Vol.2021-GI-45 No.9
2021/3/5

強化学習による連鎖型落ち物パズルゲームの研究

杉江矢[1]　橋本剛[1]

概要：近年，ゲーム AI の汎用機械学習の研究が注目されているが，成功しているゲームはブロック崩しなど単純なものがほとんどである．テトリスは人間にとっては単純に感じるが，汎用機械学習の題材としては難しいことが報告されている．DQN のような強化学習がうまくいく条件として，ランダムな操作である程度の報酬が得られる必要がある．そして，連鎖型落ち物パズルゲームと呼ばれるジャンルはこの条件を満たすと考えた．本稿では，DQN を用いて連鎖型落ち物パズルゲームであるパネルでポン・ぷよぷよ・コラムスにおいて大連鎖・高得点を獲得する AI を作成することを目的とする．結果として，ニューロエボリューションと比較して多くの連鎖を達成し高い汎用性があることがわかった．

キーワード：AI，機械学習，パズル，パネルでポン，ぷよぷよ，コラムス

A Study on Falling Block Puzzle Game by Reinforcement Learning

NAO SUGIE[†1]　TSUYOSHI HASHIMOTO[†1]

Abstract: In recent years, research on general-purpose machine learning of game AI has attracted attention, but successful games are as simple as Breakout. Although Tetris feels simple to humans, it is reported that it is difficult as a subject of general-purpose machine learning. A condition for reinforcement learning such as DQN to work is that some reward must be obtained at random operations. With this background, the game genre called chained falling block puzzle game met this requirement. The aim of this paper is to create an AI that performs many chains by machine learning using DQN, applying Tetris Attack and Puyopuyo and Columns as a chained falling block puzzle game. As a result, this method achieved the larger number of chains and more versatility than neuro-evolution.

Keywords: AI, machine learning, puzzle, Tetris Attack, Puyopuyo, Columns

1. はじめに

近年，ゲーム AI の汎用学習の研究が注目されている．汎用学習の中でも，DeepMind 社の DQN(Deep Q-Network) を用いたAtariゲームの強化学習[1]は有名である．これは，ブロック崩しやピンボール等 49 種類の簡単なゲームの内，半数以上で人間よりも高いスコアを獲得することに成功している．ほかにも，弾幕シューティングで弾を回避する[2]，ターン制 RPG においてステージを自動生成す[3]，人狼で適切な対象選択を行う[4]などさまざまな題材で研究が行われている．しかし，汎用学習が成功しているゲームは単純なものが多い．パズルゲームのテトリスは人間にとっては単純に感じるが，青木の汎用学習を用いた研究[5]では約 8 ラインしか消去できておらず，アルゴリズムベースの AI や人間によるプレイと比較すると遠く及ばない結果である．そこで本稿では，連鎖型落ち物パズルゲームと呼ばれるジャンルに注目した．

落ち物パズルゲームは，「ブロックがフィールドの最下段か他のブロックの上に落下するとそこで位置が固定される」・「ある条件を満たすとブロックが消滅し得点が入る」等の特徴を持つパズルゲームの総称である．また，パネルでポン・ぷよぷよ・コラムスのようなゲームはブロックが消滅する条件を連続で行うことで発生する「連鎖」という要素を持つため連鎖型パズルゲームと呼ばれることもある．さらに，同色のブロックを 3 つ以上並べることがブロック消滅条件となっているパネルでポン・コラムスはマッチ 3 パズルゲームと呼ばれることもある．

パズルゲームは「パズル問題を作成する」，「パズル問題を解く」という 2 つの視点から研究の題材として取り扱われることが多い．ぷよぷよを例にすると，逆向き生成法を利用してパズル問題を作成する研究[6]や，Nested Monte Carlo Search を用いてパズル問題を解く研究[7]等が行われている．このように，有名なパズルゲームはいくつかの研究が行われているが，強化学習を利用した研究や，パネルでポンのような比較的知名度が低いゲームの研究は少ない．パネルでポンとぷよぷよを対象としてニューロエボリューション（NE）を用いた先行研究では特定の条件で連鎖を発見できたが，より汎用的な条件ではうまく連鎖が学習できないという課題点があった[8]．

本研究では連鎖型落ち物パズルゲームの評価指標となる「連鎖」と「スコア」に注目し，大連鎖・高得点を獲得する AI の作成を目的とする．研究対象として，先行研究で扱ったパネルでポンとぷよぷよに加え，コラムスを題材とする．そして，機械学習の手法として DQN を用いる．また，各ゲームの学習結果を比較し，この手法が連鎖型落ち物パズルゲームというジャンルにおいて汎用的であるか

1 松江工業高等専門学校
National Institute of Technology, Matsue College.

考察する．

本稿では，まず 2 章で研究対象である 3 つのゲームの概要を説明する．次に，3 章で学習手法である DQN を解説し，4 章でその実装について説明する．次に，学習結果について 5 章で述べる．最後に 6 章で本研究のまとめを記す．

2. ゲームの概要

2.1 パネルでポン

パネルでポンは，1995 年に任天堂・インテリジェントシステムズが開発したスーパーファミコン用のゲームである（図 1）．その後は，ゲームボーイやニンテンドーDS 等のハードで発売されている他，海外にも“Puzzle League”や“Tetris Attack”の名称で展開している．ルールは盤面の下部からせり上がってくるパネルを消していく．パネルが盤面の最上部まで達してしまうとゲームオーバーになる．パネルが消滅したとき，そのパネルより上にあるパネルは下に落ちてくる．そして，落ちてきたパネルが再び消滅する条件を満たしたとき，これを連鎖と呼ぶ．図 2 に 2 連鎖が発生した例を示す．

図 1　パネルでポン
Figure 1　Tetris Attack.

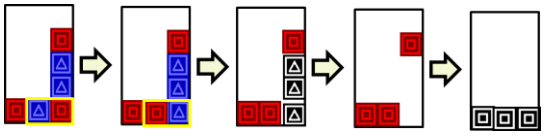

図 2　パネルでポンの 2 連鎖の例
Figure 2　Two chains of Tetris Attack.

2.2 ぷよぷよ

ぷよぷよは，1991 年にコンパイルが開発したファミコン・MSX2 用のゲームである（図 3）．2 個 1 セットで落ちてくるぷよを積んでいき，同じ色のぷよを縦横 4 つ以上繋げて消すことで得点が入る．効率よく連鎖を発生させるための定石が数多く存在し，定石の自動生成[9]や人間のプレイを模倣する[10]AI の研究も行われている．図 4 に 2 連鎖が発生した例を示す．

図 3　ぷよぷよ
Figure 3　Puyopuyo.

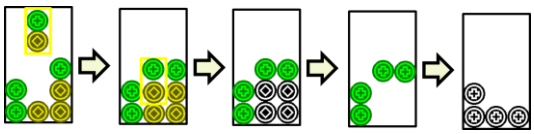

図 4　ぷよぷよの 2 連鎖の例
Figure 4　Two chains of Puyopuyo.

2.3 コラムス

コラムスは，1990 年にセガ・エンタープライゼスが開発したアーケード用のゲームである（図 5）．落ち物パズルに初めて連鎖要素を取り入れたゲームであり，さまざまな機種に移植されている．3 個 1 セットで落ちてくる宝石を積んでいき，同じ色の宝石を縦か横か斜めに 3 つ以上並べて消すことで得点が入る．図 6 に 2 連鎖が発生した例を示す．

図 5　コラムス
Figure 5　Columns.

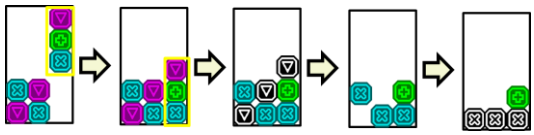

図 6　コラムスの 2 連鎖の例
Figure 6　Two chains of Columns.

3. 学習手法

DQN は DeepMind 社が 2013 年に発表した深層強化学習の手法であり，エージェントが試行錯誤を繰り返すことでゲームのルールを教えなくても状態に応じて報酬を最大化する行動を学習することができる．具体的には，以下の 3 つのステップを繰り返すことでエージェントの学習が行われる．

1．Q-network に現在の状態 s を入力し，次に行う行動 a と行動価値関数 Q(s,a)を得る

2．行動 a に基づいてゲームを進め，報酬 r と次状態 s’を得る

3．s, a, s’, r, Q(s,a)を保存し，Q-network の重みを更新する

この手法は，最初はランダムな操作で学習を進めていくため，その段階でもある程度の報酬が得られるようになっている必要がある．本研究で扱うゲームは，ランダムな操作でもブロックの消去や連鎖が発生するため比較的学習しやすいと考えられる．

4. 実験内容

実験環境として，プログラミング言語に Python，学習ライブラリに Keras を使用する．Keras は Python でディープニューラルネットワークを扱うのに適しているオープンソースライブラリである．このライブラリが持つ SequentialAPI を利用することで簡潔なコードでモデルの構築が可能になり，レイヤーや活性化関数もカスタマイズすることができる．

実験の流れとして，ゲーム画面の情報を入力とし，ある特定のコマンドを出力するエージェントを DQN により構築する．そして，プレイを進めて終了条件（ブロックが画面最上部まで積み上がりゲームオーバーになる・100 ターン経過する）を満たした場合に 1 エピソードを終了し，次エピソードに移る（図 7）．

本研究では，盤面の各マス情報を状態 s としてエージェントに与え，ゲームルールに則った操作を行動 a として出力する．そして，状態 s において行動 a を行った後の盤面の各マス情報を次状態 s’とする．なお，各ゲームの状態空間と行動空間のサイズは表 1 のようになる．そして，式(1)に示すように連鎖数が強調されるスコアを報酬 r として設定する．また，表 2 に学習パラメータを示す．

また，汎用性を見るために固定条件と変動条件の 2 種類の条件で学習を行った．パネルでポンの固定条件とは，すべてのエピソードで同じブロックの並びの盤面を使用し，変動条件は 1 エピソード毎に盤面のブロックの並びをランダムに入れ替えることを意味している．また，ぷよぷよとコラムスの固定条件とは，疑似乱数によりすべてのエピソードで配ブロック列を同じ順番で落下させ，変動条件は 1 エピソード毎にランダムな配ブロック列を落下させることを意味している．

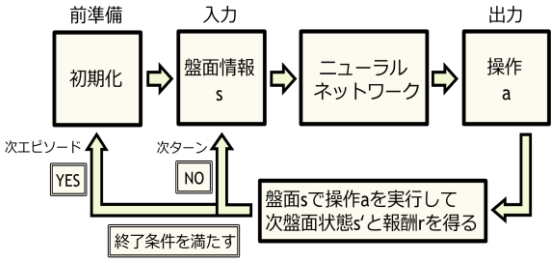

図 7　学習の流れ
Figure 7　Learning process.

表 1　各ゲームの空間サイズ
Table 1　State and action size for each game.

	状態空間サイズ	行動空間サイズ
パネルでポン	72	60
ぷよぷよ	80	22
コラムス	75	18

スコア＝消去したブロック数×連鎖数 ^{2}　　(1)

表 2　学習パラメータ
Table 2　Learning parameters.

エピソード数	10000
割引率	0.99
活性化関数	ReLu
optimizer	Adam
バッチサイズ	64
中間層	3
中間ノード数	各 128

5. 実験結果

5.1 パネルでポン

パネルでポンについて学習させた際のスコア結果を図 8，連鎖数結果を図 9 に示す．なお，図 8 では，青いグラフがスコアの推移を表しており（スケールは左側の縦軸），オレンジのグラフは青いグラフを移動平均化（区間 600）したものである（スケールは右側の縦軸）．図 8 から，1000 エピソード付近で学習が収束しており，図 9 から，最大で 6 連鎖を達成していることがわかる．今回の学習の中で最も連鎖数の多かった 6 連鎖の様子を図 10 に示す．

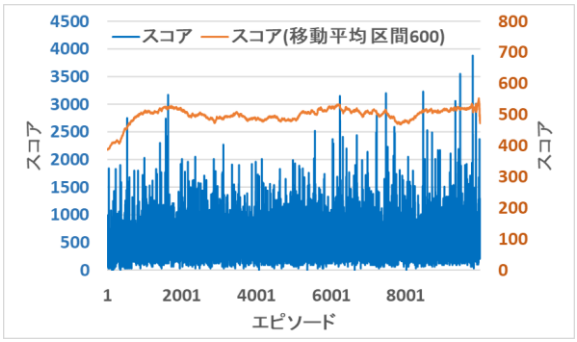

図 8　パネルでポン スコア結果

Figure 8　Results of Tetris Attack scores.

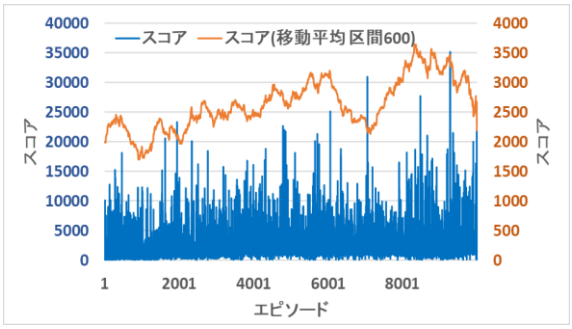

図 11　ぷよぷよ スコア結果

Figure 11　Results of Puyopuyo scores.

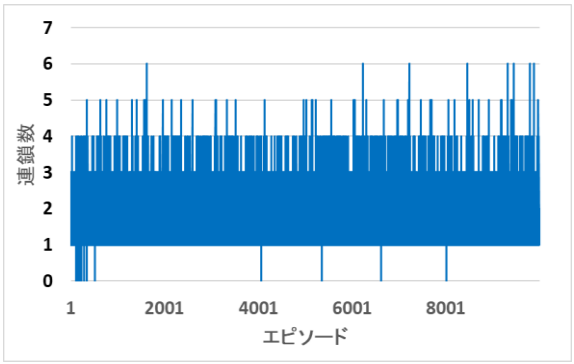

図 9　パネルでポン 連鎖数結果

Figure 9　Results of Tetris Attack chain counts.

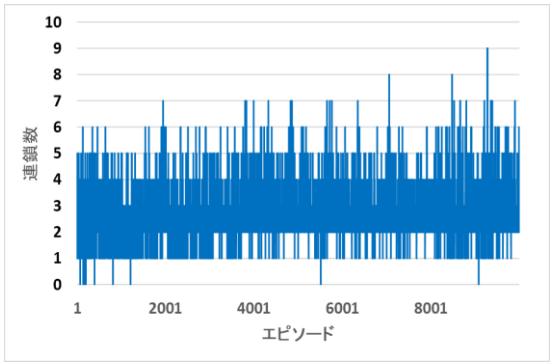

図 12　ぷよぷよ 連鎖数結果

Figure 12　Results of Puyopuyo chain counts.

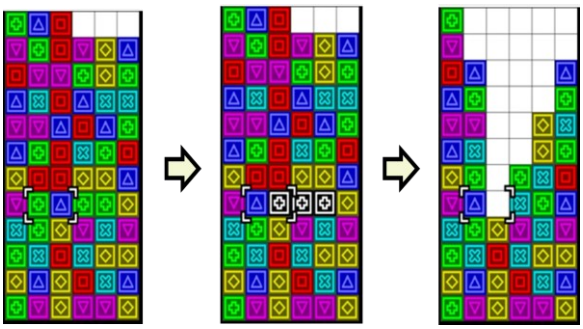

図 10　パネルでポン 6 連鎖の流れ

Figure 10　Tetris Attack 6-chain flow.

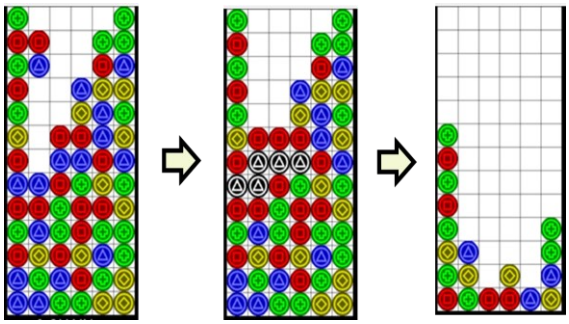

図 13　ぷよぷよ 9 連鎖の流れ

Figure 13　Puyopuyo 9-chain flow.

5.2　ぷよぷよ

ぷよぷよについて学習させた際のスコア結果を図 11，連鎖数結果を図 12 に示す．なお，図 11 では，青いグラフがスコアの推移を表しており（スケールは左側の縦軸），オレンジのグラフは青いグラフを移動平均化（区間 600）したものである（スケールは右側の縦軸）．図 11 から，徐々にスコアが増え続けており，図 12 から，最大で 9 連鎖を達成していることがわかる．今回の学習の中で最も連鎖数の多かった 9 連鎖の様子を図 13 に示す．

5.3　コラムス

コラムスについて学習させた際のスコア結果を図 14，連鎖数結果を図 15 に示す．なお，図 14 では，青いグラフがスコアの推移を表しており（スケールは左側の縦軸），オレンジのグラフは青いグラフを移動平均化（区間 600）したものである（スケールは右側の縦軸）．図 14 から，徐々にスコアが増え続けており，図 15 から，最大で 8 連鎖を達成していることがわかる．今回の学習の中で最も連鎖数の多かった 8 連鎖の様子を図 16 に示す．

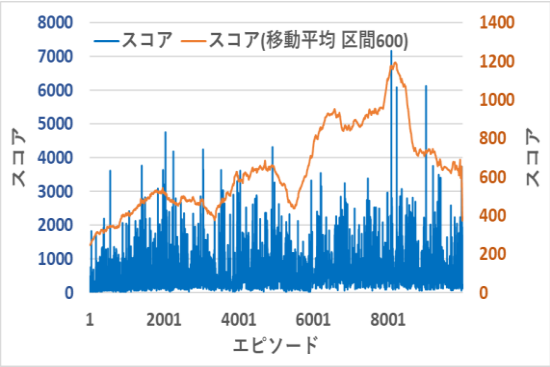

図 14　コラムス スコア結果

Figure 14　Results of Columns scores.

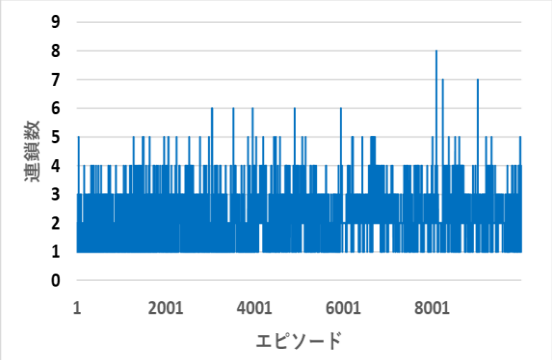

図 15　コラムス 連鎖数結果

Figure 15　Results of Columns chain counts.

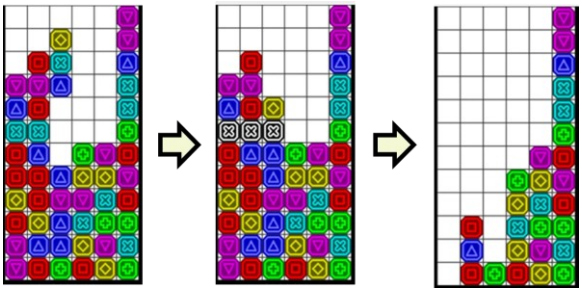

図 16　コラムス 8 連鎖の流れ

Figure 16　Columns 8-chain flow.

5.4 比較・考察

結果の比較として，本研究で使用した DQN，先行研究として行ったニューロエボリューション（NE と表記する），機械学習を用いないランダム操作（R と表記する）の 3 種の手法の連鎖数結果を表 3 に示す．なお，コラムスについては先行研究（NE）の時点では対象としていなかったため，今回追加で実験を行った．NE は，ニューラルネットワークに進化的アルゴリズムを組み合わせることによって多層パーセプトロン[11]の局所解に陥りやすいという問題点[12]を解決する手法であり，スーパーマリオで裏技を発見させるなどの研究が行われている[13]．表 3 から，DQN はいずれのゲームにおいてもランダム操作の連鎖数を上回っていることがわかる．また，NE と比較すると固定条件は同程度の連鎖数だが，変動条件ではいずれも上回っていることがわかる．学習時間においては NE は約 2 時間であるのに対し DQN は約 12 時間かかっているが，NE が対応できなかった変動条件についても連鎖を学習でき，汎用性の高さが確認できた．

パネルでポンの連鎖数が 6 連鎖と最も低かったことについては 4 章の表 1 で示したように行動空間のサイズが大きすぎてニューラルネットワークの計算が複雑化したのではないかと考えられる．パネルでポンの，隣接するブロック同士ならどこでも入れ替えられるというルールは人間にとっては比較的理解しやすい特徴であったが，逆に機械には選択肢が多過ぎて学習することが難しくなってしまったと考えられる．また，コラムスの方がぷよぷよよりも連鎖数が若干少なかったことに関しては，ぷよぷよは上級者によるプレイで 19 連鎖が観測されているのに対し，コラムスは 13 連鎖と低いため，学習の成否ではなく単にゲームシステム上の連鎖の発生しやすさに相関していると考えられる．

表 3　各手法による連鎖数結果

Table 3　Results of chain counts by each method.

	パネルでポン		ぷよぷよ		コラムス	
	固定	変動	固定	変動	固定	変動
DQN	6	6	9	9	8	7
NE	6	5	9	6	7	5
R (最大値)	5		6		5	
R (平均値)	2.6		3.1		2.3	

6. おわりに

汎用的な機械学習手法である DQN を用いて 3 種の連鎖型落ち物パズルゲームで学習を行った．その結果，固定条件だけでなく NE が対応できなかった変動条件においても連鎖を学習させることができた．この実験で得た結果を利用することで，「N 手で M 連鎖をせよ」のような問題の自動作成などに活かすことができると考えられる．また，本研究では一人用のモードのみを研究対象として扱ったが，今後は対戦プレイのような複雑かつ実践的な環境においても連鎖を行えるように，パラメータやモデルの調整などの改良について検討したい．

謝辞　本研究は JSPS 科研費 JP17K00514 の助成を受けたものです．

参考文献

[1] Volodymyr Mnih, et al.: Playing Atari with Deep Reinforcement Learning, Deep Learning Workshop NIPS 2013, pp. 1-9, 2013
[2] 野村直也，橋本剛：視覚的顕著性モデルを用いた汎用的機械学習法，ゲームプログラミングワークショップ 2018 論文集，pp. 23-29，2018
[3] ナムサンギュ，池田心：強化学習を用いたターン制 RPG のステージ自動生成，ゲームプログラミングワークショップ 2018 論文集，pp. 160-167，2018
[4] 王天鶴，金子知適：人狼エージェントにおける深層 Q ネットワークの応用，ゲームプログラミングワークショップ 2018 論文集，pp. 16-22，2018

[5] 青木勢馬，橋本剛：テトリスを題材にしたスケールダウンを利用した学習手法の開発，ゲームプログラミングワークショップ 2017 論文集，pp. 99-103，2017
[6] 高橋竜太郎，池田心：連鎖構成力向上のためのぷよぷよの問題作成，研究報告ゲーム情報学，pp. 1-7，2018
[7] 齋藤晃介，三輪誠，鶴岡慶雅，近山隆：Nested Monte Carlo Search のぷよぷよへの適用，ゲームプログラミングワークショップ 2013 論文集，pp. 134-137，2013
[8] 杉江矢，橋本剛：ニューロエボリューションを用いた連鎖型パズルゲーム AI の研究, 研究報告ゲーム情報学, pp. 1-8, 2019
[9] 富沢大介，池田心，シモンビエノ：落下型パズルゲームの定石形配置法とぷよぷよへの適用，情報処理学会論文誌，pp. 2560-2570，2012
[10] 隅山淳一朗，橋山智訓，田野俊一：ぷよぷよにおける人間のプレイデータの特徴量抽出，ファジィシステムシンポジウム講演論文集，pp. 1–4，2015
[11] Christopher M.Bishop: Neural Networks for Pattern Recognition, Oxford University Press, 1995
[12] Geoffrey E.Hinton, et al.: Reducing the Dimensionality of Data with Neural Networks, Science 313, 2006
[13] 高田亮介，橋本剛：無限 1UP を題材としたアクションゲームの裏技を発見する自己学習手法の提案，研究報告ゲーム情報学，pp.1-7，2018