

狙い撃ち手法による支援麻雀クライアントの開発 Development of an entertaining Mahjong Client by sharp shooting

浅尾 豪[†]
Go Asao

大西 建輔[‡]
Kensuke Onishi

1. はじめに

計算機が普及し始めた頃から、計算機を利用したボードゲームの研究開発がおこなわれている。計算機による麻雀のゲームの開発がいつ頃から始まったのかは定かではないが、1980年代には麻雀のゲームが存在している。

しかし、当時のクライアントには上質な思考ルーチンが存在せず、強いクライアントは牌交換をおこなっているものが多かった。現在では研究が進み、だんだんと純粋な思考ルーチンのみで強いクライアント [1] が作られてきた。最近ではモンテカルロ法を利用した研究 [2] も報告されている。

一方、私たちは強さを以外のことを求めるクライアントを作成しようと考えた。その一環として、安西の作成した支援をする麻雀クライアント (以下、支援既存と呼ぶ) の開発 [3] がある。

支援既存では、特定の相手 (以下、支援対象と呼ぶ) に対する危険度を下げることによって他の対戦相手よりも支援対象を多くあがらせようとしている。この支援既存を入れると支援対象の1位、3位をとる割合が上昇している。しかし、この支援既存には2つの欠点が存在する。1つめは、1人に対する危険度を下げることにより、全員を警戒するときよりも他の2人に対して振り込みやすくなってしまふ点である。2つめは、自分の平均持ち点が低くなるため、対戦相手にクライアントが支援をしていることを見破られてしまふ点である。

そこで我々は支援対象以外の特定の相手1人 (以下、標的と呼ぶ) を狙い撃ちするという手法を考えた。この手法では、標的を狙い撃つことで、支援既存の弱さと見破られやすさの改善を実現しようとしている。このクライアントを狙い撃ちクライアント (以下、支援狙撃と呼ぶ) とする。

2. 支援狙撃の実現方法

我々は”まうじゃん for java” [4] 上でクライアントを開発し、実験をおこなっている。”まうじゃん for java”には様々なクライアントが付属している。これらのクライアントでは、捨て牌選択時に、それぞれの手番に応じた評価値を計算し、捨て牌を決めている。自分の手牌を P 、手牌内の牌を p_i 、手牌内の捨てる牌を p_j とすると、評価値 $est(P, p_j)$ は、

$$est(P, p_j) = \sum_{i=1}^{14} (val(p_i)) + val(P) - risk(p_j)$$

で定義されている。 $val(p_i)$ は手牌 P 内のそれぞれの牌 p_i に対する評価値である。 $val(P)$ は手牌 P の面子

[†]東海大学大学院理学研究科数理学専攻

[‡]東海大学情報数理学専攻

に対する評価値である。また、 $risk(p_j)$ は捨て牌 p_j に対する危険度の評価値である。危険度の評価値は各対戦相手に対して計算するので、 $risk(p_j)$ は

$$risk(p_j) = \sum_{i=1}^3 (risk(u_i, p_j))$$

で定義されている。 $risk(u_i, p_j)$ は他家 u_i に対しての牌 p_j の危険度の評価値である。危険度の評価値はドラ牌、他家の捨て牌のスジ牌は危険と判断し、値を大きくする。また、完全安牌はドラでも危険度を0とする。

以上の評価値から基本的な麻雀のあがりを目指す通常クライアント (以下、自作と呼ぶ) を作成した。また、支援既存では捨て牌に対する危険度の評価値を

$$risk(p_j) = -risk(u_1, p_j) + risk(u_2, p_j) + risk(u_3, p_j)$$

と定義している。これは支援対象 u_1 に対しての捨て牌 p_j の危険度を通常と逆転させることで、支援対象に振込やすくしている。

一方、支援狙撃では捨て牌に対する危険度の評価値を

$$risk(p_j) = -risk(u_1, p_j) + 2 * risk(u_2, p_j) + risk(u_3, p_j)$$

と定義している。これは支援対象の危険度を逆転させたうえで、標的 u_2 に対する危険度の評価値を2倍にし、 u_2 に対して振り込まないようにしている。また、支援狙撃では u_2 以外からのロンあがりを禁止している。この制約から、リーチ時にフリテンになってしまう可能性があるため、リーチも禁止した。

3. 評価実験

3.1. 実験内容

支援麻雀の欠点において狙い撃ちをおこなう手法が有効であるかどうかを計算機実験を用いて検証する。

まず、クライアントの強さの検証として支援狙撃と”まうじゃん for Java”に付属されていた複数のクライアント (以下、既存と呼ぶ) を規定回数 (今回は半荘戦100回) だけ対戦させ、支援狙撃が標的からあがっているかを確認する。また、支援既存や自作とも対戦させ、結果を比べる。

事前に既存の中から無作為に選んだ4つのクライアント (びこたん、しゃち、面対子、なきのりう) を規定回数だけ対戦させ、その時の和了率と振込率の平均を求めた。

次に、見破られやすさの検証として複数のユーザ (今回は10人) に対戦してもらい、支援狙撃を発見できるか試してもらった。ユーザには事前知識として対面を狙い撃ちするクライアントを作成したことを伝えておいた。ユーザの対戦相手はこちらで指定する。前半の4

回は支援狙撃が参加している場合といない場合があり、ユーザには支援狙撃が参加していたかどうかを答えてもらった。このとき、支援狙撃が入っているかどうかを当てられた確率を発覚率として求めた。後半の4回は必ず支援狙撃が参加している状況で、どのクライアントが支援狙撃であったのかを答えてもらった。このとき、どのクライアントが支援狙撃かを見破られた確率を摘発率として求めた。

3.2. 実験結果

表1は、既存の4つの内、最も勝率が悪かった”なきのりう”をそれぞれ支援既存、支援狙撃、自作と入れ替えて規定回数だけ対戦させた時の和了率と振込率の表である。

表1: 自作, 既存, 支援の和了率と振込率

	和了率	振込率
支援既存	4.3%	8.1%
支援狙撃	3.9%	11.1%
自作	4.3%	10.4%
既存の平均	20.0%	15.3%

表2は、計算機実験でそれぞれのクライアントを規定回数だけ対戦させたときの勝率の表である。

表2: 順位の割合

	1位	2位	3位	4位
支援既存	4%	11%	30%	55%
支援狙撃	2%	8%	49%	41%
自作	3%	15%	35%	47%

ユーザ対戦における支援既存と支援狙撃の発覚率はそれぞれ47.5%, 47.5%であった。また、それぞれの摘発率は65.0%, 60.0%であった。

4. 考察

4.1. クライアントの強さ

表2から、自作の順位の割合は、3位と4位の割合を合わせると8割ほどある。このことから作成したクライアントは狙い撃ちをする以前から比較的弱いクライアントであるといえる。

弱い原因として考えられることは、振込率の高さである。だが、勝率の高いクライアントにも振込率が低いものが存在した。したがって振込率の高さは直接クライアントの強さと関係がないと考えられる。

次に考えられる弱さの原因は和了率の低さである。表1において、自作は全体的に和了率が低い。麻雀はあがることで他家から点数を奪うゲームなので、この和了率では点数が初期値からほとんど増えない。既存の中でも、一方、勝率が高いクライアントは和了率が20%を超えている。以上のことから、和了率の低さが弱い原因であると考えられる。

4.2. 支援狙撃と自作の比較

表1において、支援狙撃は自作に比べて和了率が減り、振込率が増えている。これは支援狙撃が標的以外からのロンあがりをしていないので、結果的に標的以外を

助けてしまっているためであると考えられる。また、表2において支援狙撃は自作に比べて2位の割合が減っている。これは他家を支援している為に上位になり難いためだと考えられる。

4.3. 狙い撃ちの効果

表3は狙い撃ち支援時と通常支援時における標的と支援対象の順位の平均の表である。

表3: 位置別の対戦相手の順位

	1位	2位	3位	4位
通常時の標的	36%	31%	17%	16%
狙撃時の標的	32%	33%	20%	15%
通常時の支援対象	27%	31%	19%	23%
狙撃時の支援対象	33%	32%	24%	11%

表3より、狙撃時と通常時では標的の1位の割合が下がっている。一方、支援対象の順位は全体的に上がっている。このことから狙い撃ち戦術によって標的と支援対象の順位に影響を与えることができていると考えられる。

4.4. 見破られやすさ

ユーザ実験より、支援狙撃の発覚率は5割未満であったが、摘発率は6割を超えていた。このことから支援狙撃は支援クライアントの参加が不明な状況では期待値通りの結果だったが、確実に参加している状況では見破られてしまっていたといえる。

また、支援狙撃と支援既存を比べると大差ない結果となった。以上のことから、狙い撃ち手法は支援クライアントの見破られやすさに対して、有用な効果を発揮することはできなかった。

5. 結論と今後の展望

今回の検証結果から、狙い撃ち手法での下家を支援する効果は、通常の支援時に比べて有効な結果を導くことができた。しかし、当初の目的である支援クライアントの強さ、見破られやすさの改善の実現に関しては、大きな成果はみられなかった。したがって、狙い撃ち手法は強さの向上、見破られやすさの改善策としては不十分であるといえる。

今回の実験から、クライアントの弱さの原因は和了率の低さにあると考えられる。そのため、和了率を向上させ、見破られやすさの改善を図っていきたい。

参考文献

- [1] AI麻雀 <http://game.e-frontier.co.jp/ai/mahjong13/>, 2014/4/30
- [2] 三木理斗, 多人数不完全情報ゲームにおける最適行動決定に関する研究, 2010年度東京大学工学系研究科電気系工学専攻修士論文
- [3] 安西諒祐, 接待もできる麻雀AIの開発, 2012年度東海大学情報数理学科卒業論文
- [4] 石畑恭平, コンピュータ麻雀のアルゴリズム, 工学社, 2007