マンカラにおける深層強化学習アルゴリズムの比較

東京都立立川高等学校　3年　山本 勇太

01. はじめに

マンカラとは、２人で交互に石を動かして遊ぶ、世界最古のボードゲームの一つ(図１)。世界中に様々なルールがある[1]。本研究では図２のような独自のルールを用いた。

強化学習とは、行動主体であるエージェントが環境との相互のやり取りによって長期的に見て最適な行動を学習する機械学習の一種である[2]。深層強化学習は、強化学習の学習モデルに深層学習モデル(多層のニューラルネットワーク)を用いたもののこと。

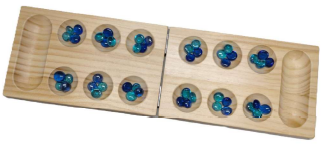

図１　マンカラ(筆者撮影)

● 本研究で設定したルール(図２)

① 図のように各ポケットに石が４つずつ入った盤面を用意。
② 自分の番には、自分のポケットから１つ選び、中の石を反時計回りに１つずつ順に入れる(青矢印)。最後の石が両端のストアに入ったらもう一回自分の番。
③ 石の移動の際、最後の石が空ポケットに入ったら向かいのポケットの中身と最後の石を自分のストアへ。ただし、向かいのポケットに石がない場合は行わない。
④ 一方のプレイヤーのポケットから石がなくなった時点で、ポケットとストアに入っている石の合計が多い方の勝ち。

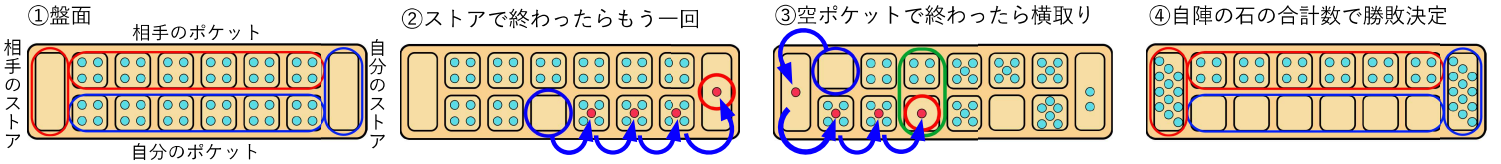

図２　本研究で設定したルール

02. 研究背景

以前の研究[3]で縮小版のマンカラの盤面・ルールでQ学習を行ったが、Qテーブルがあまりにも大きく、一般的な盤面・ルールに適応するには無理があった。そこで、深層強化学習のアルゴリズムを使って勝率の高いプレイヤーの作成を試みた。一般的な盤面・ルールに近づけるにあたり、Kalahのルール[1]を拡張した独自のルール(図２)とKalahと同じ盤面を使用した。

Hunterらによる先行研究[4]では、A3Cやモンテカルロ木探索等に基づいたエージェントを計６つ作成して強いマンカラのプレイヤーの開発が試みられ、互いに対戦させた結果、Advanced Heuristic Minimax(AHM)という手法が計算コストも勝率も優れていたため、マンカラに最適だと結論付けられた。

本研究では、深層強化学習のアルゴリズムに焦点を当てて勝率の高いプレイヤーの作成を試みるとともに、その性能の比較を行った。

03. 方法

① 以下の3種類のプレイヤーを互いに対戦させ、10万エピソードの学習を行った。(1エピソード = 勝敗がつく or 無効な手(ミス)を打つ まで)
② 学習結果の確認のため、Randomプレイヤーと1,000回対戦させた。
③ 無効な手(ミス)を打たないように、有効な手の中からQ値や選択確率の高い手を打つように設定し、再度②を行った。

○プレイヤー(アルゴリズム) ※文献[2]を参考にPythonで実装

1. **Random** … 有効な手をランダムに打つ。
2. **Deep Q Network (DQN)** … 各行動の評価値を算出するQ関数をニューラルネットワークで表現し学習する。
3. **Acter Critic (AC)** … 方策(行動決定の指針)と価値関数(行動の評価)を別々にニューラルネットワークで表現し学習する。

04. 結果

学習中の勝率等の推移は図３から図10のように、学習時間は表１のようになった(図３,４,７,８の凡例はDQNやACを主体としており、図のタイトルは「先手vs後手」のようになっている)。学習結果の確認のためのRandomとの対戦(方法②③)の結果は、それぞれ表２,３のようになった。参考として、Random同士で対戦させた際の勝率は、先手 49.7 %、後手 44.2 %、引き分け 6.1 %となった。

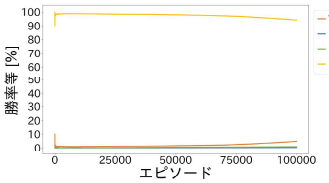

図３　DQN vs Random

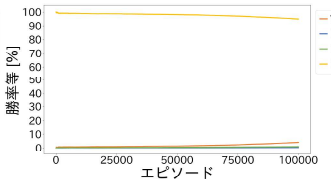

図４　Random vs DQN

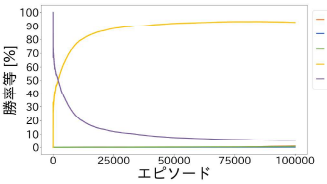

図５　DQN vs AC

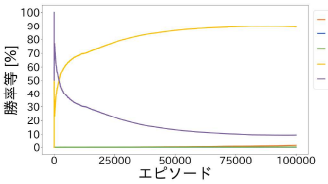

図６　AC vs DQN

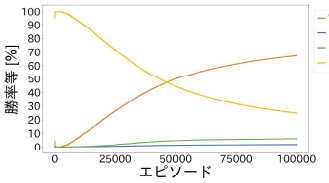

図７　AC vs Random

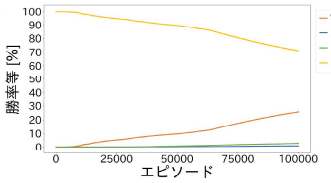

図８　Random vs AC

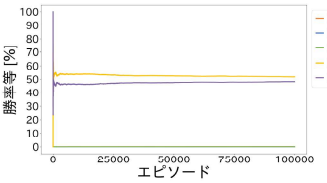

図９　DQN vs DQN2

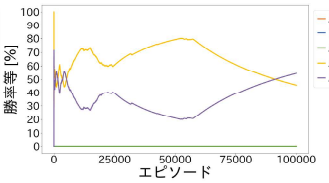

図 10　AC vs AC2

表１　学習時間

学習時間		後手		
		Random	DQN	AC
先手	Rand	-	54m 16.5	21m 28.7
	DQN	55m 58.3	63m 07.9	79m 33.9
	AC	35m 08.1	69m 57.1	36m 46.7

※Google ColaboratoryのCPUで実行

表２　学習結果の確認(ミスあり、vs Random) [%]

勝 / 敗 / 分 / ミス		学習相手		
		Random	DQN	AC
先手	DQN	78.1 / 9.6 / 2.7 / 9.6	40.8 / 54.1 / 4.4 / 0.7	77.4 / 13.9 / 5.3 / 3.4
	AC	86.5 / 7.6 / 2.0 / 3.9	0.0 / 0.0 / 0.0 / 100.0	15.3 / 5.8 / 0.9 / 78.0
後手	DQN	77.7 / 9.4 / 3.3 / 9.6	0.0 / 0.0 / 0.0 / 100.0	50.9 / 25.8 / 5.1 / 18.2
	AC	50.1 / 4.9 / 1.9 / 43.1	0.0 / 0.0 / 0.0 / 100.0	0.0 / 0.0 / 0.0 / 100.0

表３　学習結果の確認(ミスなし、vs Random) [%]

勝 / 敗 / 分 (ミスなし)		学習相手		
		Random	DQN	AC
先手	DQN	78.4 / 17.8 / 3.8	41.0 / 54.5 / 4.5	79.1 / 16.2 / 4.7
	AC	88.6 / 9.1 / 2.3	21.0 / 73.7 / 5.3	44.5 / 51.4 / 4.1
後手	DQN	76.9 / 18.7 / 4.4	43.6 / 49.4 / 7.0	51.0 / 43.2 / 5.8
	AC	67.2 / 28.7 / 4.1	36.2 / 58.3 / 5.5	38.6 / 57.9 / 3.5

05. 考察

表２より、学習相手がDQNの場合ミスの学習が全くできておらず、唯一学習できた先手のDQNも勝敗に関する学習は全くできていない。これは図３等からわかるように(DQNは学習中に一部ランダムに打つため)DQNの学習中のミスが多く、相手の学習が進まないからだと考えられる(AC同士の対戦も同様)。対策としてミスを打った際に負の報酬を与えたうえで別の手を打ち対戦を続行することが考えられる。勝率や学習時間を考えると、最も効率良く強いプレイヤーを作成できるのは、学習相手がRandomのときで、先手は勝率88.6%のAC、後手は勝率76.9%のDQNだと考えられる(勝率は表３)。

06. まとめ

以前の研究の反省を基に、独自ルールのマンカラにおいて自作のPythonプログラムで深層強化学習を行い、勝率の高いプレイヤーの作成とアルゴリズムの性能の比較を行った。強化学習同士の学習は上手くいかないことが多かったが、Randomとの学習では高い勝率となり、先手はAC、後手はDQNが勝率と学習時間の観点から最適だと分かった。今後は、強化学習同士の学習ができなかったという問題への対処を含めてプログラムを改良し、更なる勝率の向上を目指していきたい。また、他の深層強化学習アルゴリズムについても同様の比較を行っていきたい。

07. 参考文献

[1] “マンカラルール”. 公益財団法人日本レクリエーション協会. https://recreation.or.jp/activities/genki_up/mancala/. (参照 ’24-10-13)
[2] 斎藤康毅. “ゼロから作るDeep Learning ❹ 強化学習編”. オライリー・ジャパン. 2022. 376p. ISBN978-4-87311-975-5.
[3] 山本勇太. “強化学習を活用したマンカラの最善手の探索” (2024). 第6回中高生情報学研究コンテスト. https://www.ipsj.or.jp/event/taikai/86/86PosterSession/ipsj_poster/index.html (参照 ’24-10-13)
[4] Hunter, Trevon J., “The Exploration and Analysis of Mancala from an AI Perspective” (2021). Honors Theses. 257. https://digitalcommons.andrews.edu/honors/257/. (参照 ’24-10-13)