

プレイヤーに合わせて難易度を調整するカラハ AI の研究

LI ZHENGYU[†] 横山大作[‡]明治大学理工学研究科[†] 明治大学理工学部[‡]

1. はじめに

多様な対戦相手を確保したり、人間を教育するなどの目的のためには、人間に対して強すぎる、あるいは弱すぎるような AI ではなく、プレイヤーに応じて難易度を調整する AI を構築することが重要な課題である。

本研究では、プレイヤーが指すべき手を選ぶとき、可能な局面の中で評価値が中間程度の値を持つような手を選ぶようにすれば、プレイヤーの強さが調整できるのではないかと考えた。この考えに基づき、古くからアフリカや中近東、東南アジアで遊ばれている石取りゲームの一種であるカラハを題材として、Q-Learning[1]による強化学習で局面の評価値を学習し、その評価値をもとにゲーム中に戦略を調整するプレイヤーを作成した。ランダムプレイヤーを用いた対戦実験を行い、提案手法の有効性を評価した。また、Minimax Tree Search を用いたプレイヤーを作成し、探索深さを変化させて強さを変化させた時との比較を試みた。

2. プレイヤーの構築

(1) Minimax Tree

Minimax Tree の評価関数は行動した後のストア内の石の数の差である。ゲームオーバーの場合、自分が勝った時は 5000、相手が勝った時は -5000、引き分けの時は 0 である。

(2) Q-Learning

Q-Learning の Reward 値は行動した後のストア内にある石の数の差を石の総数で割った後の値である。

Q-Learning の AI は、対戦する前に、まず QTable が収束するまで十分にトレーニングしておく。

3. AI の強さの調整方法

(1) Q-Learning

Q-Learning プレイヤーの強さ調整は、QValue から行動を決定する戦略を変えることで行う。

一試合の間に、相手の行動に相応する QValue に合わせて、AI は自分が選択すべき selectQValue を調整していく。以下の式で定義する selectQValue を設定し、この値に最も近い QValue を持つ局面への手を指すことにする。

$$\text{selectQValue} = \alpha * (\max\text{QValue} + 2) + \min\text{QValue}$$

maxQValue と minQValue はその局面で可能な行動を持つ値のうち最大と最小の値である。 α は割引率で、ゲームが始まる時に 0 から 1 までの範囲内でランダムに設定される。ゲームを開始し、相手が一手指すごとに α を以下の式でアップデートする。

$$\alpha = (1 - \beta) * \alpha + \beta * \text{pre } \alpha$$

β は学習率で、pre α は相手の行動についての α 値である。相手がどの程度最善の手を指すのか、に応じた値に α を設定することが目的である。

Research on Kalah AI that adjusts the difficulty level according to the player

LI ZHENGYU[†]
Yokoyama Daisaku[‡]
Graduate School of Science and Technology, Meiji University[†]
School of Science and Technology, Meiji University[‡]

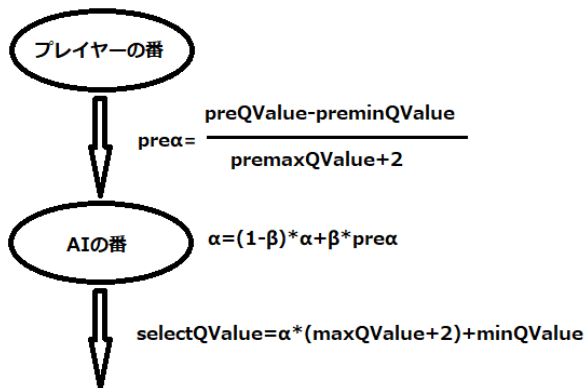


図1 SelectQValue の計算

(2) Minimax Tree

Minimax Tree プレイヤーの強さの調整は、探索木の深さを調整することで行う。深さを増やせば強くなると期待できる。

4. 評価実験

AI の対戦相手はランダムで行動する AI とする。先手と後手の選択はランダムで決まる。Q-Learning プレイヤーは selectQValue を用いる場合と、比較の為に maxQValue、minQValue を用いるものを用意する。Minimax Tree プレイヤーは探索深さを1から12まで変化させたものを用意する。これらのプレイヤーとランダム AI との対戦実験を1000回ずつ行い、勝率を求める。Q-Learning 手法の selectQValue を求めるために設定した α 値は新しい対戦をする前にリセットする。

N3M3 と N4M2 の二種類の初期盤面に対する対戦実験を行う。N は一辺の穴の数、M は対戦が始まる前に一つの穴にあった石の数である。

表1に対戦結果を示す。Minimax Tree プレイヤーとの対戦結果を見ると、探索木の深さを小さくしてもほとんど Minimax Tree の AI が勝っており、うまく強さが調整できていないことがわかる。また、深さを変化させた時に勝率が上がったり下がったりしており。単調に調整できていない点も問題である。

		N4M2	N3M3
Minimax	depth=1	17.10%	16.00%
	depth=2	24.35%	7.30%
	depth=3	6.50%	4.80%
	depth=4	6.20%	8.80%
	depth=5	6.20%	2.50%
	depth=6	9.90%	2.50%
	depth=7	10.30%	2.90%
	depth=8	11.50%	3.20%
	depth=9	10.80%	0.50%
	depth=10	12.60%	4.20%
	depth=11	10.60%	0.10%
	depth=12	11.20%	0.30%
Q-Learning	maxQValue	8.90%	9.50%
	minQValue	79.70%	64.90%
	selectQValue	49.60%	43.60%

表1 ランダムプレイヤー側から見た対戦結果

Q-Learning AI との対戦結果を見ると、maxQValue と minQValue を選択することで勝率を正しく変化させられたことがわかる。そして selectQValue を用いることで、おおむね 50% の勝率を達成することができた。

5.まとめ

本研究では、Minimax Tree と Q-Learning を利用した AI を作成し、ランダムプレイヤーとの対戦において、強さを調整できるかを評価する実験を行った。

Minimax Tree 手法で、探索深さを変化させることで難易度を調整することはうまくいかなかった。

Q-Learning 手法の場合、maxQValue と minQValue を選択すれば強さ調整はできるが、プレイヤーにとって強すぎる、あるいは弱すぎるプレイヤーになった。プレイヤーの行動に応じて selectQValue を調整する AI を用いると、プレイヤーにとってちょうど良い強さの AI が構築できる可能性が示された。これは教育用途などに利用できると考えている。

参考文献

[1] Watkins, C.J.C.H., Dayan, P. Q-learning. Machine Learning 8, pp.279-292(1992)