

麻雀初心者に向けた役作成のヒント例示システムの構築

早川 大貴^{1,a)} 上野 未貴² 井佐原 均²

概要: 本研究では麻雀初心者に対し、現在の手牌から作成可能な、点数最大の手と手数最小の手の2つのヒントを例示するシステムを構築し、有用性をアンケートにより評価した。また、眼鏡型ウェアラブルデバイスである JINS MEME を用いて、初心者が麻雀プレイ中に必要とする情報を検証した。

Learning Support System to Manage the Role of Mahjong for Beginners

DAIKI HAYAKAWA^{1,a)} MIKI UENO² HITOSHI ISAHARA²

1. はじめに

人工知能分野で不完全情報ゲームである麻雀が注目されている。ポーカーや麻雀などの不完全情報ゲームは将棋や囲碁のような完全情報ゲームとは異なり、自分の手、場に出ているカードや牌などの公開されている情報と、相手の手、次に来るカードや牌などの公開されていない情報の両方を用いながら、プレイヤー同士の駆け引きや様々な状況における最良の一手を考えプレイするゲームである。そのため、計算機資源の豊富な昨今でも、考慮すべき状況が極めて多いことから、数理的なモデル化に課題が残されている。

不完全情報ゲームはルールや役が多いため、実際にゲームをするためには、多くの知識を必要とする。また、各時点の状況や相手の手を考慮してプレイすることが必要になる。

そのため、初心者と上級者では、使用する情報に大きな差異がある。従って、初心者と上級者の情報の取捨選択の差異を調べて、麻雀の牌選択の数理モデル化に必要な情報を精査することを考える。

そこで本研究では、不完全情報ゲームである麻雀に着目

し、上級者の打ち筋を学習し、初心者が対局中に求める手役のヒントを複数例示することにより、初心者の思考過程を支援するシステムの構築を目指す。

本稿では、第一段階として、現在の自分の手牌から点数最大となるヒントと手役完成までの打牌回数が最小となるヒントの2つを例示するシステムを構築し、ユーザ実験により評価した。得られた結果から、初心者が求めるヒントについて考察をした。

2. 関連研究

麻雀に関する研究について以下に記述していく。田中ら [1] は麻雀初心者の教育に適した助言を出力できるモデルの作成を目的とし、麻雀において重要な戦略である”早く上がる” ”高得点狙い” ”振り込み回避”といった評価基準を元に行動を評価する3つのモデルを作成、これらを用いて上級者のゲーム記録を戦略別に分類し、戦略を付与した牌譜を教師として局面の情報から取るべき戦略を出力する決定木の学習をした。

築地ら [2] はディープラーニングの技術を用いて麻雀の評価関数を学習することを試みている。純粋な多層ニューラルネットとして学習した場合、ディープラーニングの技術であるオートエンコーダとドロップアウトを用いてディープラーニングの有用性を評価した。

水上ら [3] は1人麻雀と4人麻雀の差を解析し、その差を埋めることで1人麻雀コンピュータプレイヤーの4人麻雀

¹ 豊橋技術科学大学大学院
Toyohashi University of Technology, Toyohashi, Aichi 441-8580, Japan

² 情報メディア基盤センター
Information and Media Center, Toyohashi, Aichi 441-8580, Japan

a) hayakawa@lang.cs.tut.ac.jp

への適用を図っている。その際、1人麻雀にはない“降り”の要素を取り入れるために、降り売るべき局面とそうでない局面を分類している。

とつげき東北ら [4] は“捨て牌に占める 3~7 の数牌の比率”や“シャンポン待ちが可能な字牌の数”などの上級者プレイヤーの知見に基づき設定した 7 つの判断要素を元に、リーチをかけているプレイヤーが字牌待ちかどうかを判別するフィルタを作成し、プロを含む被験者と正答率を比較しており、初心者と上級者の打ち筋の違いを確認している。

本研究では、麻雀のプレイ中にユーザの段階に応じたヒントの例示ができるように、初心者が求めるヒントを調査し、要求を満たすシステムを構築する。

3. 提案手法

3.1 予備実験

本研究ではまず予備実験として、麻雀初心者が麻雀プレイ中に望むヒントがどのようなものかを調査した。その結果以下のような回答を得た。

- 打牌選択時
 - 現在の手牌から考えられる役をいくつか列挙
 - 手牌に安全牌が存在しない状況で他プレイヤーがリーチをかけている時に切れそうな牌をいくつか列挙
 - 待ち牌がより多くなるような打牌選択の出力
- 打牌後
 - 学習の一環として他の打牌案をいくつか列挙、どういう手になるか解説

本研究ではこの中から、打牌選択時に現在の手牌から考えられる役のヒントを複数例示するシステムの構築を目指した。

3.2 前提条件

今回、本研究で構築するシステムに以下のルールを採用した。

- 今回のシステムで考慮する項目
 - リーチしたかどうか
 - ツモ牌
 - 鳴きの形、鳴きの回数
- 今回のシステムで考慮しない項目
 - 槓子 (暗槓, 明槓含む)
 - ドラ

本研究の提案手法については次節で説明する。

3.3 提案手法

本研究では手役のヒントを複数例示するために、以下の 3 つのアルゴリズムを構築した。

- (1) 上がり判定アルゴリズム
- (2) 役判定アルゴリズム
- (3) 役作成ヒント例示アルゴリズム

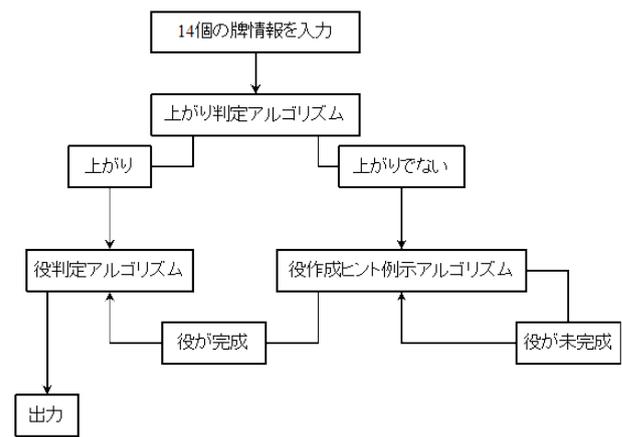


図 1 本研究で用いるアルゴリズムの流れ

3 つのアルゴリズムを組み合わせることにより、初心者に対し有用なヒントを例示できると考える。図 1 に 3 つのアルゴリズムの流れを示す。まず 14 個の牌情報を入力し、上がり判定アルゴリズムで手牌が上がりかどうかを調査する。上がりの場合は役判定アルゴリズムで手牌の役を判定し、上がりでない場合は役作成ヒント例示アルゴリズムを用いて現在の手牌から作成可能な役を役判定アルゴリズムで判定し例示する。なお、それぞれのアルゴリズムについての詳しい説明は次章に記述する。

4. アルゴリズム

この章では、本研究で構築した上がり判定アルゴリズムと役作成ヒント例示アルゴリズム、役判定アルゴリズムについて記述していく。

4.1 上がり判定アルゴリズム

上がり判定アルゴリズムでは、任意で入力された 14 の牌情報から上がりかどうかを判定する。入力は文字列で表現された 14 枚の牌情報を使用する。以下にその例を記述する。

m1 m5 m7 p1 p6 p8 p9 s2 s3 s4 s4 s5 z2 z3 z4

ここで、本研究における入力データのルールを以下に示す。

- (1) 萬子, 筒子, 索子, 字牌をそれぞれ [m, p, s, z] で表す。
- (2) 字牌は [東, 南, 西, 北, 白, 發, 中] をそれぞれ 1~7 で表す。
- (3) 各牌の情報は次の形で表す。
 - 萬子: m1~m9
 - 筒子: p1~p9
 - 索子: s1~s9
 - 字牌: z1~z7

また、鳴きで作成した順子及び刻子や自摸の情報もここで入力することができ、それぞれ鳴きの場合は順子及び刻子を () で囲み、自摸の情報は数字の直後に' を挿入するこ

とで判断される。以下にその例を示す。以下の例の場合、萬子の2~4が鳴きによって作成され、筒子の8が自摸によって入手した牌となる。

m1 m1 (m2 m3 m4) p5 p6 p8' s1 s1 z4 z4 z4

図2に上がり判定アルゴリズムのフローチャートを示す。初めに入力された14個の牌情報を種別ごとに辞書型リスト kind に格納する。kind にはキー値としてすべての牌に振り分けた番号が格納されており、各牌番号の値が手牌に存在する牌の数を表す。ここで、以下に各牌に振り分けた番号を示す。

- 萬子：1~9
- 筒子：11~19
- 索子：21~29
- 字牌：31~37

次に上がり判定で判断できない特殊役かどうかを判定する。ここで特殊役とは国士無双と七対子を示す。特殊役があった場合この時点で上がりと判定する。

以下に特殊役以外の上がり判定について記述する。

- (1) 手牌に含まれている牌番号リストの先頭の要素を i とする。
- (2) kind[i] の値が2以上すなわち手牌に番号 i の牌が2枚以上の場合、 i を雀頭リストに格納し、kind[i] = kind[i] - 2 とする。
- (3) 順子と刻子のリスト、を獲得するために、すべての牌番号 j ($j = 1 \sim 37$) に対し、以下の操作をする。 $j = 1$ とする。
 - (a) $j \leq 30$ かつ、kind[j] ≤ 1, kind[j+1] ≤ 1, kind[j+2] ≤ 1 の場合、 $j, j+1, j+2$ を順子として順子リストに格納し、kind[j], kind[j+1], kind[j+2] から1を引く。
 - (b) 手配に番号 j の牌が3枚以上すなわち kind[j] ≤ 3 の場合、 j を刻子としてリストに格納し、kind[j] = kind[j] - 3 とする。
 - (c) $j < 37$ なら、 $j = j + 1$ とし、3に戻る。 $j = 37$ の場合、kind に存在するすべてのキーに対する値が0の場合、上りとして判定。そうでない場合、手牌に含まれる牌番号リストの次の要素を i に格納し、2に戻る。手牌に含まれる牌番号リストの次の値がないならば終了する。上り判定が一度もなかった場合には、上りでないと判定する。

なお、一盃口もしくは二盃口の場合も考慮するため、(a) の操作で kind[j] と連続する2つの牌番号の値が2個以上の場合、2つ順子があると判断し、それぞれ2ずつ引いている。また、1つの手牌で複数の上がり方が存在する場合を考慮し、上がりと判定された場合、その時の雀頭、順子、刻子をリストに格納し、任意の数の結果を出力できるように設計した。

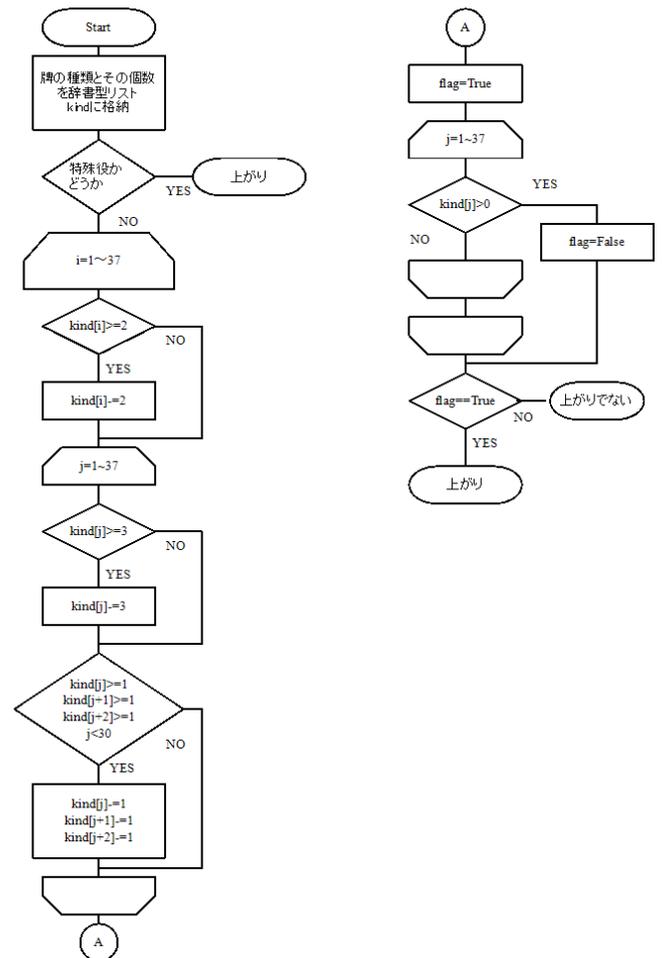


図2 上がり判定アルゴリズムのフローチャート

4.2 役判定アルゴリズム

次に役判定アルゴリズムについて以下に記述していく。ここで、役判定アルゴリズムでは先に役満を判定し、役満が出た時点で終了するように設計した。ただし、役満が複数出現する場合(例：大三元、字一色)は、役満のみでリストが構成されるように設計をした。

- (1) 入力には雀頭と順子及び刻子のリストとする。
- (2) 入力されたリストから役満かどうかを判定する。役満の場合はここで操作終了。
- (3) 役満以外の場合、入力された情報に合致した役を判定する。判定された役はリストに格納し、その役に対する翻数を足す。
- (4) 入力された情報から導出された役のリストと翻数を出力する。

入力された情報から役満以外と判定された場合、それ以外の役かどうかの判定に移行する。1つの手牌に対し役満以外の役は複数存在する場合があるため、1つの役が出た場合それのみを返すのではなく、出現した役をリストに格納してそれを返すようにした。また、役に対する翻数^{*1}の合計値も返り値とした。前述した1つの手牌で複数ある場合

*1 麻雀の点数計算を行う際に必要となる数値

に、点数が一番多く獲得できるような上がり方を出力する。

4.3 役作成ヒント例示アルゴリズム

最後に役作成ヒント例示アルゴリズムについて記述する。このアルゴリズムは、上がり判定アルゴリズムにおいて上がりではないと判定された場合に、現在の手牌の状況から作成可能な役と必要牌、その場合の理想手牌を出力するアルゴリズムである。

役作成ヒント例示アルゴリズムの手順について以下に記述していく。

- (1) 現在の手牌から作成可能な雀頭、順子、刻子及び余剰牌の情報を入力とする。なお、入力する情報はすべてリスト形式である。
- (2) 余剰牌のリスト L から i 番目の牌情報を削除する。
- (3) L 内に存在する残りの余剰牌に対し、雀頭、順子及び刻子になりえる牌を必要牌 j として選択する。
- (4) 必要牌 j を用いて雀頭、順子及び刻子になりえる牌を L 内から削除し、削除した牌と必要牌 j でなりえる雀頭、順子及び刻子の情報、必要牌 j の情報を保存する。
- (5) L 内に情報が含まれていない場合、入力された情報と(4)で新たに保存された情報を入力として役判定アルゴリズムを操作する。
- (6) L 内に情報が含まれている場合は、(1)に戻る。この時、試行回数と必要牌の情報を記録する。

ここで、(4)において雀頭になりえる牌を選択する場合は、上がり判定アルゴリズムにおいて雀頭が作成されなかった場合にのみ適用されるように設計した。

また、このアルゴリズムでは余剰牌の数から導出できる全てのヒントを例示できるようになっているため、その中から点数が最大となるようなヒントと、試行回数が最小となるようなヒントの2種類を出力し、初心者を選択肢を限定し、アルゴリズムの有用性を調査した。

5. 実験

本研究の実験として、前章で記述したアルゴリズムの動作を確認し、役作成ヒント例示アルゴリズムの有用性を初心者に対してアンケートをし、評価した。また、ウェアラブルデバイスである JINS MEME を用いて初心者が麻雀を行う上でどのような情報を要求するのかを検証した。

5.1 実験 1: アルゴリズム動作の確認

5.1.1 上がり判定アルゴリズム

上がり判定アルゴリズムの動作を確認するため、数回例を入力した。表 1 に動作結果を示す。

入力された 14 牌の情報をもとに正常に結果が出ていることを確認した。また、国士無双や七対子の特殊役も上がり判定されていることも確認した。

表 1 上がり判定アルゴリズムの動作結果

入力文	出力結果
m2 m3 m4 s2 s3 s4 s5 s6 s7 s8 p1 p1 p1 z2 z2	上がり
s2 s2 s3 p1 p2 p3 p4 p5 p6 m3 m3 m3 z7 z7	上がりでない
m1 m9 p1 p9 s1 s9 z1 z2 z3 z4 z5 z6 z7 p1	上がり
m1 m1 m4 m4 p2 p2 p3 p3 s9 s9 z2 z2 z4 z4	上がり

表 2 役判定アルゴリズムの動作結果

入力文	出力結果
m2 m3 m4 s2 s3 s4 s5 s6 s7 p1 p1 p1 z2 z2	役なし
s1 s1 s1 p2 p2 p2 p5 p5 p5 m3 m3 m3 z5 z5	四暗刻
s1 s1 s1 p2 p2 p2 p5 p5 p5 (m3 m3 m3) z5 z5	対々和, 三暗刻
m1 m2 m3 m4 m4 m4 z5 z5 z5 z6 z6 z7 z7	混一色, 小三元 役牌(白), 役牌(發)
m1 m1 m4 m4 p2 p2 p3 p3 s9 s9 z2 z2 z4 z4	七対子
m1 m1 m2 m2 m3 m3 p1 p1 p2 p2 p3 p3 s4 s4	二盃口

5.1.2 役判定アルゴリズム

次に役判定アルゴリズムの動作を確認するため、上がり判定アルゴリズムにおいて上がり判定された牌情報を用いることとし、こちらも数回例を入力した。表 2 に動作結果を示す。

結果を確認したところ、役が見つからないものに対しては役がない状態で出力され、役が存在する場合は対応する役のリストが出力された。また、役満が含まれている場合は役満のみ、役満以外の役の場合ではそれ以外の役がリストに含まれていることを確認した(表 2, 2~3 行目)。七対子と二盃口の区別ができていたことも確認した(表 2, 5~6 行目)。

しかし、表 3 のような入力をした場合、本来出るのははずの出力とは異なった結果が出るのが判明した。七対子と複合して出てくるはずの役が七対子のみで出力されてしまっていた。この理由として、七対子は上がり判定アルゴリズムにおいて判断されており、七対子が出現した時点で結果が出力される仕様になっていたためである。解決策として、七対子と複合する役には限りがあるため、七対子が出現した場合、七対子に複合される役の判定を行う必要がある。

5.1.3 役作成ヒント例示アルゴリズム

最後に役作成ヒント例示アルゴリズムの動作確認をするため、上がり判定アルゴリズムにおいて上がりでない判定される牌情報を入力として結果を確認した。表 4 に結果を示す。

上がりでない手牌を入力した場合、点数最大となるような手牌と手数が最小になる手牌の2種類が出力されたことを確認した。ここで、今回作成したアルゴリズムは、入力した手牌から作成可能な役をすべて導出するアルゴリズムであり、すべての牌を入れ替えないようなものになっているので、その挙動についても確認した。

表 3 役判定アルゴリズムで確認した誤判定

入力文	正解	出力
m1 m1 m3 m3 m4 m4 m6 m6 m7 m7 z1 z1 z2 z2	七対子, 混一色	七対子

表 4 役作成ヒント例示アルゴリズムの動作結果

入力文	出力結果 (点数最大)	出力結果 (手数最小)
s2 s2 s3 p1 p2 p3 p4 p5 p6 m3 m3 m3 z7 z7	1p 2p 3p 4p 5p 6p 3m 3m 3m 中中中 2s 2s 役=[役牌(中)] 必要牌=中 必要手数=1	1p 2p 3p 4p 5p 6p 3m 3m 3m 2s 2s 中中 役=[] 必要牌=2s 必要手数=1
m1 m1 m2 m3 m5 m6 z5 z5 z6 z6 z7 z7	1m 2m 3m 白白白發發發中中中 1m 1m 役=[大三元] 必要牌=中,1m 必要手数=2	1m 2m 3m 4m 5m 6m 白白白發發發中中 役=[混一色 小三元 役牌(白) 役牌(發)] 必要牌=4m 必要手数=1
p1 p3 p5 p5 p7 p8 p9 s3 s6 s6 s7 z2 z5 z5	7p 8p 9p 1p 2p 3p 5p 5p 5p 白白白南南 役=[混一色 役牌(白)] 必要牌=2p,5p, 白, 南 必要手数=4	7p 8p 9p 1p 2p 3p 5p 5p 5p 6s 6s 6s 白白 役=[] 必要牌=2p,5p,6s 必要手数=3

5.2 実験 2:初心者アンケート

次に、今回作成したアルゴリズムを実際に麻雀初心者で使用していただき、アンケートを実施し評価した。

5.2.1 実験設定

本研究では 1 人麻雀を用いて実験する。1 人麻雀は 4 人麻雀における不完全要素を取り除き、配牌とツモのみでプレイする麻雀である。今回はアルゴリズムの有用性を確かめるため、1 人麻雀における和了率、テンパイ率、平均得点を確認した。

対象は麻雀初心者 20 代男性 1 名で、アルゴリズムを使用しない時と使用する時でそれぞれ 10 局プレイしていただいた。また、アルゴリズム使用後にアンケートを行い、本研究のアルゴリズムの有用性を確かめた。以下にアンケートの内容を示す。

- (1) 良かった点
- (2) 悪かった点
- (3) 追加してほしい機能
- (4) 点数最大と手数最小の参考度

なお、アルゴリズムの使用回数は 1 局に対し無制限とし、アルゴリズムの性質上シャンテン数が 2 以下の場合にのみ使用できるものとした。

5.2.2 実験結果

表 5 に実験結果を示す。アルゴリズムを使用した場合、使用しなかった場合に対して和了率が 10% 向上していたが、テンパイ率と平均得点が下がってしまっていた。この理由として考えられるのは、ヒントを提示することによって、現在の手牌に一番近い役を視覚的に確認することができたが、点数最大の手と手数最小の手の 2 種類で状況を判断したときに、冗長な選択をしてしまいテンパイにできなかったものと考えられる。

また、実験後に実施したアンケートでは、以下のような

表 5 実験結果

	和了率	テンパイ率	平均得点
アルゴリズムなし	10%	50%	3900 点
アルゴリズムあり	20%	40%	2100 点

評価を得た。

- (1) 良かった点 - 目指す役が見つからない時に、自分が想像していなかった役が発見できた
- (2) 悪かった点 - 結果の表示方法が種類ごとにソートされておらず少し見づらい
- (3) 欲しい機能 - ドラの考慮、不要牌の表示、同局で利用する際の簡略機能
- (4) 参考度 - 必要手数の違いが 1 程度であれば点数最大、それ以外では手数最小

提案手法により、初心者が麻雀をプレイ中に新たな役を発見する助けができることを確認した。また、点数最大の手と手数最小の手の参考度の違いを確認し、本研究のアルゴリズムの有用性を調査できた。しかし、悪かった点や欲しい機能などの意見から、アルゴリズムの改善を行う必要性を再確認した。

5.3 実験 3:JINS MEME を用いた初心者が要求する情報の検証

本実験では、眼鏡型ウェアラブルデバイスである JINS MEME を麻雀初心者に着させ、1 人麻雀のプレイ中に任意の Web ページを用いて欲しい情報を検索させ、初心者のユーザ行動を取得し、麻雀初心者が要求する情報を検証した [5]。

本実験は実験 2 とは異なる 20 代男性 1 人を対象とし、JINS MEME を装着した状態で 1 人麻雀を 10 局プレイさせた。その際、被験者にはウェブブラウザを用いて必要と思う情報を検索させ、検索した Web ページを確認し、初心者が麻雀をする上で要求する情報を検証した。表 6 に初

表 6 初心者が検索した Web ページと参照時間

Web ページ	参照時間 (秒)
http://mjclv.com/yaku/#5	664.18
http://mjclv.com/yaku/y3.html	5.10

表 7 実験結果

	和了率	テンパイ率	平均得点
アルゴリズムなし	10%	70%	7800 点
Web 検索時	10%	30%	3000 点
アルゴリズムあり	30%	60%	3800 点

初心者が検索した Web ページと参照時間を示す。

表 6 に示した Web ページの URL を確認したところ、麻雀の役作成に関するものであった。今回の被験者は麻雀における役の作り方を知らない方のため、麻雀初心者が要求する情報として役の作成方法が必要ということを確認できた。また、検索したページの大半が 2 翻以下の役作成方法であり、参照時間全体の約 0.7% が 3 翻以上の役作成方法のページであった。今回の実験結果から初心者は Web ページを用いて検索する際、点数が高い役より上がる速度の速い 2 翻以下の役の作成方法を優先することを確認した。

また、追加実験として被験者には今回作成したアルゴリズムを用いて再度 10 局 1 人麻雀をさせ、例示したヒントのどちらを参考にしたかを検証した。なお、実験設定に関しては実験 2 と同様の条件にし、こちらも被験者に対しアンケートをした。表 7 に追加実験の結果を示す。

被験者には何も情報がない状態で 10 局、Web ページを用いて検索できる条件で 10 局、本研究で作成したアルゴリズムを用いて 10 局の計 30 局をプレイさせたが、アルゴリズムを使用した場合和了率が他の 2 つに比べ 20 % 上昇し、テンパイ率も 60 % と、3 局中 2 局でテンパイという結果が出た。この結果になった理由として、アルゴリズムなしの場合、麻雀における上がり方についてのみ理解しており、手牌を上げられる形に目指した結果テンパイ率が 70 % になったものと考えられる。また、Web 検索が可能な場合、役の作成方法については記述してあるが、実際の手牌からどのように作成して良いかわからず、結果聴牌率が 30 % になってしまったのに対し、本研究のアルゴリズムは現在の手牌から近い役を検索、表示をするものなので、前者に比べテンパイ率が上がったと考える。

また、本研究のアルゴリズムを用いて点数最大の手と手数最小の手のどちらを優先したか確認したところ、点数最大の手を参考にした割合が高かった。Web ページ検索時に対し役の作成が容易となり、より高い点数になるような選択をしたものと考えられる。しかし、実験後のアンケートでは、アルゴリズムの仕様に対し、どの牌をしてたら良いかわからない、結果が見にくいなどといった回答があったため、アルゴリズムの改善が今後の課題として挙げられた。

6. 今後の方針

本研究は 3 つのアルゴリズムを用いて、麻雀初心者に対し点数最大の手と手数最小の手の 2 種類ヒントを例示するシステムを構築したが、実験をした上で以下の修正点を発見した。

- 役判定アルゴリズムの不備
- 役作成ヒント例示アルゴリズムの最適化
- 役作成ヒント例示アルゴリズムの機能拡張

役判定アルゴリズムにおける七対子の場合の不備と、ドラの考慮や入力の際の簡略機能など、役作成ヒント例示アルゴリズムの機能拡張については随時行っていくが、役作成ヒント例示アルゴリズムの最適化が問題となっている。

実験 2 の時点で役作成ヒント例示アルゴリズムの使用条件をシャンテン数が 2 以下にしていた理由として、このアルゴリズムでは、現在の手牌状況から起こりうるすべての手牌を計算しているため、シャンテン数が増えると計算時間が指数関数的に増大してしまうからである。しかし、麻雀初心者はシャンテン数の制限を気にせず利用したいと考えるため、手牌探索方法の最適化をする必要がある。

また、今後の研究方針として、麻雀初心者にとってより有用なヒントを提示するために、麻雀上級者の打ち筋を学習し、提示する手牌の情報をより上級者に近づけていく。これにより初心者の思考過程をより支援できると考える。

7. おわりに

本研究では、麻雀初心者を対象に、麻雀をプレイするときに要求するヒントがどのようなものかを調査し、打牌選択時において現在の手牌から作成可能な役を列挙するシステムを、上がり判定アルゴリズム、役判定アルゴリズム及び役作成ヒント例示アルゴリズムからなる 3 つで構築し、実際に使用してもらい有用性を調査した。また、初心者が麻雀をプレイする上でどのような情報を検索するのかをウェアラブルデバイスである JINS MEME を用いて検証した。実験後のアンケートで麻雀中に別の可能性に気づくことができたという回答から、本研究のシステムの有用性を確認できた。また、麻雀初心者が役の作成手順に関する情報を要求することを JINS MEME の実験を通して確認した。

今後の課題として、今回作成したアルゴリズムの不備修正や最適化、機能拡張をし、麻雀上級者の打ち筋を学習し、初心者に対しより有用なヒントを例示する方法を考えることが挙げられる。

参考文献

- [1] 田中 悠, 池田 心: 麻雀初心者のための状況に応じた着手モデル選択, 第 31 回ゲーム情報学研究会, pp.1-8(2014).
- [2] 築地 毅, 柴原 一友: ディープラーニング麻雀-オートエ

- ンコーダとドロップアウトの有効性-, ゲームプログラミングワークショップ 2015 論文集, pp136-142(2015).
- [3] 水上 直樹, 中張 遼太郎, 浦 晃, 三輪 誠, 鶴岡 慶雅, 近山 隆: 降りるべき局面の認識による 1 人麻雀プレイヤーの 4 人麻雀への応用, ゲームプログラミングワークショップ 2013 論文集, pp1-7(2013).
- [4] とつげき東北, 伊藤 毅志: 牌譜の解析による麻雀の分析, 人工知能学会誌, Vol.24, No.3, pp355-360(2009).
- [5] 森下 真敬, 上野 未貴, 井佐原 均: MemeManual: Git 学習時の学習者行動に応じたマニュアルの自動生成, インタラクシオン 2017, 1-510-30, (2017)