5B-05

ルービックキューブ求解への深層学習適用検討

水澤　悟†　　清　雄一†

電気通信大学大学院†

1　はじめに

ルービックキューブの状態数は約 4.3*10^19 通りであり，メモリに解をすべて記録するのが難しいパズルである．ルービックキューブの求解に探索アルゴリズムである Iterative Deepening A*(IDA*)と深層学習(ニューラルネット)を組み合わせて利用した例はあるが [1]，求解手数が IDA*とヒューリスティクスを用いた既存手法 [2] にくらべ長く課題である．本研究ではこの課題の解決を目指して，4 通りの手法の提案を行った．しかし解手数が深い場合問題を解くことができなかったため，課題を分析した．

2　提案手法

表 1 に本研究で検討した手法の一覧を示す．

表 1 検討手法一覧

	手法	環境の形式	求解モデルの形式	求解モデルが求めるもの	世界モデルと求解モデル学習タイミング
2.1	操作の世界モデル化	操作世界モデル	Dense+argmin	残り手数	個別
2.2	RNNの世界モデル化	RNN世界モデル	Dense+argmax	状態に応じた最善手	同時
2.3	オフライン学習の適用	オフラインシミュレータ	強化学習	期待値を最大化する操作	-
2.4	IDA*+下界枝刈り関数	オンラインシミュレータ	Dense	残り手数(下界)	-

2.1. 操作の世界モデル化

世界モデルは環境を深層学習のモデルに学習させたものである．一般に環境を学習させると微分が可能になり，解を導くモデル(求解モデル)の学習が容易になる．ルービックキューブの場合でも同様に容易になると考えた．

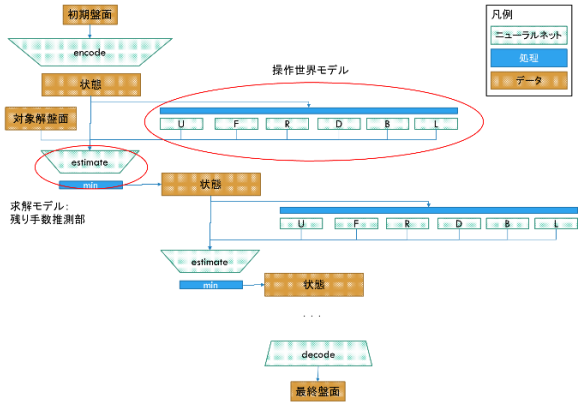

図 1 操作世界モデルを利用したモデル

A Study of Deep Learning in Rubik's Cube Solving
†The University of Electro-Communications

モデルは図 1 に示す通りである．encoder はルービックキューブの盤面を特徴量にエンコードする．U,F,R,D,B,L はルービックキューに対する操作のニューラルネットのモデルである．decoder は操作により出力された特徴量をルービックキューブの盤面にデコードする．

モデルは初期盤面を入力されると盤面をエンコードし，エンコードされた盤面に対して学習された操作 U,F,R,D,B,L を適用し，estimator は対象解盤面と操作結果を入力され，それぞれの操作結果に対して残り手数の予想を出力する．それらの手数から最小値となる操作を選択し，その結果を次の入力とする．

2.2. RNN の世界モデル化

2.1 のモデルでは世界モデルと求解モデルを個別に学習させたが，図 2 に示す本モデルでは世界モデルを GRU(RNN)にし求解モデルと同時に学習させた．

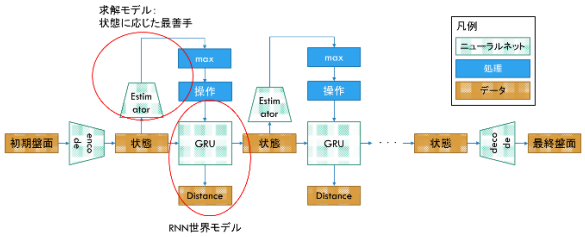

図 2 RNN 世界モデルを利用したモデル

2.3. オフライン学習を用いた方法

本モデルではオフライン学習を適用した．オフライン学習は車両の走行などのように安全面の観点から実環境で評価を行いづらい場合でも強化学習を行う方法であり，評価関数に未観測なデータに対する評価が適切になる工夫がしてある．ルービックキューブは状態数が多く，学習で予想した状態から離れた状態が評価時に入力されることも多いと考え効果があると考えた．

2.4. IDA*+下界枝刈り関数

本モデルでは IDA*の残り手数の下界を用いた枝刈り関数をニューラルネットに置き換えた．

IDA*の操作の選択に優先度をつけるのが，先行手法 [1]であるが，この手法は，操作の選択を間違った場合，求解できない，最短手数がもとまらないといった課題がある．既存手法 [2]では簡

易化したルービックキューブにおいて，解までの手数(下界手数)をあらかじめメモリに収まるサイズで全盤面に対して計算し，これを枝刈りに使用している．

ただ，簡易化したルービックキューブはヒューリスティックに決められており，この判定部分をニューラルネットに置き換えることでより効率的な下界手数を提示できると考えた．

3　評価結果

学習済みモデルでの評価結果を表 2 に示す．

表 2 モデル学習条件と評価結果

	学習用データ	100%解ける最大手数	解ける最大手数
操作の世界モデル化	入力:対象解盤面，盤面 出力:残り手数	0	0
RNNの世界モデル化	入力:初期盤面 出力:操作列，Distance列，最終盤面	2	5
オフライン学習の適用	observations: 現状の盤面 actions:操作 rewards: (-1(操作するごと),100(解けた状態)) + (解けた状態とのMSE差分) terminals:0,1	4	11
IDA*+下界枝刈り関数	入力:状態 出力:残り手数(下界)	3	5

2.1 操作の世界モデル化の場合，操作の学習自体は操作ごとに accracy90%を達成したが，求解モデルの学習自体は学習が進まず評価が不可能だった．2.2 RNN を用いたモデルの場合，100%解ける最大手数が 2，解けることがある手数の最大数が 5 となった．2.3 オフライン学習の適用の場合，100%解ける最大手数が 4,解けることがある手数の最大数が 11 となった．2.4IDA*+枝刈り関数の場合，100%解ける最大手数が 4,解けることがある手数の最大数が 11 となった．

4　考察

実験では解手数が浅い場合は十分な性能を示したが，解手数が深くなると解くことができなかった．これは学習データがルービックキューブの状態数に対してすくな過ぎることが理由として考えられる．

図 3 に示すのはルービックキューブの解手数と状態数の関係である [3]．横軸は解までの手数，縦軸は状態数であり対数スケールである．

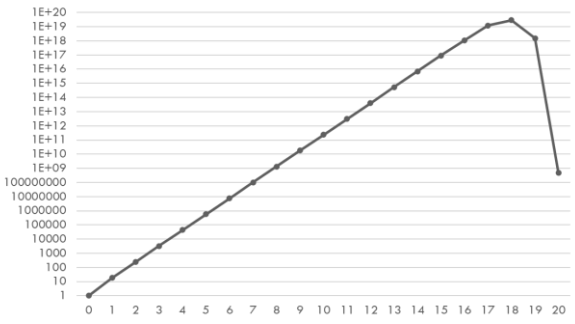

図 3 解までの手数と状態数

学習においてはすべての手数盤面について均等に学習ができるように，すべての手数盤面が学習データに含まれるようにしたが，各手数と総状態数の関係は対数スケールなため，手数が大きい場合割合としてはとても少なくなる．

またすべての盤面に対して手数を保存したデータを作成しようとしても，既存手法を用いた場合.約 1.2e+16 時間がかかり，データとしても約 800 エクサバイトが必要になり現実的でない．

このため強化学習などでデータを作成しつつ同時に学習をする手法が有望と考えられるが，今回の実験で作成した 2.4 の環境では 1 秒間に 1000 状態しか学習できず，すべての状態に対して学習できたといえる状態になるための学習時間が膨大になる．

一方 2.3 のようにオフライン学習の手法を用いた場合，提案手法の中では一番結果が良かった．これは未知の盤面に対する評価がうまく学習できているといえる．

これらを勘当すると解手数が大きい場合にも学習ができるモデルを作成するためには 1.未知盤面への評価がうまくいくような評価関数の工夫，2.高速なシミュレーション環境が必要といえる．

5　まとめ

本論文ではルービックキューブの求解を深層学習のモデルに実施させる方法を検討した．操作の世界モデルを用いて次の手を予測させる方法，RNN を用い世界モデルと求解モデルを同時に学習させる方法，オフライン学習の手法を利用する方法，IDA*の下海枝刈り関数を学習させる方法を提案し比較した．それらの手法は解手数が浅い場合は十分な性能を示した．しかし解手数が深い場合問題を解くことができなかった．今後，評価関数の工夫により未知の盤面への評価性能の向上と，高速なシミュレーション環境の作成のため RNN を利用した簡略化などを検討していく．

参考文献

[1] F. Agostinelli, S. McAleer, A. Shmakov , P. Baldi, “Solving the Rubik’s Cube with Deep Reinforcement Learning and Search,” Nature Machine Intelligence, Volume 1, 2019.

[2] H. Hayakawa , H. Murao, “Optimal Rubik’s Cube Solver on GPU,” GPU Technology Conference, 2013.

[3] T. Rokicki, H. Kociemba, M. Davidson , J. Dethridge, “God’s Number is 20,” 2010. [オンライン]. Available: http://cube20.org/.