

パズルの局面の機械学習のためのデータ生成

Data generation for machine learning of puzzle positions

神保 秀司†
Shuji JIMBO

玉井 慎悟†
Shingo TAMAI

和氣 卓史†
Takufumi WAKE

1 はじめに

本研究で扱うパズルは、一人遊びのカードゲームのフリーセルである。フリーセルは、すべてのカードを表にした状態でゲームを進める完全情報ゲームである。任意のフリーセルの局面からの最短手順の解を求めるアルゴリズムの設計について提案する。本研究は、ディープラーニングを応用した少ない探索量で最短手順に近い解を求めるアルゴリズムの開発の補助手段に位置付けている。ディープラーニングを応用したフリーセルソルバーを開発する主な理由の一つは、現実世界の計算機を使った知的処理の効率化への応用である。フリーセルの局面の求解を大量に実行する場合に GPU のような質は悪いが安価な並列機構により効率化できれば、現実世界の類似の問題を大量に処理する場合にも同様の効率化が期待できる。

2 準備

フリーセルのルールは、次の通りである。

- 初期局面では、52枚の全てのカードが表が見えるようにタブローと呼ばれる8つの列に積み重ねられている。列の地(底)は上に、天は下になるように配置する。各列のカードの枚数は、4つの列が7枚で他の4つの列が6枚である。
- フリーセルと呼ばれる1枚のカードを置く場所が4箇所あり、さらにホームセルと呼ばれるカードを置く場所がある。
- フリーセルの手は、フリーセルかタブローの天のカードを1枚だけ取り上げ、次に、取り上げたカードをフリーセルかタブローかホームセルに置くという2段階から成る。ただし、取り上げたカードを置ける場所は、次の4通りである。(a) 空のフリーセル、(b) 空のタブロー列、(c) タブローの天

のカードで、順位が取り上げたカードよりも1大きく、かつ、取り上げたカードと色が異なるものの上、(d) 取り上げたカードと同じスーツで順位が1小さいカードがホームセルに有るときのホームセル。

- 52枚のカード全てをホームセルに移動することがゲームの目的である。

3 A* アルゴリズム

ディープラーニング用の質の良い訓練データとして最も望ましいものは、最短手順の解である。初期局面とは限らない与えられたフリーセルの局面からの最短手順の求解は、許容的ヒューリスティック関数を使ったA* アルゴリズムで実装できる。その場合ヒューリスティック関数の性能が良い程探索量が少ない。

Paulらは、許容的なヒューリスティック関数 $h(n)$ としてホームセル以外にあるカード枚数に関数 $m_e(n)$ を加えたものを提案し、その有効性を主張している [1]。ただし、 $m_e(n)$ の正確な値を求めるのは困難であり、1スートデッドロックサイクルと2スートデッドロックサイクルの検出により近似値を求めることを提案している。1スートデッドロックサイクルの検出により次の条件 A を満たすカード x を直接ホームセルに移動することができないことが分かり、これらのカードの枚数を許容的ヒューリスティック関数に加算することができる。

条件 A: x がタブロー内にあり、かつ、同じ列の地の側に x と同スーツで順位が x よりも小さいカードが存在する。

一方、2スートデッドロックサイクルの検出は、手間が極めて大きいため、本研究では、デッドロックグラフの部分グラフの強連結成分への分解に基づいて許容的ヒューリスティック関数の性能の向上を図ることを提案する。

局面 n に対するデッドロックグラフは、52枚のカードを点とし、次で定義する有向辺集合をもつ52点の有向グラフ $G(n)$ である。 $G(n)$ の有向辺は2種類からな

† 岡山大学大学院自然科学研究科

Graduate School of Natural Science and Technology, Okayama University

り、一つは、 v と w のスートが同じで w の順位が v の順位よりも 1 小さいときの vw であり、定常辺と呼び、もう一つは、同一のタブロー列内で w が v の天側に隣接しているときの vw であり、封鎖辺と呼ぶ。条件 A を満たすカード x の地側に隣接しているカード y から x への封鎖辺 yx をすべて除去したグラフ $G'(n)$ の部分グラフの強連結成分で辺を含むものの個数は、許容的ヒューリスティック関数に加算することができる。従って、線形時間アルゴリズムである Kosaraju のアルゴリズムにより $G'(n)$ を強連結成分に分解し、サイズの大きい強連結成分に対して、適当な辺の除去により辺を含む強連結成分の個数の増加を図る。

4 深さ優先探索

深さ優先探索は、計算資源が乏しい場合に有効であり、解が存在しないことを証明する場合は、一般的に A^* アルゴリズムよりも効率的である。さらに、深さ優先探索の 1 局面当りの処理時間と主記憶使用量が A^* アルゴリズムに比べて小さいため最短手順を求める場合でも A^* アルゴリズムに匹敵する性能が得られる可能性がある。本研究では、一般的な分枝の優先順位付けの工夫以外に次に述べる解手順の改善手法を取り入れている。

解手順の中にカードを場所 a から b に移動する手 (a, b) があり b がホームセルでないとする。この手の理由は、その後次の 3 つの目的のどれかを達成するためである。(a1) a の場所に別のカードを置くため。従って、その後 (x, a) の形の手が現れる。(a2) a がタブロー内でカード k が a から移動したことにより現れたカードを別の場所に移動するため。従って、その後 (a, x) の手が現れる。(b) カード k が移動した先 b がタブロー内で、 b の場所に別のカードを置くため。従って、その後 (x, b) の形の手が現れる。

b がホームセルでないので、 (a, b) の手の後必ず (b, y) の形の手が現れる。上の (a1), (a2), (b) の目的達成の前に (b, y) の形の手が現れたら、最初の (a, b) の手は無駄だったことになり、最初の (a, b) を省いて (b, y) を (a, y) に変えればよいことになる。これによって 1 手短くなる。

5 局面の表現形式

A^* アルゴリズムでは、大量の局面を主記憶上に保存する必要があるのでサイズが小さいことが望ましいが、保存と復元の手間はなるべく減らしたい。また、与え

られた局面が解をもたないことを深さ優先探索で証明する場合には、同一局面を再訪問したことを確実に判定するためにハッシュ表に訪問した全局面の完全な情報を保存する必要があるが、その場合なるべく簡潔な表現形式が望ましい。

次のような局面の表現形式の採用が考えられる。各カード x に対して、その存在状況を表す情報を次のように 3 ビットの整数値 $y(x)$ で表現できる。 x がフリーセル内にあるとき $y(x) = 0$ 、 x がタブロー内の地のカードであるとき $y(x) = 1$ 、 x がホームセル内にあるとき $y(x) = 2$ 、 x がタブロー内にあり、 x の地側に x よりも順位が 1 大きく x と色が異なるカードの内弱いスートの方が隣接しているとき $y(x) = 3$ 、同様のカードの内強いスートの方が隣接しているとき $y(x) = 4$ 、 x がタブロー内にあり、 x と x の地側に隣接しているカードが両方とも初期局面から動いていないとき $y(x) = 5$ で表す。さらに、 x がキングのとき $y(x)$ を 2 ビットで表現でき、8 枚のカードの $y(x)$ の並びを 21 ビットで表現できるので、局面全体を $21 \cdot 6 + 2 \cdot 4 = 134$ ビット (17 バイト) で表現することができる。

また、詳細は省略するが、 x がタブロー内のカードのときに天の側に隣接しているカードの情報を取り出せるように各カードの存在状況を表す情報を 4 ビットの整数値で表現することも可能である。この表現形式は、 A^* アルゴリズムの設計に応用する予定である。

6 おわりに

提案した許容的ヒューリスティック関数を使った A^* アルゴリズムと深さ優先探索に基づいた最短手順を求める探索アルゴリズムの実装は、本論文執筆時点で未完成である。講演では、完成したソルバーによる実験結果を公表する予定である。

謝辞 本研究の一部は、株式会社アースライズカンパニーの支援により実施された。本研究は主に九州大学情報基盤研究開発センターの研究用計算機システムを利用した。

参考文献

- [1] Gerald Paul and Malte Helmert. Optimal solitaire game solutions using A^* search and deadlock analysis. In *Ninth Annual Symposium on Combinatorial Search*, 2016.