Vol.2022-GI-47 No.19
2022/3/19

ターン制戦略ゲームにおける優先順位付き経験再生型深層学習の適用

竹内裕哉 [1,a)]　松原仁 [2.a)]

概要：本研究は，ゲームにおいて人間より優れた行動選択を行うシステムの開発を目的とする．ターン制戦略ゲームジャンルを対象として，優先順位付き経験再生型深層強化学習アルゴリズムを導入することで優れた行動選択を行うゲーム AI の開発を試みた．本研究は，Atari2600 環境において人間のスコアを上回った Ape-x と Ape-x を改良しスコアを大幅に上回った R2D2 という二つの手法に着目した．自作のゲーム環境で Ape-x の実装を行い，R2D2 は実装の可能性を述べた．実験では Ape-x と DQN との二つの比較実験を行った．その結果，DQN と比べて Ape-x は高い性能を見せるが，予想以下の性能となった．予想以下の性能となった考察として，固定された盤面があまり存在しないことと Actor の数が少なかったことが原因だと考えられる．

キーワード：ゲーム AI，意思決定問題，Ape-x

Prioritized Experience-Replay deep learning in turn-based strategy game

HIROYA TAKEUCHI[1,a)]　HITOSHI MATSUBARA[2)]

Abstract: The purpose of this study is to develop a system that can make better action choices than humans in games. We attempted to develop a game AI that can make superior action choices by introducing a prioritized experience-replaying deep reinforcement learning algorithm for the turn-based strategy game genre. We focused on two methods: Ape-x, which outperformed the human score in the Atari 2600 environment, and R2D2, which improved on Ape-x and significantly outperformed it. We implemented Ape-x in our own game environment, and described the possibility of implementing R2D2. In our experiments, we compared Ape-x with DQN. As a result, Ape-x showed higher performance than DQN, but the performance was lower than expected. The reason for the lower performance is that there were not many fixed boards and the number of Actors was small..

Keywords: Game AI, Decision-making problems, Ape-x

1. はじめに

近年, 将棋や囲碁など 2 人完全情報ゲームで AI が人間のトッププレイヤーに勝利するなど，強い AI の研究は大きく進んでいる. 2016 年に Google DeepMind 社が AlphaGo を発表しトップ棋士の一人，イ・セドルを破ったことで世界を驚愕させた[1]. 2017 年には，同社が後続として AlphaZero を開発し，トップ棋士を下した AlphaGo に対して 100 戦 100 勝と大きくその性能を発展させた. 将棋の分野では, 2013 年, 2014 年にプロ棋士 VS コンピュータの電王戦が行われ，コンピュータ側の圧勝であった．2017 年には，Ponanza がトッププロである佐藤天彦名人に勝利した.

また，現在の AI 技術は囲碁やチェスなどのアナログゲームだけに留まらず, PlayStation4 や Xbox Series X などのコンシューマゲーム機にもゲーム AI 技術は大きな応用の可能性が示されている．三宅(2008)は、ゲーム AI 技術を研究することで人工知能の研究に活躍の場が与えられると述べている[2]．また三宅(2015)は，ゲーム産業における人工知能技術はこの 20 年で進歩し，アカデミックな人工知能研究の成果を取り組む土台が出来上がった．今後はゲーム産業とアカデミック

1 公立はこだて未来大学大学院システム情報科学研究科
Graduate School of System Information Science、Future University Hakodate
a) g2120025@fun.ac.jp
2 はこだて未来大学
Future University Hakodate
a) matsubar@fun.ac.jp

な研究の協調体制がお互いの進化を加速していくことを期待していると述べている [3].

本研究は人間より優れた行動選択を行うゲーム AI の開発を目指す．優れたゲーム AI の研究において，囲碁や将棋などより行動選択，合法手数が多いゲームジャンルでは人間より優れた結果を出すAIが少ない．そこで本研究は，ターン制戦略ゲームジャンルを対象として，優先順位付き経験再生型深層強化学習アルゴリズムを導入することで優れた行動選択を行うゲームAIの開発を行った．

2. ターン制戦略ゲームと TUBSTAP について

この章では，ターン制戦略ゲームと本研究の実験環境の参考とした TUBSTAP について説明する

2.1 ターン制戦略ゲーム

ターン制戦略ゲームあるいはターン制ストラテジーゲームとは，プレイヤがターンごとにある場（フィールド，盤面）において複数の駒を任意の個数分操作し，勝利条件を目指すゲームジャンルの一種である．原義によれば，将棋，チェスや囲碁などもターン制戦略ゲームの一種であると言える．三宅(2021)は自著でターン制戦略ゲームの定義を次のように採用している[4].

・場(フィールド)があり，場を俯瞰する視点がある
・キャラクター(駒)あるいはその集団があり，俯瞰視点からプレイヤが指示を与えることができる　指示の形式は位置への移動，目的の遂行などさまざまである
・指示を与えられたキャラクターあるいはその集団は，指示通りに，あるいは目的を遂行するために自律的に行動する
・勝利条件や達成目標がある

2.2 TUBSTAP

本研究では実験環境として TUBSTAP[5]を参考にした．TUBSTAP とは Turn-Based Strategy Games as an Academic Platform の略称である．北陸先端科学技術大学院大学がターン制戦略ゲームの学術用基盤プラットフォームとして提唱したゲームである．また，任天堂社が開発したコンピュータゲーム Advance Wars Day of ruin[6]を基に簡略したものである．TUBSTAP はターン制戦略ゲームの様々な研究において実験環境として利用されている．TUBSTAP の大まかなルールは以下の点である．

・盤面，マップの設定は自由
・対戦相手は 2 人
・駒の種類は限定されている
・1 ターンでそれぞれの駒を任意に操作できる
・マップにはそれぞれ要素が存在する

ターン制戦略ゲームの中には非完全情報型のゲームも存在するが，TUBSTAP は駒やマップなどの情報をプレイヤがすべて把握できる完全情報ゲームである．TUBSTAP における駒とマップの種類は図 2 に示されているとおりである．本研究の実験環境としてこの TUBSTAP を参考にし，Python 用として開発し簡単なゲーム環境を実験環境として用いた．

ユニット	射程	攻撃力						移動力	移動コスト					
		F	A	P	U	R	I		道路	平原	林	山	海	陣地
戦闘機 F	1	55	65	0	0	0	0	9	1	1	1	1	1	1
攻撃機 A	1	0	0	105	105	85	115	7	1	1	1	1	1	1
戦車 P	1	0	0	55	70	75	75	6	1	1	2	-	-	1
自走砲 U	2-3	0	0	60	75	65	90	5	1	1	2	-	-	1
対空戦車 R	1	70	70	15	50	45	115	6	1	1	2	-	-	1
歩兵 I	1	0	0	5	10	3	55	3	1	1	1	2	-	1

図 2　TUBTSAP 内の各設定

3. 関連研究

この章では TUBSTAP に強化学習と深層学習を適用した研究を紹介する．

3.1 強化学習を適用した研究

藤木ら(2014)はターン制ストラテジーゲームにおける行動選択に対し防御的な手を重視する深さ限定モンテカルロ探索:Depth-Limited Monte-Carlo(DLMC)[7]を提案している．この研究では，TUBSTAP を対象に，新たに提案した深さ限定モンテカルロ法と通常のモンテカルロ法を同シミュレーション数，同サンプル数で標準 AI と戦わせた実験を行った．結果として，勝率は僅差であったが深さ限定モンテカルロ法が上回り，深さ限定モンテカルロ法は 10 倍ほどモンテカルロ法より高速であったとされている．

また武藤ら(2015)は TUBSTAP を対象に，思考アルゴリズムの研究を行った[8]. ターン制ストラテジーゲームは，同一ターン中に複数の駒を好きな順番で動かせるため，探索空間が膨大となり優れた AI を作るのが困難な対象である．この研究では，ユニット行動木 UCT(Upper Confidence Tree)探索を新たに提案し，藤木らの深さ限定モンテカルロ法と比較を行い，結果とし

てユニット行動木 UCT 探索の方が勝率が高いことが示されている．

3.2 深層学習を適用した研究

2018 年ごろから深層学習や深層強化学習を実装する研究が見られるようになってきた．本研究の関連研究として 2019 年には TUBSTAP に深層学習の適用を試みた研究がある．木村(2019)は深層学習である RNN と CNN をターン制戦略ゲームに適用した[9]．ネットワークの出力段を移動元，移動先，攻撃先の 3 つに分割することで必要ニューロン数を低減させて TUBSTAP に導入することに成功した．また，一方で DQN 単体ではターン制戦略ゲームにおける有効手の学習成功割合は悪かった．しかし Profit Sharing を組み合わせることで学習の成功割合を上げることに成功した．

図 3 は木村の実験のエピソード成功率の割合を示している．1 イタレーション当たり 4000 ステップ実行しており，そのエピソード成功率の割合が 40%未満と半分を下回っている．ここでのエピソードとは，1 ゲーム開始時から自分または相手の駒どちらかが全滅した時点で 1 エピソードとする．

この研究から TUBSTAP に深層学習，深層強化学習が適用でき，そのゲーム AI の性能を上げることができると考えた．

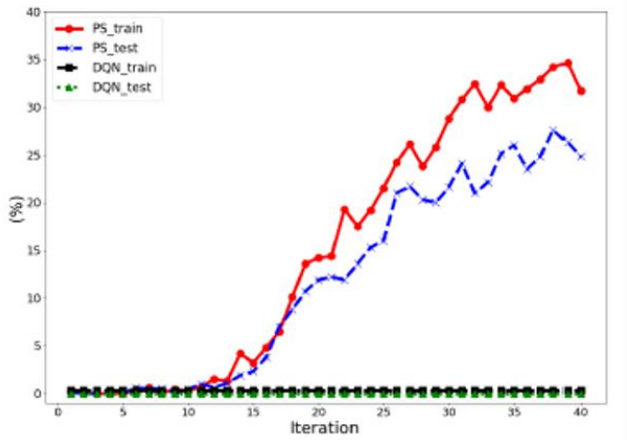

図 3　木村(2019)の実験結果

4. 深層強化学習

この章では，本研究で実装する Ape-x とその改良手法である R2D2 について説明する．

4.1 Ape-x

Ape-x[10]は 2018 年に開発された優先順位付き経験再生型深層強化学習である．Ape-x は一つの Learner と複数の Actor とリプレイメモリで構成されている．通常の深層強化学習手法とは異なり，リプレイメモリにそれぞれ優先度をつけ，より有用なリプレイメモリを頻繁にサンプリングする．優先順位が高いデータを共有し，それに基づいた Actor の行動を Learner が学習することで従来の手法より性能を高くしている．

4.2 R2D2

R2D2[11]は Ape-x を改良した手法である．R2D2 での改良点は大きく二つある．一つは，R2D2 は Recurrent Network の一種である LSTM を使用する．LSTM を使用することで時系列的特徴量を抽出し，より正確な Q 値を推定することが可能になる．2 つ目は，内部状態もリプレイメモリに保存している．これは、経験再生の際に初期内部状態を正確に復元させることが目的である．そして，学習の中で最新のネットワークに存在する内部状態と初期内部状態には差が生まれる．初期内部状態を正確に復元することでその差を解消し，学習の精度を高めることができる．

図 4.1 と図 4.2 が Atari2600 での人間と Ape-x と R2D2 のスコアを比較した図である．青文字の部分がそれぞれのゲームでの最大スコアである．二つの図を見ると明らかだが，大半のゲームにおいて R2D2 が Ape-x のスコアを上回っていて，上回っているゲームのスコアの中では数倍以上のスコアを出しているものもある．一方で，Ape-x が R2D2 を上回っているゲームが複数ある．このように R2D2 が Ape-x を改良した手法であるのにもかかわらず，Ape-x にスコアで負けた部分の差が存在する．本研究では，ターン制戦略ゲームジャンルにおいて Ape-x の性能を調査し，R2D2 の適用の可能性を述べる．

GAMES	HUMAN	REACTOR	IMPALA(S/D)	APE-X	R2D2
alien	7127.8	6482.1	1536.0/15962.1	40804.9	**229496.9**
amidar	1719.5	833.0	497.6/1554.8	8659.2	**29321.4**
assault	742.0	11013.5	12086.9/19148.5	24559.4	**108197.0**
asterix	8503.3	36238.5	29692.5/300732.0	313305.0	**999153.3**
asteroids	47388.7	2780.3	3508.1/108590.1	155495.1	**357867.7**
atlantis	29028.1	308257.5	773355.5/849967.5	944497.5	**1620764.0**
bank_heist	753.1	988.7	1200.3/1223.2	1716.4	**24235.9**
battle_zone	37187.5	61220.0	13015.0/20885.0	98895.0	**751880.0**
beam_rider	16926.5	8566.5	8219.9/32463.5	63305.2	**188257.4**
berzerk	2630.4	1641.4	888.3/1852.7	**57196.7**	53318.7
bowling	160.7	75.4	35.7/59.9	17.6	**219.5**
boxing	12.1	99.4	96.3/**100.0**	**100.0**	98.5
breakout	30.5	518.4	640.4/787.3	800.9	**837.7**
centipede	12017.0	3402.8	5528.1/11049.8	12974.0	**599140.3**
chopper_command	7387.8	37568.0	5012.0/28255.0	721851.0	**986652.0**
crazy_climber	35829.4	194347.0	136211.5/136950.0	320426.0	**366690.7**
defender	18688.9	113127.8	58718.3/185203.0	411943.5	**665792.0**
demon_attack	1971.0	100188.5	107264.7/132827.0	133086.4	**140002.3**
double_dunk	-16.4	11.4	-0.4/-0.3	12.8	**23.7**
enduro	860.5	2230.1	0.0/0.0	2177.4	**2372.7**
fishing_derby	-38.8	23.2	32.1/44.9	44.4	**85.8**
freeway	29.6	31.4	0.0/0.0	**33.7**	32.5
frostbite	4334.7	8042.1	269.6/317.8	9328.6	**315456.4**
gopher	2412.5	69135.1	1002.4/66782.3	120500.9	**124776.3**
gravitar	3351.4	1073.8	211.5/359.5	1598.5	**15680.7**
hero	30826.4	35542.2	33853.2/33730.6	31655.9	**39537.1**
ice_hockey	0.9	3.4	-5.3/3.5	33.0	**79.3**
jamesbond	302.8	7869.3	440.0/601.5	21322.5	**25354.0**
kangaroo	3035.0	10484.5	47.0/1632.0	1416.0	**14130.7**
krull	2665.5	9930.9	9247.6/8147.4	11741.4	**218448.1**
kung_fu_master	22736.3	59799.5	42259.0/43375.5	97829.5	**233413.3**

図 4.1　R2D2 と Ape-x と人間のスコアの比較

-1

montezuma_revenge	**4753.3**	2643.5	0.0/0.0	2500.0	2061.3
ms_pacman	6951.6	2724.3	6501.7/7342.3	11255.2	**42281.7**
name_this_game	8049.0	9907.1	6049.6/21537.2	25783.3	**58182.7**
phoenix	7242.6	40092.3	33068.2/210996.5	224491.1	**864020.0**
pitfall	**6463.7**	-3.5	-11.1/-1.7	-0.6	0.0
pong	14.6	20.7	20.4/**21.0**	20.9	**21.0**
private_eye	**69571.3**	15177.1	92.4/98.5	49.8	5322.7
qbert	13455.0	22956.5	18901.3/351200.1	302391.3	**408850.0**
riverraid	17118.0	16608.3	17401.9/29608.0	**63864.4**	45632.1
road_runner	7845.0	71168.0	37505.0/57121.0	222234.5	**599246.7**
robotank	11.9	68.5	2.3/13.0	73.8	**100.4**
seaquest	42054.7	8425.8	1716.9/1753.2	392952.3	**999996.7**
skiing	**-4336.9**	-10753.4	-29975.0/-10180.4	-10789.9	-30021.7
solaris	**12326.7**	2165.2	2368.4/2365.0	2892.9	3787.2
space_invaders	1668.7	2448.6	1726.3/43595.8	**54681.0**	43223.4
star_gunner	10250.0	70038.0	69139.0/200625.0	434342.5	**717344.0**
surround	6.5	6.7	-8.1/7.6	7.1	**9.9**
tennis	-8.3	23.3	-1.9/0.5	**23.9**	-0.1
time_pilot	5229.1	19401.0	6617.5/48481.5	87085.0	**445377.3**
tutankham	167.6	272.6	267.8/292.1	272.6	**395.3**
up_n_down	11693.2	64354.3	273058.1/332546.8	401884.3	**589226.9**
venture	1187.5	1597.5	0.0/0.0	1773.5	**1970.7**
video_pinball	17667.9	469365.8	228642.5/572898.3	546197.4	**999383.2**
wizard_of_wor	4756.5	13170.5	4203.0/9157.5	46204.0	**144362.7**
yars_revenge	54576.9	102759.8	80530.1/84231.1	148594.8	**995048.4**
zaxxon	9173.3	25215.5	1148.5/32935.5	42285.5	**224910.7**

図 4.2　R2D2 と Ape-x と人間のスコアの比較-2

5. 実験内容

この章では実験環境と実験Ⅰ，Ⅱについて説明する．

5.1 実験環境

Ape-x 手法について以下の環境で実験Ⅰ，実験Ⅱを行った．本研究では，自作の簡易型ターン制戦略ゲームを実験環境とした．OpenAI gym の自作環境を参考に gym 環境での実験を行った．自作ゲーム環境においてのルールが以下の通りである．

・駒は敵味方 1 つずつ(拡張次第で増減可)
・駒の要素(HP，攻撃力，移動力)，マップの要素(地形要素、マップサイズ)は変更できる
・駒の行動は 1 つの行動で 1 マスずつの移動と攻撃を伴ったものになる
・駒同士での有利不利は存在しない
・どちらかの駒の HP が 0 になった場合もしくは規定ターン数までの行動でエピソード終了
・敵駒の行動はランダム
・報酬の与え方は，敵駒を倒した場合 100，移動が可能であるマスに移動した場合 1，不適切な行動をした場合-1 をそれぞれの場合に対して与える
・不適切な行動をした場合，ターンを終了せず適切な行動をとるまでそのターンを終了しない

gym 環境では，Atari2600 のようなゲーム群が存在し，どのゲームもゲーム画面が描画されるが，自作のゲーム環境では描画なしでの設計にした．本研究で使用したマップは以下の 3 つのマップである．マップ表示の描画は TUBSTAP を使用した．図 5.1 は実装手法が基本的な行動が可能かどうか判別するためのマップである．図 5.2 は中央に行動できないマスを設定し，障害マスを避ける行動をとれるかどうか判別するためのマップである．図 5.3 は迷路状に設定し，移動できないマスが多い場合に適切な行動がとれるかどうか判別するためのマップである．

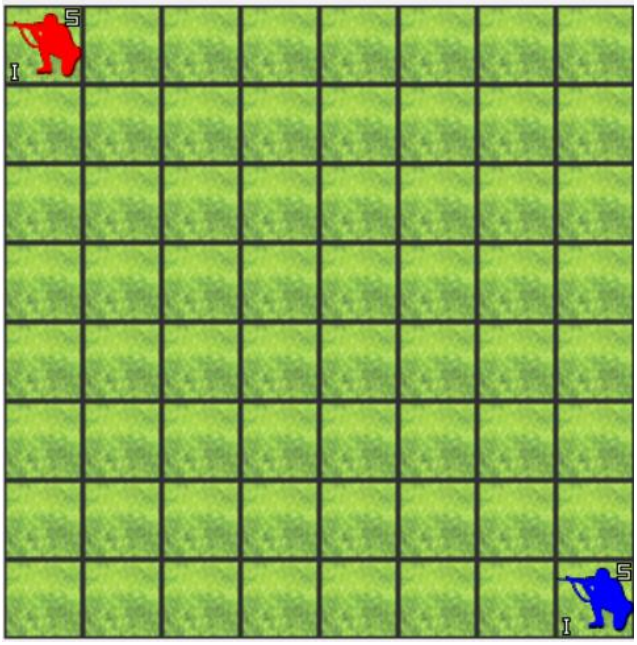

図 5.1　実験マップ 1

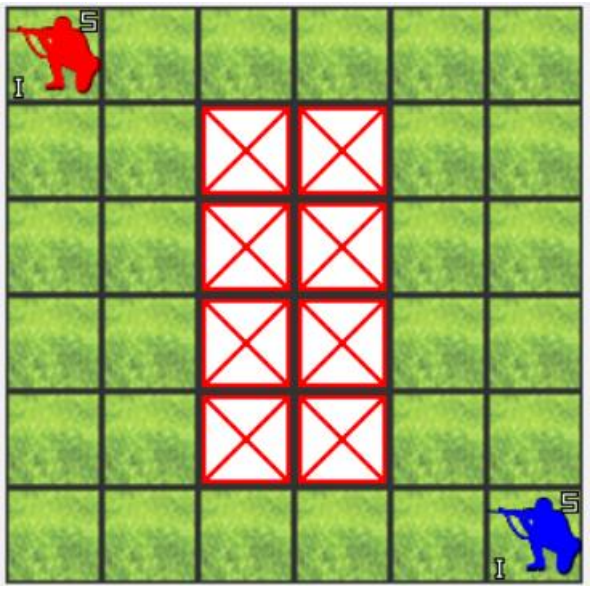

図 5.2　実験マップ 2

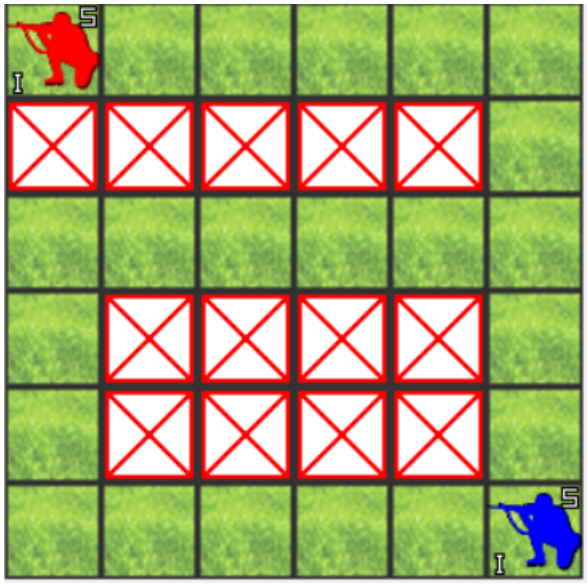

図 5.3　実験マップ 3

5.2　実験 I

実験 I は自作ゲーム環境で 3 つのマップにおいて，DQN と Ape-x が適応できるかどうか判別するための実験を行った．対戦相手の駒を全滅させるか規定のターン数まで伸びた場合は規定ターン数までの行動を 1 エピソードとする．

実験 I において，駒の移動を上下左右とその場に位置する 5 つの行動に設定した．5 つの行動に限定することで，ゲーム画面の描画なしでの画像処理を行わない Ape-x の性能を調査した．

5.3　実験 II

実験 II は駒の移動範囲を上下左右に加えて各斜め方向の移動に拡張し行動選択数を増やした．実験 I と同じマップを使用し，行動選択数が増えた場合の Ape-x の性能を調査した．

6.　結果と考察

この章では実験 I と実験 II について結果と考察を述べる．

6.1　実験 I

マップ 1 においての DQN と Ape-x の結果が図 6.1 と図 6.4，マップ 2 においての結果が図 6.2 と図 6.5，マップ 3 においての結果が図 6.3 と図 6.6 である．

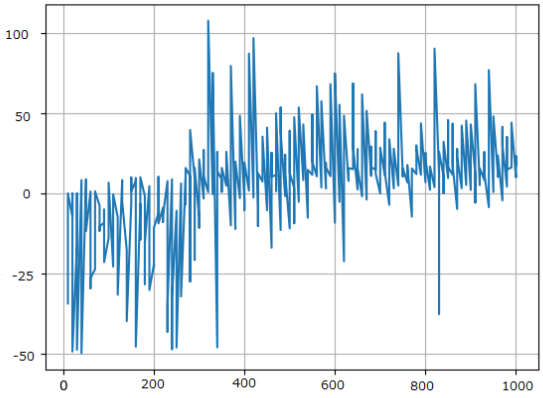

図 6.1　DQN　実験 I - 1

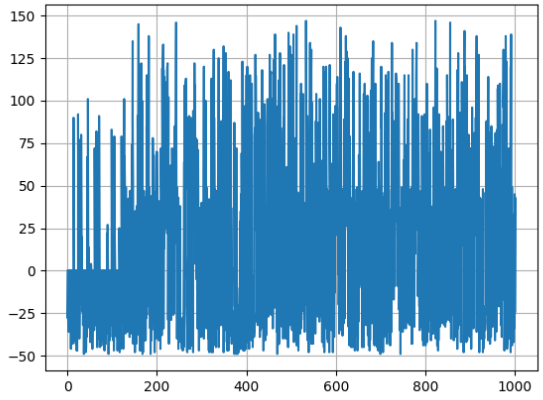

図 6.2　DQN　実験 I -2

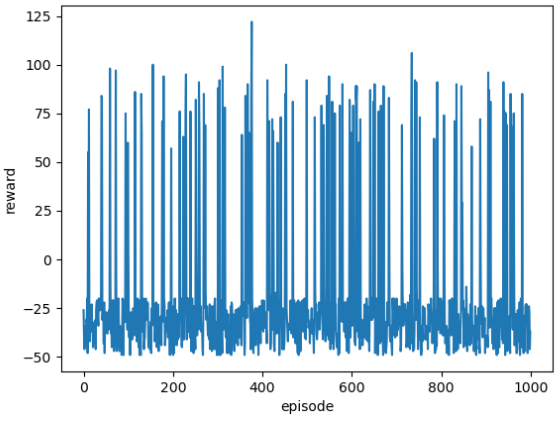

図 6.3　DQN　実験 I -3

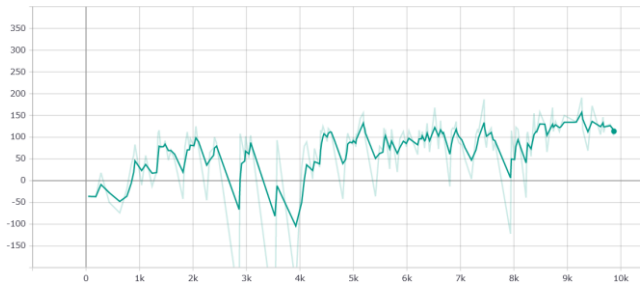

図 6.4　Ape-x　実験Ⅰ-1

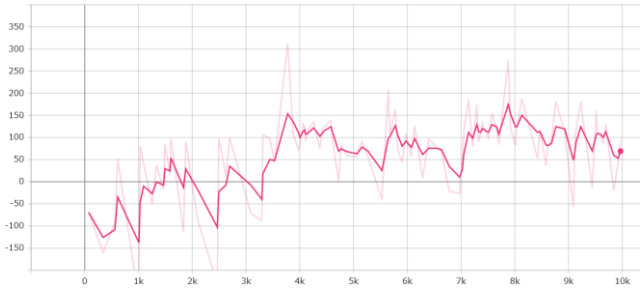

図 6.5　Ape-x　実験Ⅰ-2

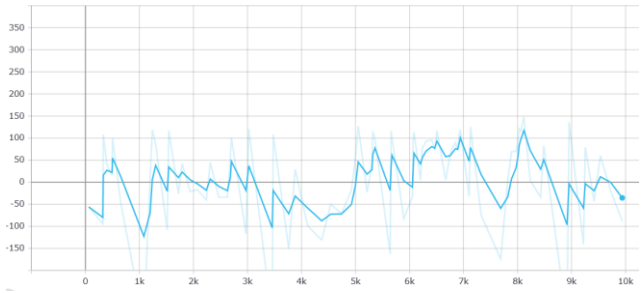

図 6.6　Ape-x　実験Ⅰ-3

いての最終獲得報酬とエピソード数のグラフである．

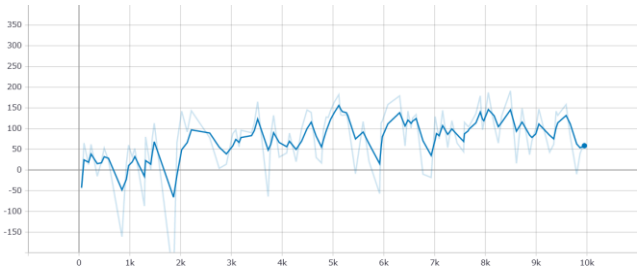

図 6.7　Ape-x　実験Ⅱ-1

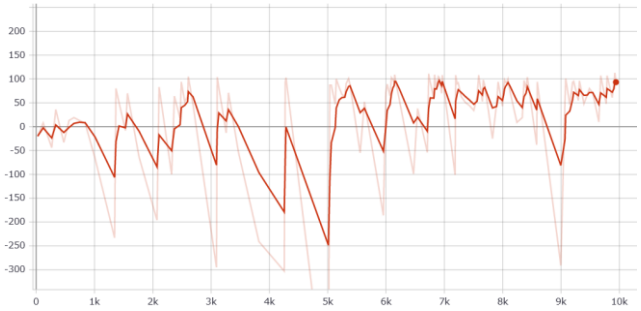

図 6.8　Ape-x　実験Ⅱ-2

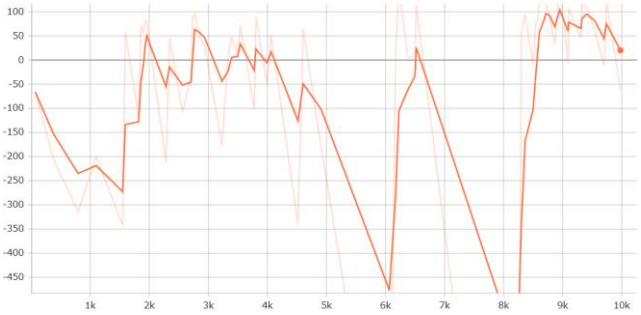

図 6.9　Ape-x　実験Ⅱ-3

DQN はマップ 1 ではエピソード数が 400 を超えると徐々に敵駒を倒す行動をするようになったことがわかる．マップ 2 ではマップ 1 と比べて報酬がマイナス，つまり不適切な行動を取ることが多くなったことがわかる．マップ 3 では他の 2 つのマップと比べてエピソード全体で不適切な行動をとることが多くなったことがわかる．マップ 2 とマップ 3 は移動できないマスを増やし，マップ 3 においては迷路状となっているのが原因であると考える．

Ape-x も DQN と同様にマップ 2，マップ 3 と徐々に最終的な獲得報酬が下がっていることがわかる．しかし，DQN と比べて全体を通して報酬が高い方へ収束していくことがわかる．よって Ape-x は DQN と比べて徐々にではあるが，より最適な行動をとることができる手法であると考える．

6.2　実験Ⅱ

実験Ⅱでは，移動範囲を上下左右に加えて各斜め方向の移動を増やし，探索空間を広げた場合の Ape-x の性能調査を行った．以下の図はそれぞれのマップにお

実験Ⅱでは斜め方向の移動を増やしたことで実験Ⅰと比べて変化が見られた．図 6.7 を見ると，図 6.1 と比べて報酬がマイナスであるエピソードが少なくなったことがわかる．これは斜め方向の移動が加わったことで移動できる範囲が増えたことで移動できないマスに対する行動を取りにくくなったことが原因であると考える．図 6.8 を見ると，マップ 2 ではマップ 1 と比べて移動できないマスが増えたことで実験Ⅰと同様報酬がマイナスを取るエピソードが増えたことがわかる．図 6.9 を見ると報酬がマイナスを取るエピソードがとても増えたことがわかる．これはマップ 3 が他のマップと比べて迷路状となっていることで移動できないマスが多いことが原因であると考える．実験Ⅱの結果をまとめると，行動選択が増えることで移動できるマスが多いマップにおいては高い報酬を取りやすくなったと言える．しかし，移動できないマスが多いマップに対しては報酬がマイナスの場合が多くみられ，マップそれぞれに対して適応しているとは言えない結果となった．

6.3 考察

自作のゲーム環境で Ape-x の実装に成功し，実験Ⅰと実験Ⅱを行った．結果としては，DQN と比べて移動できるマスが多いマップでは高い報酬をとる行動に収束していくが，移動できないマスが多いマップの場合不安定な行動をとることが多いことがわかった．移動できないマス，つまり障害物が多いマップにおける考察としてまとめる．マップ 2 は中央に障害物が存在しているマップである．2，3 ターン目には移動できないマスに対しての行動は少ないが，その後の行動においては障害物への対応が必要になってくる．図 6.2 と図 6.5，図 6.8 を見ると，DQN は 1000 エピソードまで障害物への対応に苦戦していることがわかる．一方で Ape-x は少ないエピソード数では障害物への行動が多かったが，障害物への対応を学習し中盤以降のエピソード数では高い報酬を得られる行動，つまり敵駒を倒すような行動選択を多くとることができたと考える．

マップ 3 は迷路状に障害物が存在しているマップである．序盤のターン数では左右にしか移動することしかできなく，一方通行状の通路を抜けてもまだ障害物が多いマップとなっており，移動できるマスへの行動が少ないと報酬がマイナスになることが多くなってくる．図 6.3 と図 6.6，図 6.9 を見ると，DQN はエピソード数が多くなっても最後まで障害物への対応ができなかった，あるいはできた回数が少ない結果となったことがわかる．Ape-x もマップ 1 やマップ 2 ほど高い報酬を取るような結果が見られなかった．報酬が DQN よりも高くなることが多いが，報酬が 100 前後のラインに到達していることが少ない．この結果から序盤は障害物への学習ができていなく左右の行動のみで終了したり障害物への行動をずっととり続けてしまっていることがわかる．中盤終盤のエピソード数でも相手駒へ到達し倒すことが少なく，相手駒を倒す経験が非常に少なかったことが，報酬が 100 前後のラインに到達しなかった原因であると考える．この問題に対して，終盤の盤面のみを先に学習させる，つまり敵を倒すような行動を学習させたのちにマップ 3 における初期位置から学習させるとより報酬が高くなるような行動をとることができたと考えられる．今回の実験で，終盤の盤面の学習を先に行うことで，自作のゲーム環境においてより性能の高い Ape-x が見られる可能性が示されたと考える．

また，敵の行動がランダムであることで固定された盤面があまり存在しないことが学習の妨げとなることが報酬の獲得具合が不安定な原因であると考えられる．将棋やチェスでは定石と呼ばれる一定の行動，盤面が存在し AI 側もその状況に対応して学習を行う．今回の環境では，マップサイズが小さいことからランダム行動であるが同じ盤面が何度も存在することもある．しかし，より広いマップサイズに拡張した場合を考えるとランダム行動の場合だと定石が存在しないため今回の実験より低い報酬を取ることが考えられる．そのためある程度一定の動きを見せる標準なアルゴリズムを基にした対戦エージェントの実装も考慮する必要がある．また今回自作のゲーム環境では画像描画を行わなかったことで画像処理を行わない適用への拡張がうまくいかなかったことが考えられる．R2D2 も LSTM を使用することから，画像処理の工程が必要であると考える．本研究の環境では画像描画の工程が存在しないため，Ape-x の改良手法である R2D2 でも Ape-x と比べて格段に高い結果を残すとはいえないと考える．また，Ape-x 本文中では Actor の数が 360 ととても多く，本研究では Actor を 2 つに限定しているため予想していた性能より下回ったと考える．Actor の数を減らしてもある程度の性能を出すことはできるが，Actor の数が多いほどより様々な経験，状況を生み出すことができる．そのため学習の幅が狭まったのが，予想していた性能より下回った原因の一つであると考える．

7. 結論

本研究で使用した自作のゲーム環境について，この環境は TUBSTAP を参考に Python 用に開発した．このためテストプレイヤが存在しなく，理想の評価値のラインを提示することができなかった．人間であればどの程度のターン数がかかり，どの程度の報酬を取得できるかという情報が存在しなかった．つまりどの程度の行動選択を取ることができれば一定以上の性能であるという指標が存在しなかった．今後の展望としては，この自作のゲーム環境において人間にテストプレイをしてもらい，人間のプレイヤによる一定の評価値のラインを取得し Ape-x や R2D2 との比較をしたいと考える．また本環境で R2D2 を実装する場合，画像処理の工程が必要になるため環境の方でも画像描画ができるような拡張の必要が考えられる．画像描画の拡張をした場合の画像処理を行う Ape-x の性能も調査したいと考える．

また本研究において TUBSTAP 環境に実装しなかったが，実装に成功し実験した場合，木村の研究と比較することを検討していた．実装した場合 Ape-x では合法手出力の割合やエピソード成功率は，DQN 手法と Profit-Sharing を組み合わせた木村の手法の実験結果より 20%以上の向上が見込めると考える．一方で R2D2 は Ape-x を改良した手法であり，時系列的特徴

を捉えることが可能になりその性能を大きく向上した手法であるため，更に合法手出力の割合とエピソード成功率が向上したと考える．

また，TUBSTAP 環境に実装した場合，対戦相手のアルゴリズムとして TUBSTAP に標準で搭載されているモンテカルロ法アルゴリズムや Military-Stattf-system アルゴリズムを予定していた．これらの手法は探索の深さによって性能が変わるものであると考えており，Ape-x と R2D2 の手法でもアクターの数による制限やリプレイメモリからなる様々な盤面による経験再生の不足により勝率が低くなると考える．一方で，囲碁や将棋などのモンテカルロ法のみのアルゴリズムでは人間のトッププレイヤーに勝利は難しいが，深層学習を組み合わせることで AlphaZero などはその性能を大幅に向上させている．このことから Ape-x と R2D2 の両手法の勝率がモンテカルロ法アルゴリズムや Military-Staff-system アルゴリズムより高くなる可能性は十分にあったと考える．

8. 今後の展望

今後の展望としては，自作のゲーム環境では TUBSTAP 環境の探索空間と比べて小さい空間になっていることから，駒を増やしたり移動できる範囲を増やすことでより広い探索空間における Ape-x と R2D2 の性能の調査を行いたいと考える．また，TUBSTAP 環境に Ape-x と R2D2 を実装しその性能の調査も行いたいと考える．TUBSTAP では大会が開かれることもあり，Ape-x と R2D2 を実装したエージェントで他の手法が実装されたエージェントと対戦させることでどのような結果が生まれるのかも調査したいと考える．

参考文献

[1] D Silver，J Schrittwieser，K Simonyan，I Antonoglou，A Huang, A Guez, T Hubert, L Baker, M Lai, A Bolton, Y Chen, T Lillicrap, Fan Hui, L Sifre, G van den Driessche, T Graepel, D Hassabis，“Mastering the game of Go without human knowledge，” Nature，Vol．550，pp．354-359(2016).

[2] 三宅 陽一郎，"ディジタルゲームにおける人工知能技術の応用"，人工知能学会誌，Vol. 23，No. 1，pp. 44-51(2008).

[3] 三宅 陽一郎，"ディジタルゲームにおける人工知能技術の応用の現在"，人工知能学会誌，Vol．30，No．1，pp．5-64(2015).

[4] 三宅 陽一郎, "戦略ゲーム AI 解体新書 -ストラテジー&シミュレーションゲームから学ぶ最先端アルゴリズム-"，翔泳社(2021).

[5] "ターン制戦略ゲーム 学術用基盤プロジェクト TUBSTAP"，http://www.jaist.ac.jp/is/labs/ikeda-lab/tbs/.

[6] "Advance Wars: Days of Ruin"，http://www.nintendo.com/games/detail/nLeg9iJkPgq3fWBcqtpDNWUJ4IvmaQBY.

[7] 藤木 翼，村山 公志朗，池田 心，"ターン制ストラテジーのための状態評価型深さ限定モンテカルロ法"，エンターテイメントと認知科学研究ステーション 第 8 回 シンポジウム，2014-3(2014).

[8] 武藤 孝輔，西野 順二，"ターン制戦略ゲームにおける UCT とファジィ評価の適用"，第 31 回ファジィシステムシンポジウム，Vol．2，No．4， pp．226-229(2015).

[9] 木村 富宏："ターン制戦略ゲームへの深層学習の適用"、情報処理学会 第 41 回ゲーム情報学研究会、Vol. 2019-GI-41, No．5, pp．1-8 (2019).

[10] Dan Horgan，John Quan，David Budden，Gabriel Barth-Maron，Matteo Hessel，Hado van Hasselt，David Silver，”Distributed Prioritized Experience Replay”，arXiv preprint arXiv:1803.00933,2018.

[11] Steve Kapturowski，Georg Ostrovski，John Quan，Remi Munos，Qill Dabney，”Recurrent Experience Replay in Distributed Reinforcement Learning”， International Conference on Learning Representations, 2018.