

モンテカルロ法に基づく Flood-It の AI に関する研究

周・伊藤研究室 学部 4 年 小田将也

1 はじめに

Flood-It とは, $n \times m$ の色分けされたグリッド上で行うゲームである. Flood-It はグリッドの一番左上にあるマスをもとに (ピボット) として進めていくゲームであり, 1 回の操作でピボットをある色から別の色へ変更する. ピボットから上下左右に現在の色のみを辿ってたどり着くことができるマスの集合が, 自分の領地であり, ピボットの色の変更と同時に自分の領地の全てのマスの色が同じ色になる. 図 1 に操作の例を示す. 操作前図 (a) の斜線で表した 3 マスが自分の領地であり, ここにピボットの色を 1 から 2 へ変える操作を適用すると, 操作後図 (b) のグリッドが得られる. この操作を繰り返し $n \times m$ マス全てのマスを自分の領地にすることが目的である. このとき, 操作数が最小となるような操作列を求めることが Flood-It の問題である. Flood-It の問題に関しては, 既知の結果としてグリッドの大きさが $n \times n$ ならば NP 困難 [3], $3 \times n$ ならば NP 困難 [4], $2 \times n$ ならどんな色数でも多項式時間で解けることが知られている [5].

今回は, Flood-It を二人対戦用ゲームにしたものを考える. 二人用 Flood-It は, 同じく $n \times m$ の色分けされたグリッド上で行う. 一番左上のマスと一番右下のマスを各プレイヤーのピボットとし, グリッドが二色になるまで交互にピボットの色を変える操作を繰り返す. 最終的に相手プレイヤーよりも自分の領地が広がれば自分の勝ち, というルールである.

本研究では, 二人用の Flood-It に対して, モンテカルロ法に基づく対戦用 AI を作成する.

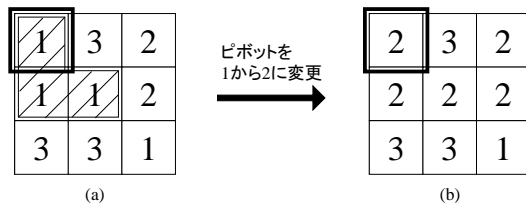


図 1: 1 回の操作の例. マス目の中の数字は色を表す.

2 定義・準備

2.1 グラフ

無向グラフ $G = (V, E)$ を頂点集合 V と辺集合 $E \subseteq V^2$ の組とする. 本研究では, G は単純かつ有限であるとする. V の 2 頂点 u, v に対して, $\{u, v\} \in E$ ならば u と v は G において隣接しているという.

グラフ $G = (V, E)$ 内の点 v_0 から点 v_l への歩道 W は, 連続した点の列 $\langle v_0, v_1, \dots, v_l \rangle$ である. ここで, $1 \leq i \leq l$ なる各整数について, $\{v_{i-1}, v_i\} \in E$ である. このとき, l を歩道の長さという. さらに, v_0 から v_l への歩道を $\langle v_0, v_l \rangle$ 歩道と呼ぶ.

$\langle u, v \rangle$ 歩道に含まれる u と v 以外の全ての頂点が異なるとき, $\langle u, v \rangle$ 歩道は $\langle u, v \rangle$ 道 (パス), あるいは単に u と v を結ぶ道 (パス) と呼ばれる. また, 長さが正である $\langle u, u \rangle$ 道は閉路 (サイクル) と呼ばれる. グラフ G の任意の 2 点 $u, v \in V$ の間に常に $\langle u, v \rangle$ 道があるとき, G は連結であるという. もし G に閉路が存在しないかつ連結ならば, G は木である.

グラフ $G = (V, E)$, $G' = (V', E')$ において, $V' \subseteq V$ かつ $E' \subseteq E$ であるとき, G' は G の部分グラフという. 特に, $E' = \{\{u, v\} \in E \mid u, v \in V'\}$ であるとき, G' は G の誘導部分グラフという. もし G' が木であるならば, G' は誘導木という. 図 2 は誘導木の例である.

2.2 誘導木遷移問題

G における二つの誘導木 T_0, T_r が与えられたとき, T_0 から T_r へ段階的に遷移可能かどうかを判定する問題を誘導木遷移問題 (ITRECONF) という. ここで, 誘導木遷移問題において段階的に遷移可能であるとは, 以下の条件を満たす誘導木の系列 $\langle T_0, T_1, \dots, T_\ell \rangle$ が存在することという.

- 各 $i \in \{0, 1, \dots, \ell\}$ に対し, T_i は誘導木であり, $T_\ell = T_r$ である.
- 各 $i \in \{0, 1, \dots, \ell - 1\}$ に対し, T_{i+1} は T_i にトークンの操作をちょうど一回行うことで得られる.

ここで, 誘導木の各頂点にトークンが置かれているとすると, 誘導木遷移問題には TJ ルール, TS ルール, TAR ルールの三つの遷移条件がある.

TJ ルールとは各 $i \in \{1, 2, \dots, \ell\}$ に対し, $T_{i-1} \setminus T_i = \{u\}$ かつ $T_i \setminus T_{i-1} = \{v\}$, つまり $u \in T_{i-1}$ のトークンを $v \in T_i$ へ移して T_{i-1} から T_i を得ることを表す.

TS ルールとは各 $i \in \{1, 2, \dots, \ell\}$ に対し, $T_{i-1} \setminus T_i = \{u\}$ かつ $T_i \setminus T_{i-1} = \{v\}$ かつ $\{u, v\} \in E$, つまり $u \in T_{i-1}$ のトークンを辺 $\{u, v\}$ に沿って $v \in T_i$ へ移して T_{i-1} から T_i を得ることを表す.

TAR ルールとは各 $i \in \{1, 2, \dots, \ell\}$ に対し, $|T_i - 1 \triangle T_i| = 1$, つまり $u \in T_{i-1}$ のトークンを削除, または $u \notin T_{i-1}$ のトークンを追加して T_{i-1} から T_i を得ることを表す.

系列 $\langle T_0, T_1, \dots, T_\ell = T_r \rangle$ を誘導木遷移問題における T_0 から T_r への遷移列という. 図 2 は誘導木 T_0 から T_r への TS ルールの下での遷移を表しており, 遷移列は $\langle T_0, T_1, T_2 = T_r \rangle$ となる.

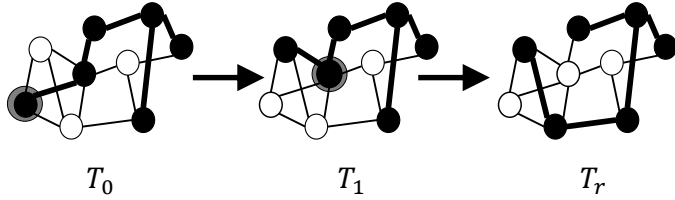


図 2: 遷移列の例

また, 最短誘導木遷移問題を T_0 から T_r へ長さが高々 ℓ の遷移列において段階的に遷移可能かどうかを判定する問題と定義する.

誘導木遷移問題と最短誘導木遷移問題は, TJ ルールと TS ルールにも同様に定義することができる.

3 TS ルールの $W[1]$ 困難性

TS ルールの下での ITRECONF における解サイズを k , 遷移列の長さを ℓ と表すこととする. 解サイズ k と遷移列の長さ ℓ の二つをパラメータとしたとき, 次の定理を示した.

定理 1. ITRECONF は TS ルールの下で $k + \ell$ をパラメータとしたときに $W[1]$ 困難である. ただし, k は誘導木のサイズであり, ℓ は遷移列の長さである.

我々は, 独立点集合問題から ITRECONF へのパラメータ化帰着により, 本問題の $W[1]$ 困難性を示す.

独立点集合問題は, グラフ $G = (V, E)$ と正整数 t が入力として与えられる. このとき, S がグラフ G の独立点

集合であり, $|S| = t$ となるような頂点の部分集合 $S \subseteq V$ が存在するかを問う問題である.

独立点集合問題について以下の定理が知られている.

定理 2. ([1]) 独立点集合問題は独立点集合のサイズ t をパラメータとしたときに $W[1]$ 完全である.

まずはインスタンスの構成を与える.

独立点集合問題の任意のインスタンスであるグラフ $G' = (V', E')$ を考える. これに対して, ITRECONF のインスタンスを構成するため以下のグラフ $G = (V, E)$ を与える.

$$V = K' \cup \{v, x\} \cup U \cup W$$

$$E = \{v, x\} \cup \{\{v, y\} \mid y \in K' \cup U \cup W\} \cup \{\{u_i, w_j\} \mid i, j = 1, \dots, t+1\} \cup \{\{v_x^i, v_y^j\}, \{v_y^i, v_x^j\} \mid i, j = 1, \dots, t, i \neq j, x, y = 1, \dots, n, \{v_x, v_y\} \in E'\} \cup \{\{v_a^i, v_a^j\} \mid i, j = 1, \dots, t, i \neq j, a = 1, \dots, n\}$$

ただし, $U = \{u_1, \dots, u_{t+1}\}$, $W = \{w_1, \dots, w_{t+1}\}$, $V(K_i) = \{v_1^i, v_2^i, \dots, v_n^i\} (i = 1, \dots, n)$, $n = |V'|$ であり, K_1, \dots, K_t はグラフ G' の頂点集合の完全グラフである. $K' = K_1 \cup \dots \cup K_t$ とする. また, ITRECONF の初期解 T_0 , 目標解 T_r を $T_0 = \{v, x\} \cup U$, $T_r = \{v, x\} \cup W$ とする. T_0, T_r は誘導木となっている. グラフ G の図を図 3 に示す.

以下では, 定理 1 の証明を与える.

証明. 本章で構成したインスタンスを用いる. この構成は, 独立点集合問題の入力の多項式時間で行うことができる. また, ITRECONF の誘導木のサイズおよび遷移列の長さは独立点集合問題の入力の t のみに依存している.

次に, グラフ G' においてサイズ t の独立点集合が存在するとき, かつそのときに限りグラフ G において TS ルールの下で T_0 から T_r へ遷移可能であることを示す.

グラフ G' においてサイズ t の独立点集合が存在するとき, グラフ G において TS ルールの下で T_0 から T_r へ遷移可能であることを示す. グラフ G' においてサイズ t の独立点集合が存在するとき, K_1, \dots, K_k において, G' のサイズ t の独立点集合に対応する頂点集合を考える. K_1 において, G' のサイズ t の独立点集合に対応する頂点集合から 1 点 s_1 を選び, K_2 において, G' のサイズ t の独立点集合に対応する頂点集合から K_1 で選んだ頂点と異なる頂点 s_2 を選ぶ. 同様に, K_t において, G' のサイズ t の独立点集合に対応する頂点集合から K_1, \dots, K_{t-1} で選んだ頂点と異なる頂点 s_t を選ぶ. このように選んだ頂点の集合を $S = \{s_1, s_2, \dots, s_t\}$ とする. 独立点集合の定義より S の任意の 2 点間には辺が存在しないので, $T_S = \{v, x\} \cup S$ は誘導木である. したがって, 以下のような遷移列を構成することができる. まず初めに, 各 $i \in \{1, \dots, t\}$ における $u_i \in U$ のトークンを S に移動す

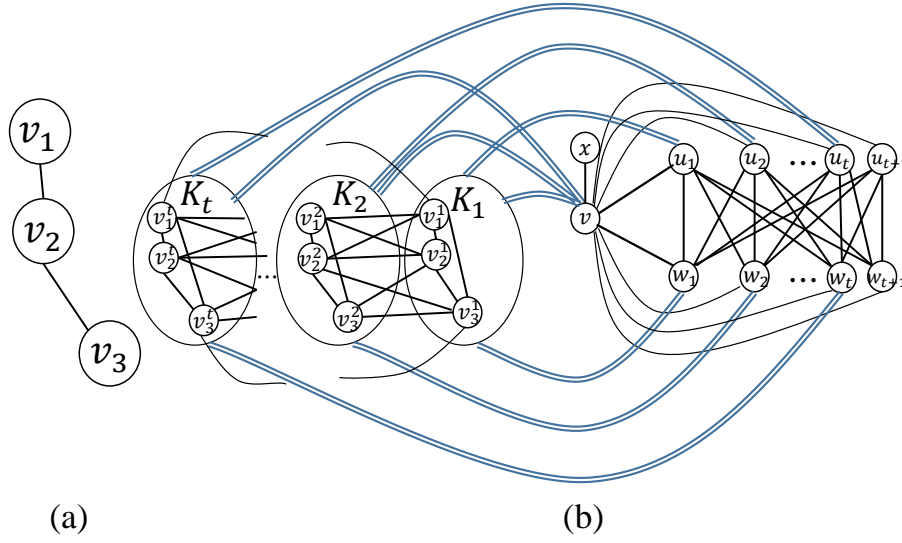


図 3: (a) 独立点集合問題のグラフ G における頂点 v_1, v_2, v_3 およびそれらに接続している辺. (b) インスタンスの構成. 二重線は $K_i, i = 1, \dots, t$ の全ての頂点に接続していることを表す. また, (a) における独立点集合問題のグラフ G の頂点 v_1, v_2, v_3 に対応する頂点 $v_1^i, v_2^i, v_3^i (i = 1, \dots, t)$ の辺の接続は図のようになっている. ここで $\{v_1^i, v_3^j\} \notin E(i, j = 1, \dots, t, i \neq j)$ であることに注意する

る. 次に u_{t+1} のトークンを w_{t+1} に移動する. 最後に, S のトークンを一つずつ $w_i \in W, i \in \{1, \dots, t\}$ に移動する. 以上により, グラフ G において TS ルールの下で T_0 から T_r へ遷移可能である.

グラフ G において TS ルールの下で T_0 から T_r へ遷移可能であるとき, グラフ G' においてサイズ t の独立点集合が存在することを示す. グラフ G において TS ルールの下で T_0 から T_r へ遷移可能であるとき, W の任意の 1 点に初めてトークンが移動する瞬間があるはずである. v にトークンが置かれていないとすると非連結になってしまうため, 常に v にはトークンが置かれており, また x にも常にトークンが置かれている. W の任意の 1 点に初めてトークンを移動したとき, U にトークンが残っていると U の任意の 1 点と v と W に追加した 1 点で閉路ができてしまう. したがって, W の任意の 1 点に初めてトークンが移動する直前において, v, x と W の任意の 1 点に移動するトークン以外の t 個のトークンはすべて S に置かれている. 以上により, グラフ G' においてサイズ t の独立点集合が存在する. \square

4 まとめと今後の課題

参考文献

- [1] R.G. Downey and M.R. Fellows, Fixed-parameter tractability and completeness II: on completeness for $W[1]$, Theoretical Computer Science 141(1-2), pp. 109–131, 1995.
- [2] R.A. Hearn and E.D. Demaine, PSPACE-completeness of sliding-block puzzles and other problems through the nondeterministic constraint logic model of computation, Theoretical Computer Science 343(1-2), pp. 72–96, 2005
- [3] T. Ito, E.D. Demaine, N.J.A. Harvey, C.H. Papadimitriou, M. Sideri, R. Uehara and Y. Uno, On the complexity of reconfiguration problems, Theoretical Computer Science 412(12-14), pp. 1054–1065, 2011
- [4] M. Kamiński, P. Medvedev and M. Milanić, Complexity of independent set reconfigurability problems, Theoretical Computer Science 439, pp. 9–15, 2012

表 1: 再帰アルゴリズム

アルゴリズム 2 : $Recursive(n)$, n :整数

```

1: for  $x = n$  to  $\lceil n/2 \rceil + 1$ 
2:    $x_3$  を  $x$  番のバケツに move
3:    $x_2$  を  $x$  番のバケツに move
4: end for
5: for  $x = 1$  to  $n$ 
6:   1 番のバケツから  $\lceil n/3 \rceil$  番のバケツにある  $x_1$ 
     のボールを  $\lceil x/3 \rceil$  にリラベルする
7: end for
8: if  $\lceil n/3 \rceil$  番のバケツに  $y_2$  か  $y_3 (y \in [n])$  がある then
9:   そのボールを  $\lceil n/3 \rceil$  にリラベルする
10: 最初の  $\lceil n/3 \rceil$  個のバケツについて
      $Recursive(\lceil n/3 \rceil)$  を適用しボールを並び替える
11: ラベルを元に戻す
12: for  $x = n$  to 2
13:    $x_1$  を  $x$  番のバケツに move
14: end for

```

- [5] M. Karoński and Z. Palka, On the size of a maximal induced tree in a random graph, *Mathematica Slovaca* 30, pp. 151–155, 1980
- [6] A.E. Mouawad, N. Nishimura, V. Raman, N. Simjour and A. Suzuki, On the parameterized complexity of reconfiguration problems, *Algorithmica* 78(1) , pp. 274–297, 2017
- [7] K. Wasa, K. Yamanaka and H. Arimura, The complexity of induced tree reconfiguration problems, *Proceedings of The 10th International Conference on Language and Automata Theory and Applications (LATA 2016)*, *Lecture Notes in Computer Science* 9618, pp. 330–342, 2016