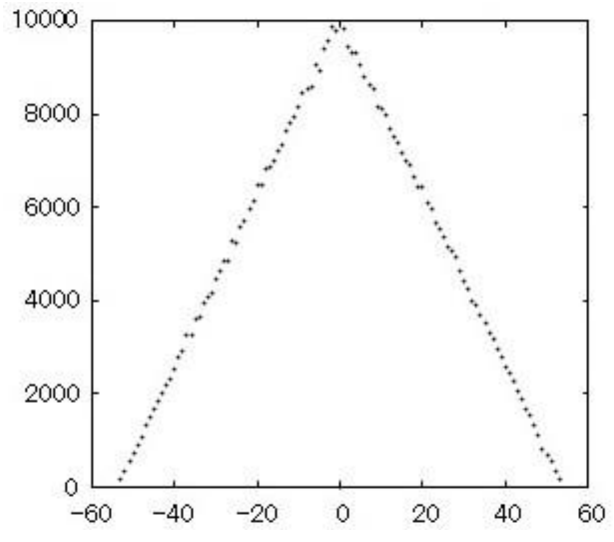

図 1.　乱数並び替え 1 万回

これにより，乱数による並び替えを無作為化が行われているかの基準にする本研究において“十分に無作為化が行われた並び替え 1 万回の試行において，前後の差-53から 53 までの出現回数を記録すると，$y = 9990 - 185x$に従う”と言える．

しかし，1 万回の試行において，$y = 9990 - 185x$から結果が外れる値を許容可能な差として“許容誤差”と定義した．許容誤差の算出方法は以下の通りである．

1. 疑似乱数による並び替えを 1 万回試行し，前後の差-53 から 53 までの出現回数を記録する
2. 1 を千回繰り返し，-53 から 53 までそれぞれ千個の出現回数を昇順に並び替える
3. 下位 50 個を外れ値とし，950 番目の出現回数をそれぞれの許容誤差として設定する

各シャッフルにおいてもそれぞれ 1 万回試行し，前後の差を計算する．$y = 9990 - 185x$を反転させたグラフと重ね合わせ，その差が許容誤差の範囲に収まっているかを計算し，どれほど無作為化ができているかを評価する．

3.1.3　無作為化率の算出方法

各シャッフル 1 万回の試行において，前後の差－53 から 53 の出現回数がどれほど許容誤差を満たしているかを計算する．この許容誤差を満たしているほど無作為化が出来ていると評価する．その値を“無作為化率”と定義し，百分率で表現する．

3.1.4 より各シャッフルの試行による無作為化率について説明する．無作為化率をグラフで表したデータでは，X 軸はシャッフルの試行回数，Y 軸は無作為化率を表している．

ここでは具体例をあげて説明する．3.1.4 のデータにおける“ヒンズー25 回の試行における無作為化率”とは以下の手順により算出したデータである．

1. 初期配置のトランプ 54 枚に対しファローを 25 回試行する
2. 0 を除く－53 から 53 までの前後の差が出現した回数をそれぞれ記録する
3. トランプの配置を初期配置に戻す
4. 1 から 3 を 1 万回繰り返し，ファロー25 回の試行 1 万回における前後の差合計出現回数を記録する
5. 前後の差それぞれが出現回数の許容誤差を満たしているかを判定する
6. 許容誤差を満たす前後の差の個数を，0 を除く－53 から 53 までの前後の差の合計数 106 で割り，100 を乗じることで無作為化率を算出する

3.1.4　各シャッフルの無作為化率

図 2 がヒンズーにおける試行回数と，無作為化率を記録したグラフである．

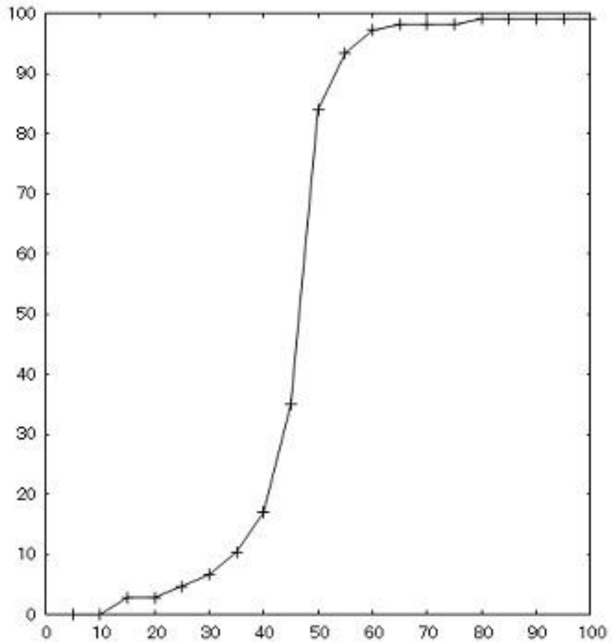

図 2.　ヒンズーによる無作為化率

50 回の試行で 84.0%，55 回の試行で 93.4%というように，無作為化に近づけるために必要な試行回数が多いシャッフルであることが，このグラフから分かる．

図 3 がディールにおける試行回数と，無作為化率を記録したグラフである．

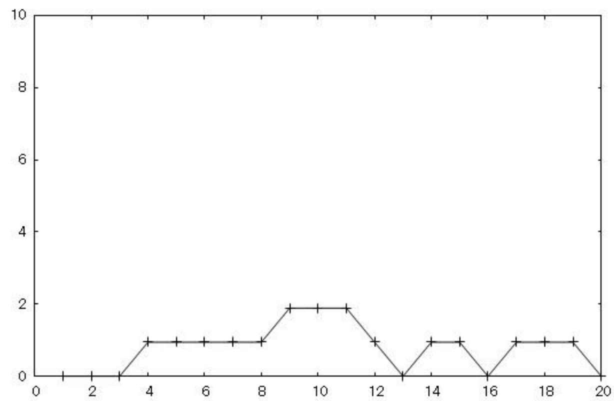

図 3.　ディールによる無作為化率

2.1 で説明したようにディールは規則的な並び替えであるため，試行回数を増やすほど無作為化率が増大するシャッフルではない．このグラフでは20回までの試行を記録したが，より試行回数を増やしたとしても現実的に実行可能な回数では，無作為化率が飛躍的に増大することはなく，ディールだけでは無作為化は不可能であると言うことが出来る．

図 4 がファローにおける試行回数と，無作為化率を記録したグラフである．

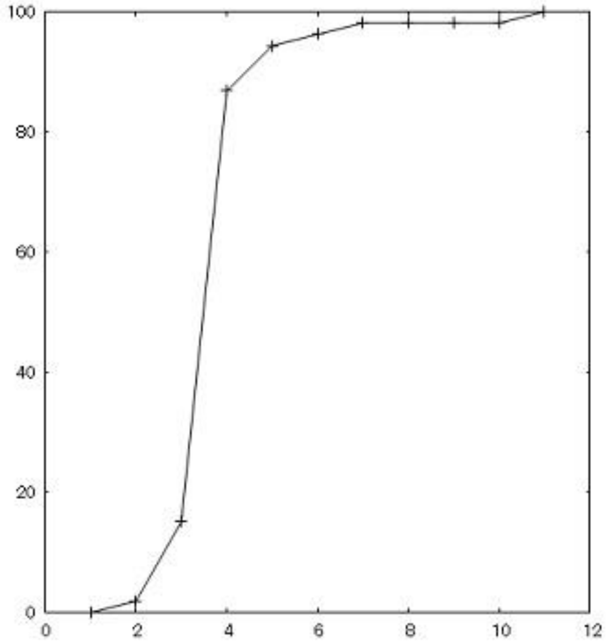

図 4.　ファローによる無作為化率

ファローは少ない試行回数で無作為化に近づけることが可能であり，4 回の試行で 86.8%，5 回の試行回数で 94.3% の無作為化率となっている．この結果から，ファローは 3 種類のシャッフルのうち，最も少ない試行回数で無作為化に近づけることが可能であると言える．

3.2.1　カードの移動配置を記録

0 から 53 までのカードが，シャッフルおよび乱数による並び替えによる操作の後，どの位置に移動したかを記録する．カードの移動配置は 0 から 53 までの 54 通りである．例えば，山札の一番上のカードである 0 が，並び替え試行後も一番上に存在する場合“0 が 0 に移動した”と表現する．同様に山札の一番下に移動した場合は“0 が 53 に移動した”このように表現する．

3.2.2　許容誤差の設定

乱数による並び替えを 9990 回試行し，54 枚のカードが並び替え後においてどの位置に移動したかを記録する．例えば，山札の 1 番上のカード 0 に注目した場合，考えられる移動場所は 0 から 53 の 54 通りである．移動場所を選ぶ確率が全て等しい場合，9990 回の試行における 1 つの場所当たりの移動回数は$9990 \div 54 = 185$となる．これは山札の一番上 0 だけではなく，残りの 1 から 53 も同様であり，一度の移動場所の記録において，$54 \times 54 = 2916$の移動回数が算出される．

しかし，9990 回程度の試行回数であれば，乱数による並び替えであったとしても，185 から外れた値になる可能性が高い．そこで，前後の差を比較する評価と同様に，許容可能な差を“許容誤差”と定義した．許容誤差の算出方法は以下の通りである．

1. 乱数による並び替えを 9990 回試行し，2916 の算出結果の中から，マイナスを含む 185 から最も離れた値を 1 つ取り出し，その値と 185 との差を記録する
2. 1 を 1 万回繰り返し，185 から最も離れた値と 185 との差 1 万個を昇順に並び替える
3. 下位 500 個を外れ値とし，9500 番目の値と 185 との差を許容誤差として算出する

上記の方法で許容誤差を計算した結果，45 となった．つまり，9990 回の試行において$185 + 45 = 230$以下かつ，$185 - 45 = 140$以上の出現回数であれば，許容誤差に収まると判断し，2916 の移動回数においてこの条件を満たすほど無作為化に近づいていると判断する．

各シャッフルをそれぞれ 9990 回試行し，2916 の移動回数の中で，許容誤差を満たす数をグラフとして表現する．

3.2.3　各シャッフルにおける許容誤差

図 5 はヒンズーにおける許容誤差を満たす数をグラフで表した結果である．なお，グラフの X 軸は山札の上から X 番目に配置されていたカードを表し，目盛は 0 から 53 までの整数である．Y 軸は X 番目のカードが 54 通りの移動配置において，どれだけ許容誤差を満たしているかを表し，目盛は 0 から 54 までの整数である．また，グラフの下の数字はヒンズーの回数を表している．

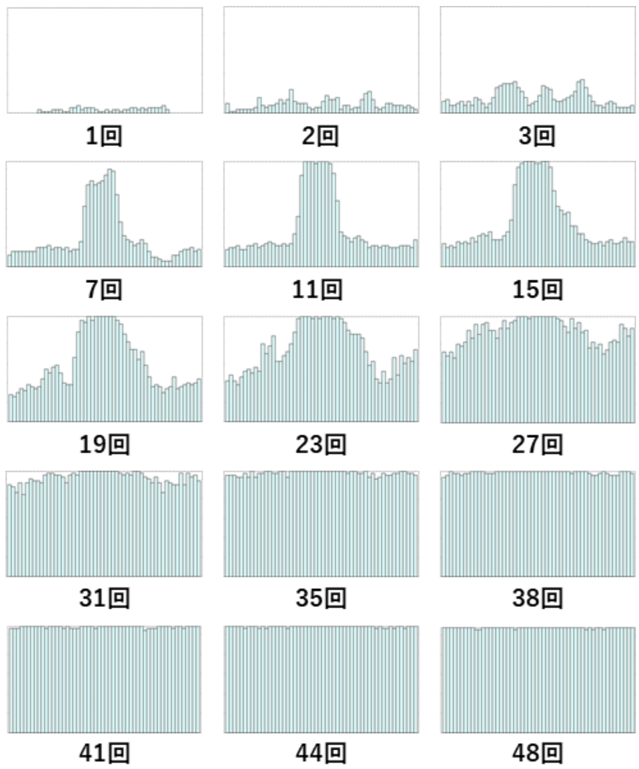

図 5.　ヒンズーによる許容誤差を満たす数

ヒンズー48 回までの結果である．この結果からヒンズーにおいて，山札の中心は少ない回数で許容誤差を満たしている，つまり無作為化が進んでいることが分かる．一方で中心から外れると許容誤差を満たす数は，回数を増やすことで全体として増えていることが分かる．35 回から X 番目

の配置全体として許容誤差を満たす数が54に近づいていることが分かる．特に41回以降はほとんどの配置において，許容誤差を満たす数が54に近い結果となり，全体として無作為化が進んでいることが分かる．

図 6 はディールにおける許容誤差を満たす数をグラフで表した結果である．グラフの読み取り方はヒンズーと同様であり，下の数字はディールの回数を表している．

図6.　ディールによる許容誤差を満たす数

ディール30回までの結果である．前後の差の比較でも述べたように，ディールは規則的な並び替えであり，このシャッフル掛けで無作為化を行うことは不可能である．上の結果ではそれが顕著に表れており，30 回までの試行においてX番目の配置における許容誤差を満たす数は 0 という結果となった．

図 7 はファローにおける許容誤差を満たす数をグラフで表した結果である．グラフの読み取り方はヒンズーと同様であり，下の数字はファローの回数を表している．

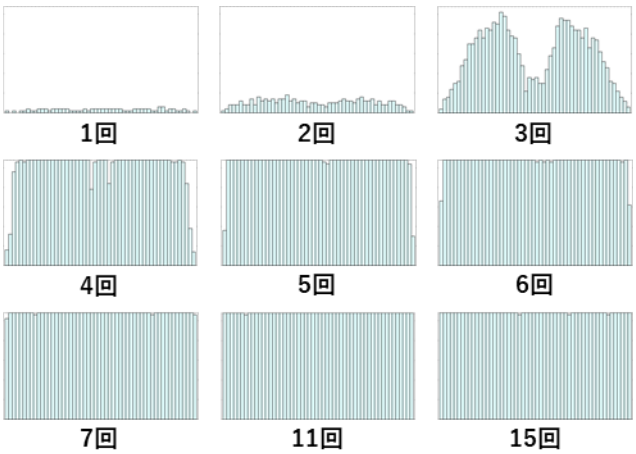

図 7.　ファローによる許容誤差を満たす数

ファロー15 回までの結果である．3 回のグラフに注目すると，2 項分布を 2 つ組み合わせたような形になっている．これは 2.1 で説明したように，ファローをプログラムで再現するにあたり，2 項分布を取り入れたためだと推測できる．また，山札の中心に近い配置から 2 つに分かれている理由は，ファローの性質である山札をおおよそ半分に分けてから入れ込む操作を再現したためだと推測できる．

4 回の段階で全ての配置において許容誤差を満たす数が 54 に近いように見えるが，山札の一番上と一番下に近い配置は許容誤差を満たす数が少ないことが分かる．この傾向は 6 回まで見られるが，7 回目以降はほとんど全ての配置において許容誤差を満たす数は 54 となっている．

この結果から，6 回では山札の一番上と一番下の配置において許容誤差を満たす数が少なくなるが，7 回以上ファローを繰り返すことにより，カードの移動配置を記録する評価方法において，無作為化できることが分かった．

4. 各シャッフルの無作為化の点で見た評価

ここまで各シャッフルによるカードの無作為化を，2 つの評価方法により評価した．その結果，各評価によって無作為化に近づくまでの回数には差が見られたが，最も少ない試行回数で無作為化が成された状態に近づけることが可能なシャッフルはファローであり，最も一般的なシャッフルであるヒンズーでは，ファローと比べ回数を重ねなければ不可能であることが分かった．また，ディール単独の試行ではどちらの評価方法においても，無作為化を行うことは不可能であるという結論となった．

このことから，最も手間をかけることなくカードを無作為化したい場合，有効なシャッフルはファローであることが分かった．ファローだけでの無作為化を試みるならば，数十回繰り返す必要があり非常に手間である．

参考文献

[1] 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（GI）Vol.2011-GI-25, No.4, pp. 1-8, 情報処理学会, 2011/3/5

[2] 井手　広康, ”不完全情報ゲームにおける攪拌に関する研究”, 平成 30 年度愛知県立大学大学院情報科学研究科修士論文

[3] 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-”, 平成 23 年度（第 33 回）数学入門公開講座テキスト, pp. 1-3, 京都大学数理解析研究所, 2011/8/4