Title	Flood Filling Gameの計算量に関する研究
Author(s)	福井，宏行
Citation	
Issue Date	2013-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/11317
Rights	
Description	Supervisor:上原　隆平，情報科学研究科，修士

修 士 論 文

Flood Filling Gameの計算量に関する研究

北陸先端科学技術大学院大学
情報科学研究科情報科学専攻

福井 宏行

2013年3月

修 士 論 文

Flood Filling Gameの
計算量に関する研究

指導教員　上原 隆平 教授 （代理：石原 哉 教授）

審査委員主査　石原 哉 教授
審査委員　浅野 哲夫 教授
審査委員　小川 瑞史 教授

北陸先端科学技術大学院大学
情報科学研究科情報科学専攻

1010058 福井 宏行

提出年月: 2013 年 2 月

概 要

Flood Filling Game は，Flood-It!，Mad Virus，Honey-Bee Game などと呼ばれ，オンライン上で遊ばれている彩色ゲームの一種である．これらのゲームは，プレイヤーが盤面上のある特定のセルの色を変えると，色の変わるセルに隣接している同じ色のセルも共に色が変わる．これを繰り返し，盤面の色を統一することを目指すゲームである．この Flood Filling Game を一般化すると，一般のグラフ上での彩色問題となる．この一般化の際に，色を変えるセルを選択できるものを「Free Flood Filling Game」と呼ぶ．

本研究では，この一般化した Free Flood Filling Game の計算量クラスを求めた．まず，tree 上では 3 色以上では NP 完全であることを証明し，及び path 上では何色であっても多項式時間で解けるアルゴリズムを提案した．次に，色の数に制限の無い状態での，多項式時間で解けるグラフクラスと NP 完全であるグラフクラスとのよりタイトな境界を求めるため，caterpillar，proper interval graph に対し計算量クラスを求め，そのどちらも NP 完全であることを証明した．また，split graph に対しても，色の数に制限が無い場合は NP 完全であることを証明し，色の数が定数個以下であれば多項式時間で解けるアルゴリズムを提案した．以上の結果を整理し，既存研究と比較検討することで，色の数に制限の無い Free Flood Filling Game の計算量クラスが変化する境界を正確に求めることができた．

目 次

第1章　はじめに

1.1　研究背景

情報・物質の拡散に関する問題は，現代科学において情報分野のみならず，決められた範囲全体への情報の拡散速度の向上や，病原菌や火災のような災害の拡がる速度の研究などの分野で必要とされており，重要視されている．そのため，このような問題のモデル化は，最近よく研究されている．

本研究では，このような問題のモデル化の１つである「Flood Filling Game」を研究対象とする．Flood Filling Game とは，色の塗られたセルで構成された初期盤面が与えられ，プレイヤーは毎回特定のセルの色を変えるゲームである．このとき，色の変わるセルに隣接している同色のセルも一続きの領域とみなし，共に同じ色に変化する．これを繰り返すことで同色の領域を拡げ，なるべく少ない手数で盤面全体の色を統一することを目指す．このゲームの自然な一般化を行い，様々に条件を変えることで，多項式時間で解ける場合や，NP 完全となる場合を明らかにし，その条件を整理することが，本研究の目的である．

1.2　本稿で扱うグラフクラス

本稿で扱う，もしくは関連するグラフクラスを説明する．

グラフ $G = (V, E)$ の頂点集合 $V = \{v_1, v_2, ..., v_n\}$ において，全ての辺 $\{v_i, v_{i+1}\}$ が辺集合 E に含まれており，それ以外の辺は E に含まれていないとき，そのグラフ G を path といい，$(v_1, v_2, ..., v_n)$ と書く．このとき，頂点 v_1, v_n を，path G の端点と呼ぶ．グラフ

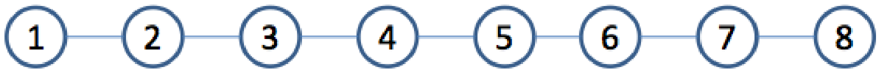


図 1.1: path

$G = (V, E)$ の頂点集合 $V = \{v_1, v_2, ..., v_n\}$ において，全ての辺 $\{v_i, v_{i+1}\}$ および辺 $\{v_n, v_1\}$ が辺集合 E に含まれており，それ以外の辺は E に含まれていないとき，そのグラフ G を cycle といい，$(v_1, v_2, ..., v_n, v_1)$ と書く．任意の i, j に対して $\{v_i, v_j\} \in E$ であるとき，そのグラフを clique という．すなわち，clique は全ての頂点同士が連結したグラフである．

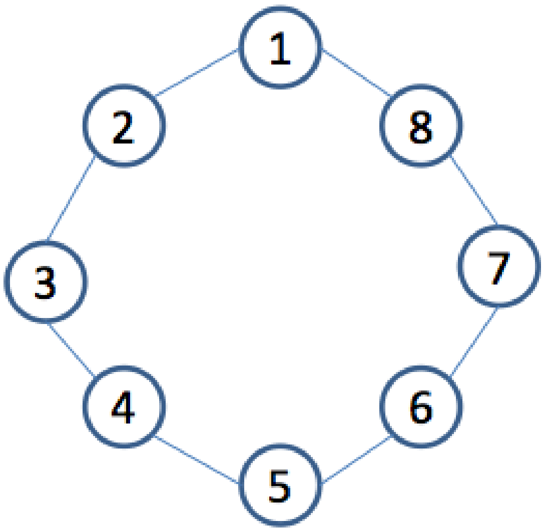


図 1.2: cycle

グラフ $G = (V, E)$ において，cycle を部分グラフとして持たないとき，そのグラフを tree という．すなわち，任意の頂点部分集合 $U \subseteq V$ において，部分グラフ $G[U] = (U, F)$ が cycle にならないグラフである．

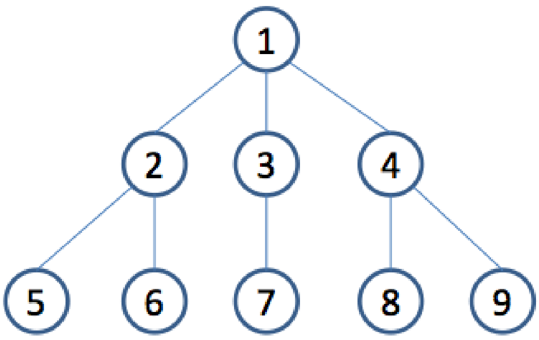


図 1.3: tree

$V = \{v_1, v_2, ..., v_n\}$ であるグラフ $G = (V, E)$ が，$1 \leq i, j \leq n$ である全ての i, j に対し，$\{v_i, v_j\} \in E$ であるとき，またその時に限り $I_{v_i} \cap I_{v_j} \neq \emptyset$ である区間の集合 $\mathcal{I} = \{I_{v_1}, I_{v_2}, ..., I_{v_n}\}$ で表現できるとき，グラフ G を interval graph といい，この区間の集合 $\mathcal{I}$ を G の interval representation という．区間 I に対し，区間の左側の端点と右側の端点をそれぞれ $L(I), R(I)$ と呼ぶ．このとき，$L(I) \leq R(I)$ であり，$I = [L(I), R(I)]$ である．一方の区間がもう一方の区間に完全に含まれるような異なる 2 つの区間 I, J が存在しない interval representation を，proper であるといい，このような proper である interval representation を持つグラフを proper interval graph と呼ぶ．interval graph G の interval representation $\mathcal{I}$ が，全て同じ長さの区間で構成できるとき，G を unit interval graph と

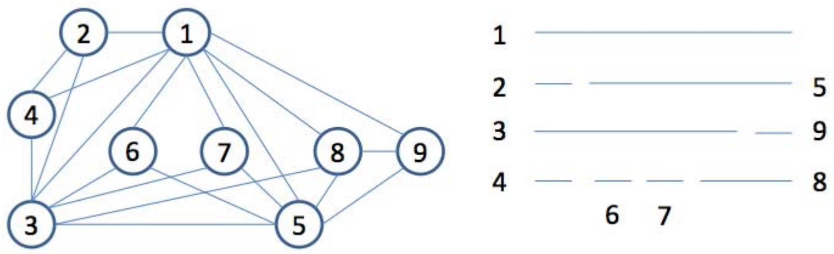


図 1.4: interval graph

呼ぶ．proper interval graph と unit interval graph は等価であることが知られている [11]．これらは互いに，容易に変換することが可能である [2]．

グラフ $G=(V,E)$ を構成する頂点集合 V が，グラフ $G[B]$ が path である頂点部分集合 B と，全ての頂点が，頂点集合 B のうちただ一つの頂点のみと連結する頂点部分集合 H に分けられるとき，グラフ G を caterpillar という．このとき，B 及び $G[B]$ を背骨，H に含まれる全ての頂点を毛と呼ぶ．また，caterpillar G は interval graph でもある．

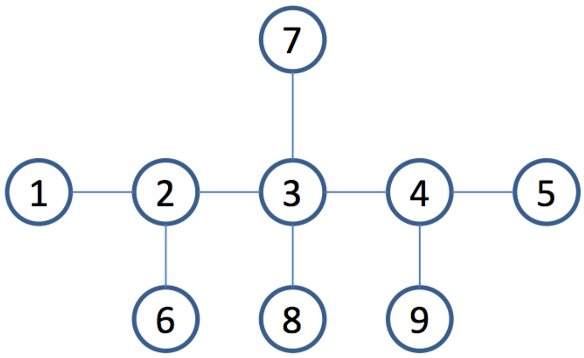


図 1.5: caterpillar

グラフ $G=(V,E)$ を構成する頂点集合 V が，グラフ $G[C]$ が clique である頂点部分集合 C と，全ての頂点が独立した頂点部分集合 I に分けられるとき，グラフ G を split graph という．

1.3 Flood Filling Game

1.3.1 Flood Filling Game の一例

Flood Filling Game のより詳細な内容について，例を挙げながら説明する．

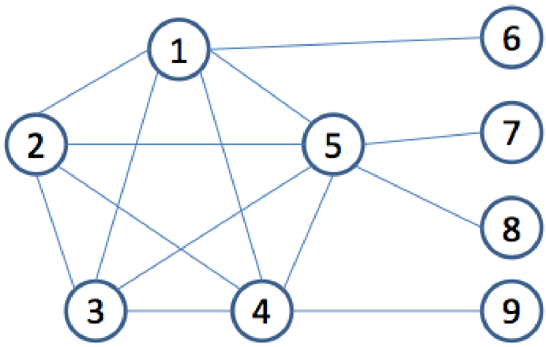


図 1.6: split graph

Flood Filling Game の代表的なものに Flood-It![1] というゲームが存在する．これは，１人用のゲームで盤面はグリッド状であり，プレイヤーは最も左上のセルの色のみを変えることができる．決められた手数以内に盤面全てのセルの色を統一することが，プレイヤーの目的である．

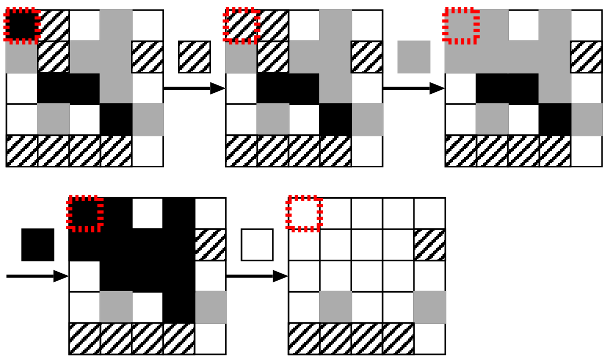


図 1.7: Flood-It!

１人用のゲームで盤面がハニカム状である，Mad Virus [2] というゲームも存在する．プレイヤーが色を変えることのできるセルは，盤面の端からランダムに選ばれた４つのセルのうちから１つを選択する．これも，決められた手数以内に盤面全てのセルの色を統一することが，プレイヤーの目的である．

また，Honey-Bee Game[3] というゲームも存在し，これは人間対コンピュータの２人用ゲームである．盤面はハニカム状であり，プレイヤーは最も左上の，コンピュータは最も右下のセルの色を変えることができる．ただし，プレイヤーは今のコンピュータの，コンピュータは今のプレイヤーの色と同じ色に変えることはできない．これを盤面の全てのセ

[1] http://floodit.appspot.com/
[2] http://http://www.bubblebox.com/play/puzzle/539.htm
[3] http://www.ursulinen.asn-graz.ac.at/Bugs/htm/games/biene.htm

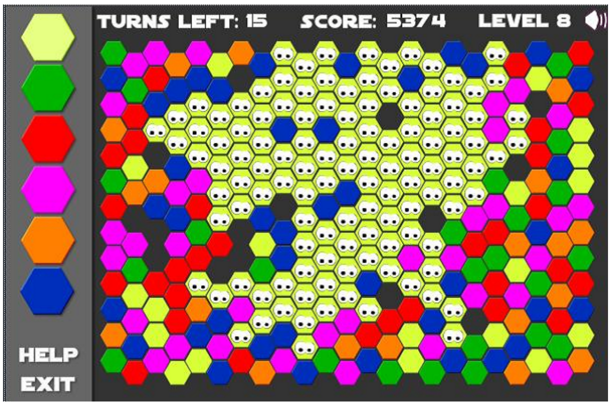


図 1.8: Mad Virus

ルがプレイヤーかコンピュータのどちらかの色になるまで繰り返し，半分以上のセルを自分の色にすることが目的である．

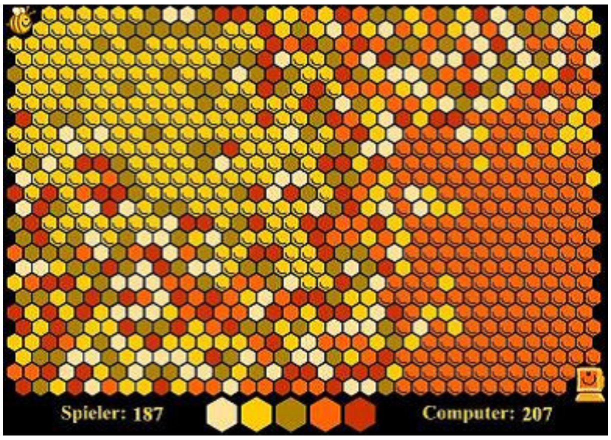


図 1.9: Honey-Bee game

これらのゲームはウェブ上でプレイすることが可能である．

1.3.2　Flood Filling Game の一般化

Flood Filling Game を以下のように一般化する．

盤面は，一般の連結，単純な無向グラフに対して考えることにする．Flood Filling Game において，連結でないグラフは別々に解くしかない．単純でないグラフの場合，多重辺と普通の辺において差異が無く，自身への辺も意味を成さない．隣接したセルの色を変えるという性質を表現すると，無効グラフとするのが自然である．特定の１つのセルのみ色を変えることのできる「Fixed」と，色を変えるセルを１手ごとに自由に選ぶことのできる

「Free」を考える．盤面に現れる色の数が，ある定数で制限されているのか，制限が無いのかを考える．また，ゲームの人数が１人か２人かの設定を考えることもある．

以下，本研究では，１人ゲームについて考えるものとする．また，本稿でグラフと言った場合，一般の連結，単純な無向グラフについて考えるものとする．

上記のパラメータの変化によって，盤面の全てのセルの色を統一する最小の手を求める最適化問題（以下，最適化問題），および盤面の全てのセルの色を k 手で統一できるかを決定する問題（以下，決定問題）を考え，その計算量がどのように変化するかを研究する．

free の Flood Filling Game（以下，Free Flood Filling Game）の一般化についてより厳密に定義すると，以下のようになる．

入力として，まず，グラフ $G = (V, E)$ が与えられる．$V = \{v_1, v_2, ..., v_n\}$ は頂点集合，$E = \{e_1, e_2, ..., e_m\}$ は辺集合である．色集合 $C = \{c_1, c_2, ..., c_k\}$ が存在し，色の数に制限がある場合，k は定数である．また，各頂点の現在の色である，$col(v) \in C$ が与えられる．$col(v_i) = c_j$ となっている場合，頂点 v_i の現在の色が c_j であることを表し，初期状態で $col(v_i) = c_j$ である場合，頂点 v_i の初期状態の色が c_j であることを表す．頂点部分集合 $U \subseteq V$ が与えられたとき，$F = E \cap \{\{u, v\} | u, v \in U\}$ であるグラフ $G[U] = (U, F)$ を誘導部分グラフという．色 $c \in C$ に対して，頂点集合 V の中で $col(v) = c$ である全ての頂点 v による集合を部分集合 $V_c \subseteq V$ とする．頂点 v と色 c に対して，$col(v) = c$ であるとき，色 c 近傍 $N_c(v)$ を，グラフ $G[V_c]$ 上における，頂点 v を含む連結成分とする．$col(v) \neq c$ であるとき，色 c 近傍 $N_c(v) = \emptyset$ とする．$v \in V, c \in C$ である彩色指令 $(v, c) \in V \times C$ が与えられると，頂点 v 及び頂点 v の色 $col(v)$ 近傍 $N_{col(v)}(v)$ に含まれる頂点全ての色を c に変える．すなわち，変換後は $col(v) = c$ となる．t 個の彩色指令列 $(v_1, c_1), (v_2, c_2), ..., (v_{t'}, c_{t'})$ が与えられるとき，初期状態でのグラフを G_0，$i(i = 1, 2, ..., t)$ 番目の彩色指令 (v_i, c_i) までの変換を行った状態でのグラフを G_i とする．

Free Flood Filling Game の最適化問題，決定問題は，それぞれ以下のように定義できる．

問題 1 *Free Flood Filling Game* の最適化問題
入力． 頂点集合 V に含まれる全ての頂点に初期状態での色 $col(v) \in C$ が与えられたグラフ $G = (V, E)$
出力． グラフ G を単色のグラフに変換する彩色指令列 $(v_1, c_1), (v_2, c_2), ..., (v_t, c_t)$ のうち，彩色指令列の長さ t の値が最も小さいもの

問題 2 *Free Flood Filling Game* の決定問題
入力． 頂点集合 V に含まれる全ての頂点に初期状態での色 $col(v) \in C$ が与えられたグラフ $G = (V, E)$，及び整数 t
出力． グラフ G を単色のグラフに変換する彩色指令列 $(v_1, c_1), (v_2, c_2), ..., (v_t, c_t)$ のうち，彩色指令列の長さが定数 t 以下であるものが存在するか

このとき，各グラフクラス，条件に対して最適化問題，決定問題を解く際の計算量を求めるのが，本研究の内容である．

1.4　既存の研究結果

Flood Filling Game は，2010 年頃から急速に研究が進められているゲームである．Fleischer，Woeginger [4] により，以下が示された．

- fixed，split graph に対して，色の数に制限が無い場合 NP 完全，色の数が制限されている場合多項式時間で解ける．
- fixed，tree に対して，３色以上の場合 NP 完全である．
- fixed，cocomparability graph に対して多項式時間で解ける．

また，Clifford ら [1][3] により，以下が示された．

- fixed/free，一般のグラフに対して，３色以上の場合 NP 完全である．

更に，Lagoutte [8] により，以下が示された．

- fixed/free，一般のグラフに対して，２色以下の場合多項式時間で解ける．

これらの結果から，単純なグラフクラスに対しては，最適化問題が多項式時間で解ける場合が多いのではないかと考え，主に，色を変えるセルを自由に選ぶことのできる「Free Flood Filling Game」を対象に研究を行った．

第2章　本研究での結果

2.1　全体の結果

本研究では，Free Flood Filling Game の１人ゲームの最適化問題に関して，以下の定理を証明した．

まず，ゲームの一般化問題においてよく研究対象に挙げられる，tree と path に対して計算量を求めた．

定理 3 *tree* 上において，３色以上の *Free Flood Filling Game* の決定問題は NP 完全である．

定理 4 *path/cycle* 上において，色の数に関わらず，*Free Flood Filling Game* の多項式時間は多項式時間で解ける．*path/cycle* の頂点数が n のとき，$O(n^3)$ 時間，$O(n^2)$ 領域で解ける．

tree と path で結果が分かれたため，どこが計算量の難しくなる境界なのかを求めるため，proper interval graph と caterpillar に対し計算量を求めた．

定理 5 *caterpillar* 上において，色の数に制限の無い *Free Flood Filling Game* の決定問題は NP 完全である．

定理 6 *proper interval graph* 上において，色の数に制限の無い *Free Flood Filling Game* の決定問題は NP 完全である．

その他，split graph に対しても計算量を求めた．

定理 7 *split graph* 上において，色の数に制限の無い *Free Flood Filling Game* の決定問題は NP 完全である．

定理 8 *split graph* 上において，色の数が k 色以下の *Free Flood Filling Game* の最適化問題は $O((k!)^2 + n)$ 時間で解ける．

これらのうち，決定問題の NP 完全性を証明する際には，以下の補題を用いる．

補題 9 一般のグラフに対して，*Flood Filling Game* の決定問題は NP に入る．

証明． グラフ G と整数 t が Flood Filling Game の決定問題の yes インスタンスならば，t 手で盤面の色を統一可能な彩色指令 $(v_1, c_1), (v_2, c_2), ..., (v_t, c_t)$ が存在する．その長さは高々頂点数 n 手であり，それを実際に塗って，盤面の色が統一されるかどうかを確認することは，多項式時間で可能である． ∎

補題 9 は，fixed/free の違い，色の数の制限を問わず，１人ゲームである限り，あらゆるグラフクラスに対して成立する．すなわち，１人ゲームの Flood Filling Game に対して，決定問題が NP 完全であることを示すには，その決定問題が NP 困難であることを示せばよい．

2.2 tree 上での NP 完全性の証明

tree 上において，３色以上の Free Flood Filling Game の決定問題は NP 完全であることを示す．

証明． 以下の補題により示す．

補題 10 *tree* 上において，３色以上の *Free Flood Filling Game* の決定問題は NP 困難である．

証明． 既存の結果である，fixed，tree に対して，３色以上の場合 NP 完全である [4] ことから還元する．

fixed の場合の元の tree を n 個コピーし，fixed な頂点 (root) を共有する．このとき，元の n 頂点の tree は，$k(\leqq n-1)$ 手で彩色できる．よって，合成してできた tree は，全ての手で共有した root を選ぶことで，k 手で 1 色にできる．このとき，もし共有した root 以外の頂点を選択した場合，同じ手を n 回繰り返さなければならないため，手数のロスしか生まない．したがって，彩色手数の決定問題は，元の fixed の問題と本質的に同じである． ∎

補題 9，補題 10 より，tree 上において，３色以上の Free Flood Filling Game の決定問題は NP 完全である． ∎

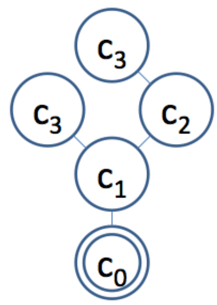


図 2.1: 元の問題の木

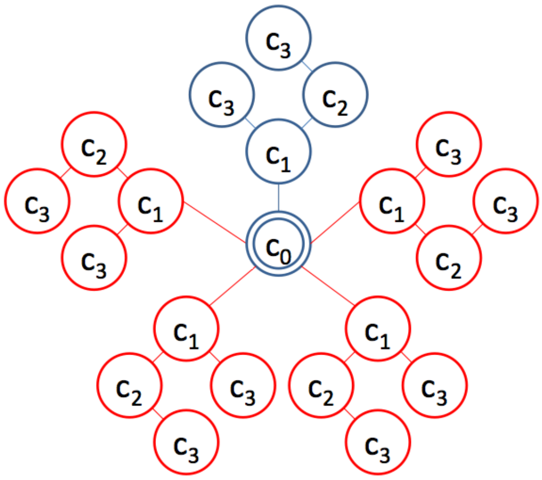


図 2.2: root を共有した木

2.3 path/cycle 上での多項式時間アルゴリズム

頂点数 n の path 上における Free Flood Filling Game の最適化問題は，色の数に関わらず $O(n^3)$ 時間，$O(n^2)$ 領域で解けることを示す．

証明． 動的計画法を用いて，多項式時間アルゴリズムを構成する．

頂点 i から頂点 j までの間にある全ての頂点 $(v_i, ..., v_j)$ を色 c で塗るための最小手数をテーブル $T[i, j, c]$ とする．このとき，頂点 i を色 c で塗るための最小手数 $T[i, i, c]$ は，元々その頂点が色 c で塗られているかによって決まり，$T[i, i, c] = 0$ or 1 である．

以下，$[i < j]$ において，頂点 $(v_i, ..., v_j)$ を色 c で塗るための最後の一手は，以下の４通りである．

- $(T[i, i', c'] + 1) + T[i' + 1, j, c]$（頂点 $i'(i \leqq i' < j)$ よりも左側を色 c' から色 c に塗り変える—$O(Cn)$）
- $T[i, i', c] + (T[i' + 1, j, c'] + 1)$（頂点 $i'(i \leqq i' < j)$ よりも右側を色 c' から色 c に塗り変える—$O(Cn)$）
- $T[i, i', c] + (T[i' + 1, j', c'] + 1) + T[j' + 1, j, c]$（頂点 $i', j'(i \leqq i' < j' < j)$ の間を色 c' から色 c に塗り変える—(OCn^2)）
- $T[i, j, c'] + 1$（頂点 i, j の間を色 c' から色 c に塗り変える—$O(C)$）

よって，$T[i, j, c]$ は，全ての i, j, c に対して上記４つをそれぞれの i', j', c' に対して調べ，そのうちの最小のものを選べばよい．このアルゴリズムを単純に実装すると，色の数が C であるとき，ループ全体が $O(Cn^2)$ 時間，上記４つの最後の一手のうち最長である３つ目のパターンが，$O(Cn^2)$ 時間，$O(Cn^2)$ 領域必要であるため，全体では $O(C^2n^4)$ 時間，$O(Cn^2)$ 領域必要である．

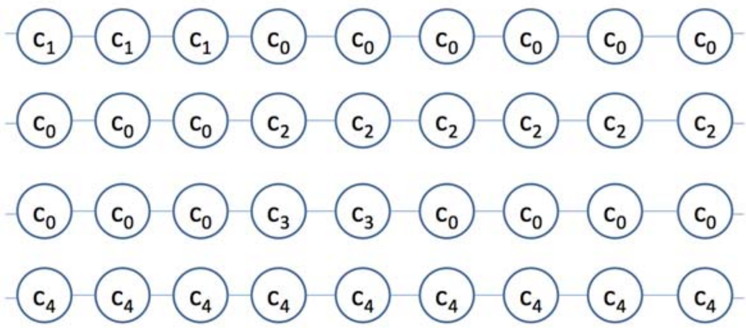


図 2.3: 最後の一手

このとき，3つ目のテーブル $T[i,i',c]+(T[i'+1,j',c']+1)+T[j'+1,j,c]$ は，他のテーブルに比べて計算時間が大きくなる．しかし，これは $T[i,j',c]+T[j'+1,j,c]$ と本質的に同値であり，元と比べてパラメータが減るため計算時間が n 減る．よって，計算時間を $O(C^2n^3)$ にすることができる．このとき，記憶領域は 2 倍になるだけであり，本質的には変化しない．

また，4つ目のテーブルなどに現れる $T[i,j,c']+1$ は，頂点 i から頂点 j までの間にある全ての頂点 $(v_i,...,v_j)$ を色 c 以外の１色で塗るための最小手数のテーブル $\bar{T}[i,j,c] = \min_{c'\neq c} T[i,j,c']$ を作り，同時に更新･管理することで，計算時間が C 減る．テーブル $T[i,j,c]$ を更新する際，色 c を除く全ての色 c' に対し，$\bar{T}[i,j,c']$ との比較を行い，$T[i,j,c]$ の方が小さい場合，$\bar{T}[i,j,c']$ の値を $T[i,j,c]$ の値にすればよい．この演算は $O(C)$ 時間，$C(Cn^2)$ 領域で可能である．よって，上記のアルゴリズムで $T[i,j,c']$ を求める際には，$\bar{T}[i,j,c]$ の値を用いるだけでよい．この演算は定数時間で可能である．以上より，全体の計算時間を C 減らし，$O(Cn^3)$ 時間にすることができる．

更に，区間 $[i,j]$ が色 c で塗られているとき，次に変更すべき色はセル $i-1$ の色かセル $j+1$ の色のみである．この事実を用いて，計算時間，記憶領域を C 減らすことができる．ループ全体で，各区間に対して C 回調べているところを，今調べている区間 $[i,j]$ における $i-1$ および $j+1$ の色についてのみ調べればよい．記憶領域に関しても，各区間 $[i,j]$ に対して $O(n)$ 個の記憶領域を持たずに，i,j のそれぞれの初期状態での色の分の 2 個のみを保持していればよい．以上より，全体の計算時間を $O(n^3)$，記憶領域を $O(n^2)$ にすることができる．

cycle の場合，最初に指定した条件 $[i<j]$ が無くなり，$[i>j]$ の場合，途中で n 番目，1 番目の頂点を間に挟むことになる．しかし，計算時間，記憶領域ともに定数倍であり，本質的には変化しない．

以上から，長さ n の path 上における Free Flood Filling Game の最適化問題は，色の数に関わらず $O(n^3)$ 時間，$O(n^2)$ 領域で解ける． ■

2.4 caterpillar 上での NP 完全性の証明

caterpillar 上において，色の数に制限の無い Free Flood Filling Game の決定問題は NP 完全であることを示す.

証明. 以下の補題により示す.

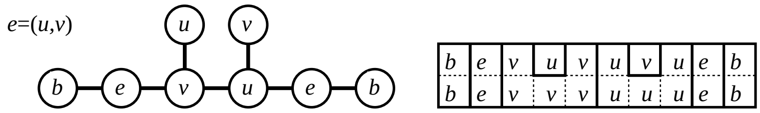
補題 11 *caterpillar* 上において，色の数に制限の無い *Free Flood Filling Game* の決定問題は NP 困難である.

証明. 有名な NP 完全問題である，頂点被覆問題から還元する.

問題 12 頂点被覆問題
入力. グラフ $G = (V, E)$，及び整数 k
出力. 全ての辺 $e = \{u, v\} \in E$ が，$e \cap S \neq \emptyset$ かつ $|S| = k$ であるような頂点部分集合 $S \subseteq V$ が存在するか

以下，$G = (V, E)$ 及び k を頂点被覆問題の入力とし，$n = |V|, m = |E|$ とする.

まず，頂点被覆問題のグラフ G における辺 $e_i = (u_i, v_i)$ に対応する caterpiller のガジェットを構成する．ガジェットの形状は，$\mathrm{path}(b_1, b_2, b_3, b_4, b_5, b_6)$ における頂点 b_3 に毛 h_3，頂点 b_4 に毛 h_4 がついたものである．それぞれの頂点の初期状態での色は，$col(b_1) = col(b_6) = b, col(b_2) = col(b_5) = e_i, col(b_3) = col(h_4) = u_i, col(b_4) = col(h_3) = v_i$ とする．このとき，b, e_i, u_i, v_i はそれぞれ異なる適当な色である．すなわち，グラフ G の各頂点に対応する色と，各辺に対応する色と，それら全てと違う色 b を用意することになる.



(a) gadget by a caterpillar (b) 2x*n* representation

図 2.4: 辺 $e = (u, v)$ に対応するガジェット

このガジェット上での Free Flood Filling Game を考えるとき，この 4 色の初期盤面から 1 手で盤面上の色を減らすことができないため，3 手で盤面を 1 色にすることは不可能であるが，頂点 h_3, h_4 の色 v_i, u_i のうち，どちらか一方を残すことで，残りの頂点は 3 手で色 b にすることができる．具体的には，頂点 h_4 の色 u_i を残す場合，彩色指令列 $(b_3, v_i), (b_3, e_i), (b_3, b)$ により，頂点 h_4 以外の頂点を色 b にすることができる.

このガジェットを辺の数 m だけ作り，辺 e_i に対応するガジェットの頂点 b_6 と，辺 e_{i+1} に対応するガジェットの頂点 b_1 をそれぞれ共有することで，グラフ G に対応する，caterpillar 上での Free Flood Filling Game となる．この caterpillar は多項式時間で作ることが可能である.

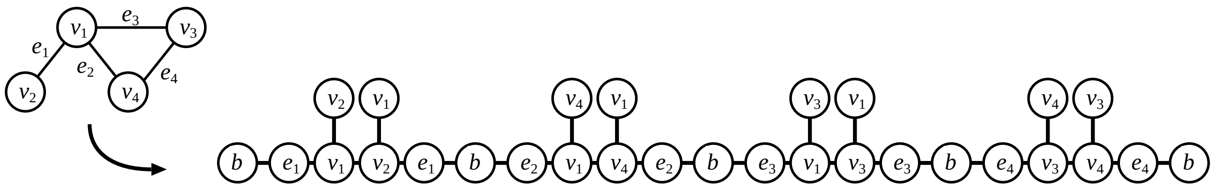


図 2.5: 頂点被覆問題に対応する caterpillar の例

次に，このガジェット上での Free Flood Filling Game を解くことは，グラフ G の頂点被覆問題を解くことと同値であること示す.

上記の例により，1 つのガジェットの背骨は 3 手で色 b にすることができる．ガジェットの数は $|E| = m$ 個であるため，caterpillar 全体の背骨は $3m$ 手で色 b にすることができる．背骨を塗り終えると，全ての色の同じ毛を 1 手で塗ることができるようになる．そのため，背骨を塗り終えた段階で，最も毛に残っている色の数が少なくなるようにすれば良い．このガジェットは，背骨を塗り終えたときに，対応する辺の端点に対応する色のどちらか一方のみが背骨に残るようになっている．このとき，caterpillar の毛に残っている色の数が最も少なくなるのは，ガジェットの彩色の際に，常にグラフ G の頂点被覆 S に含まれる頂点に対応する色を残してきた状態である．よって，caterpillar の毛を塗るのに必要な手数は $|S|$ 手である．背骨の分を足すと，$3m + |S|$ 手で塗れることになる.

以上より，頂点被覆問題は，caterpillar 上での Free Flood Filling Game へ還元できるため，caterpillar 上での色の数に制限の無い Free Flood Filling Game の決定問題は NP 困難である. ∎

補題 9，補題 11 より，caterpillar 上において，色の数に制限の無い Free Flood Filling Game の決定問題は NP 完全である. ∎

この証明のアイディアは，$2 \times n$ の盤面上での NP 完全性の証明と本質的に同様である [10].

2.5 proper interval graph 上での NP 完全性の証明

proper interval graph 上において，色の数に制限の無い Free Flood Filling Game の決定問題は NP 完全であることを示す.

証明. 以下の補題により示す.

補題 13 *proper interval graph* 上において，色の数に制限の無い *Free Flood Filling Game* の決定問題は NP 困難である.

証明. 有名な NP 完全問題である，頂点被覆問題から還元する．補題 11 と同様に，頂点被覆問題に対応する，proper interval graph 上での Free Flood Filling Game のガジェットを作る.

以下，$G=(V,E)$ 及び k を頂点被覆問題の入力とし，$n=|V|, m=|E|$ とする．

まず，頂点被覆問題のグラフ G に対応する proper interval graph のガジェットを構成する．ガジェットの interval representation $\mathcal{I}$ は，以下のようになる．

色の集合 C は，$n+m(m-1)+1$ 個のそれぞれ異なる色である，$V \cup \{w^i_j \mid 1 \le i \le m, 1 \le j \le m-1\} \cup \{b\}$ とする．

まず，背骨となる区間を置く．初期状態での色が b である $i+1$ 個の区間 $I_i (0 \le i \le m)$ を，$I_i=[4mi, 4mi+3]$ に配置する．次に，全ての辺に対して，各頂点に対応する区間を置く．全ての辺 $e_i=\{u_i,v_i\} \in E (1 \le i \le m)$ に対して，初期状態の色が $col(J_i)=u_i, col(J'_i)=v_i$ である２つの区間 J_i, J'_i を，$J_i=[4mi-2m, 4mi-2m+3], J'_i=[4mi-2m, 4mi-2m+3]$ に配置する．すなわち，J_i, J'_i は同じ位置の区間である．そして，背骨と頂点に対応する区間を繋ぐ区間を置く．全ての辺 $e_i=\{u_i,v_i\} \in E (1 \le i \le m)$ に対して，初期状態の色が $col(W^i_j)=w^i_j (1 \le j \le m-1)$ である $m-1$ 個の区間 $W^i_1, W^i_2, ..., W^i_{m-1}$ をそれぞれ２個ずつ作り，それぞれ $W^i_j=[4mi-2m-2j, 4mi-2m-2j+3], W'^i_j=[4mi-2m+2j, 4mi-2m+2j+3]$ に配置する．

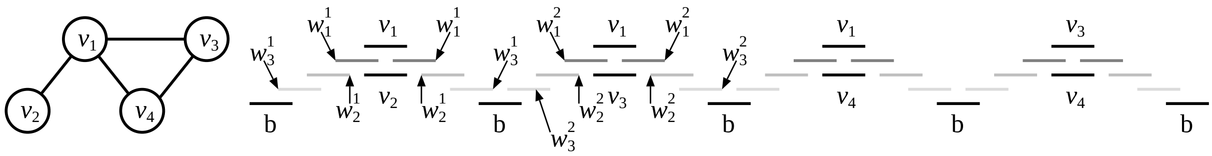


図 2.6: 頂点被覆問題に対応する interval representation の例

これは，グラフ上では，初期状態での色が $col(u_i)=u_i, col(v_i)+v_i$ である直接連結した頂点 u_i, v_i と，初期状態での色が $col(b_{i-1})=b$ である頂点 b_{i-1} が，初期状態での色が $col(w^i_j)=w^i_j$ である $m-1$ 個の頂点からなる path$(w^i_1, w^i_2, ..., w^i_{m-1})$ で，u_i, v_i と初期状態での色が $col(b_i)=b$ である頂点 b_i が，初期状態での色が $col(w'^i_j)=w^i_j$ である，$m-1$ 個の頂点からなる path$(w'^i_1, w'^i_2, ..., w'^i_{m-1})$ で連結された状態である．頂点 b_{i-1} から頂点 b_i までが，頂点被覆問題の辺 e_i に対応するガジェットである．すなわち，辺 e_i に対応するガジェットの頂点 b_i と，辺 e_{i+1} に対応するガジェットの頂点 $b_{(i+1)-1}$ は共通の頂点であり，interval representation $\mathcal{I}$ 上においても共通の区間である．

interval representation $\mathcal{I}$ は proper の条件を満たしている．以上により，頂点被覆問題のグラフ G に対応する proper interval graph の全体のガジェットが構成される．このガジェットは多項式時間で作ることが可能である．

次に，このガジェット上での Free Flood Filling Game を解くことは，グラフ G の頂点被覆問題を解くことと同値であること示す．

初期状態での色が w^i_j である頂点は，辺 e_i に対応するガジェット内に２つ存在し，他のガジェットには存在しない．この２つの頂点は，ガジェットの中央部である頂点 u_i, v_i 側から塗ることで，同時に彩色可能であり，外側からでは同時に彩色することはできない．これらの頂点は，１つの辺に対するガジェット内に $m-1$ 組存在するため，たとえ辺 e_i に対応するガジェットの頂点 u_i または頂点 v_i の色と辺 e_{i+1} に対応するガジェットの頂点 u_{i+1}

または v_{i+1} に同じ色があったとしても，それを同時に彩色するためには，$m-1$ 手の余分な手が必要になる．よって，このガジェットは，それぞれの辺に対応する部分を内側から塗るのが最短の手である．

このとき，m 手で頂点 u_i または頂点 v_i を残して，ガジェット 1 つを色 b にすることができる．残した頂点は，先にガジェット全体の色を統一することで，同じ色の頂点を幾つでも同時に塗ることができるが，背骨になる頂点 b_i は 3 つ以上同時に連結させることができない．よって，先に背骨を同じ色にしてしまうのが最短の手である．たとえば，頂点 u_i の色 u_i を残す場合，彩色指令列 $(v_i, w_1^i), (v_i, w_2^i), ..., (v_i, w_{m-1}^i), (v_i, b)$ により，頂点 u_i 以外の頂点を m 手で色 b にすることができる．このガジェットが m 個存在するため，全体で m^2 手で背骨を色 b にすることができる．

上記の例により，1 つのガジェットの背骨は m 手で色 b にすることができる．ガジェットの数は $|E| = m$ 個であるため，proper interval graph 全体の背骨は m^2 手で色 b にすることができる．背骨を塗り終えると，全ての色の同じ毛を 1 手で塗ることができるようになる．そのため，背骨を塗り終えた段階で，最も毛に残っている色の数が少なくなるようにすれば良い．このガジェットは，背骨を最短で塗り終えたときに，対応する辺の端点に対応する色のどちらか一方のみが背骨に残るようになっている．このとき，全体に残っている色の数が最も少なくなるのは，ガジェットの彩色の際に，常にグラフ G の頂点被覆 S に含まれる頂点に対応する色を残してきた状態である．よって，残りの頂点を塗るのに必要な手数は $|S|$ 手である．背骨の分を足すと，$m^2 + |S|$ 手で塗れることになる．

以上より，頂点被覆問題は，proper interval graph 上での Free Flood Filling Game へ還元できるため，proper interval graph 上での色の数に制限の無い Free Flood Filling Game の決定問題は NP 困難である． ∎

補題 9，補題 13 より，proper interval graph 上において，色の数に制限の無い Free Flood Filling Game の決定問題は NP 完全である． ∎

2.6 split graph 上での NP 完全性の証明

split graph 上において，色の数に制限の無い Free Flood Filling Game の決定問題は NP 完全であることを示す．

証明． 以下の補題により示す．

補題 14 *split graph* 上において，色の数に制限の無い *Free Flood Filling Game* の決定問題は NP 困難である．

証明． 既存の結果である，fixed，split graph に対して，色の数に制限が無い場合 NP 完全である [4] ことから還元する．

fixed の場合の元のグラフ $G = (V, E)$ は，clique$K$ と独立頂点集合 I で構成されている．K に含まれる頂点を全て色ごとコピーし，I' とする．fixed な頂点 (root) が K に含まれて

いる場合 root に，I に含まれている場合 root と接続されている K に含まれている頂点に，I' の頂点を全て接続する．こうして作成されたグラフ G' もまた，split graph である．

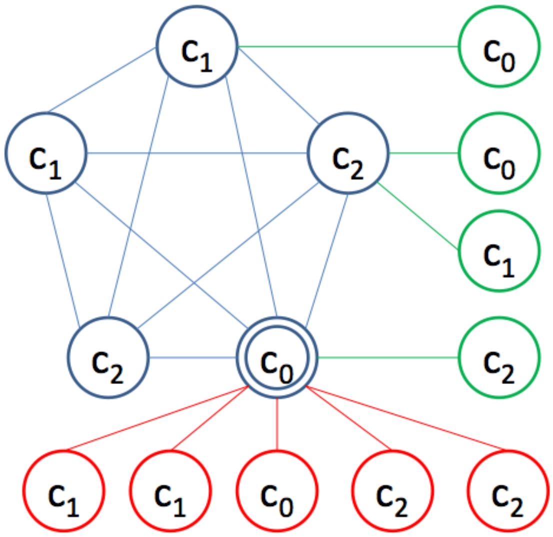


図 2.7: I' を接続した split graph

split graph の Free Flood Filling Game を解く際には，まず clique を 1 色にし，その後独立頂点集合を塗るのが最小の手数である．独立頂点集合の頂点を塗るとき，同じ色は 1 手で塗り終えたいため，先に clique を塗り終えることは，手数のロスにはならないためである．clique を塗り終えたとき，独立頂点集合に含まれる色の数を最小にする塗り方が，最小の手数である．

このとき，clique$K$ は，全ての頂点が連結されているため，彩色のとき clique のどの頂点を選択しても彩色に必要な手数は変わらず，高々$|K|-1$ 手で 1 色にできる．このとき，I' の頂点の色は，clique を 1 色にする際に，少なくとも 1 度選択されている．そのため，彩色の際に常に I' を連結した頂点を選択し続けていれば，clique が 1 色になったとき，I' の全ての頂点も clique と同じ色に塗り終えられている．もし，このとき I' を連結した頂点以外を選択した場合，clique を塗り終えた際，I' に未彩色の頂点が残る場合があり，手数のロスしか生まない．残らなかった場合も，手数の得にはならない．

独立頂点集合 I に含まれる全ての頂点は clique に連結しているため，1 色になった clique のどの頂点を選択しても必要な手数は変わらず，高々$|I|$ 手で 1 色にできる．このとき，もし I に含まれる頂点を選択した場合，選択した頂点を同じ色の頂点が I に含まれていた場合，手数のロスしか生まず，また同じ色の頂点が存在しなかった場合も，手数の得にはならない．

したがって，全ての盤面において，I' を連結した頂点を選択し続ければよいため，彩色手数の最適化問題は，元の fixed の問題と本質的に同じである． ∎

補題 9，補題 14 より，split graph 上において，色の数に制限の無い Free Flood Filling Game の決定問題は NP 完全である． ∎

2.7 色数制限ありの split graph 上での多項式時間アルゴリズム

split graph 上において，色の数が k 色以下の Free Flood Filling Game の最適化問題は，$O((k!)^2+n)$ 時間で解けることを示す.

証明． 動的計画法を用いて，多項式時間アルゴリズムを構成する．

補題 14 で述べたように，split graph の Free Flood Filling Game を解く際には，まず clique を 1 色にし，その後独立頂点集合を塗るのが最小の手数である．

色の数が k 色以下の場合，clique に含まれる色の数は k 色以下なので，clique は高々 $k-1$ 手で塗ることができる．これを全ての塗り方について試した場合，塗り方は全部で高々$(k-1)!$ 通りである．また，その後の独立頂点を塗る際にも，高々$k-1$ 手で塗ることができる．これも塗り方は全部で高々$(k-1)!$ 通りである．すなわち，全体での塗り方は，全部で高々$((k-1)!)^2$ 通りである．そして，これらの頂点を実際に塗るのにかかる時間は，高々$n-1$ 回である．

以上から，頂点数が n，色の数が k 色以下の時，$O((k!)^2+n)$ 時間で解ける． ∎

第3章　おわりに

3.1　本研究でわかったこと

Free Flood Filling Game の計算複雑性について，以下のことが分かった．

- 3 色以上で塗られた tree に対する Free Flood Filling Game の決定問題は，NP 完全である．
- path 上における Free Flood Filling Game の最適化問題は，色の数に関わらず多項式時間で解ける．
- 色の数に制限が無い場合，Free Flood Filling Game の決定問題は，caterpillar, proper interval graph, split graph 上でも NP 完全となる．
- 色の数に制限を加えると，Free Flood Filling Game の最適化問題は，split graph 上でも多項式時間で解ける．

3.2　考察及び既存の結果との関係性

本研究では，定理 4 と定理 5 によって，色の数に制限の無い Free Flood Filling Game において，path/cycle と caterpillar で計算量クラスが異なることを示した．path/cycle は，グラフの頂点の次数が全て 2 以下である．一方 caterpillar は，最大次数 3 以上の頂点が存在し，値が小さい程グラフが path に近い形をしていることを表す尺度でもある path 幅が 1 である．すなわち，caterpillar は，path/cycle に非常に近いグラフでありながら，計算量クラスが異なるという結果が得られたのである．これは非常にタイトな結果であり，free で色の数に制限の無い場合についての，問題の難しくなる境界を正確に求めたと言える．

split graph の結果を見ると，色の数に制限が無い Fixed Flood Filling Game が NP 困難である [4] という既存結果に対し，本研究では，色の数に制限のある Free Flood Filling Game は多項式時間で解けるという結果が得られた．一方 path は，色の数に制限が無い Fixed Flood Filling Game が $O(n^2)$ で解ける [9] という既存結果に対し，本研究では，色の数に制限のある Free Flood Filling Game は $O(n^3)$ で解けるという結果が得られた．このことから，特別簡単な構造のグラフクラスを除き，「色の数に制限が無い Fixed Flood

グラフクラス	fixed	fixed, k 色以下
一般のグラフ	NP-C	NP-C ($k \geq 3$) [1] P ($k \leq 2$) (自明)
(□/△/hex.) grid	NP-C	NP-C ($k \geq 3$) [9]
tree	NP-C [4]	NP-C ($k \geq 3$) [4]
split graph	NP-C [4]	P [4]
co-comparability graph	P [4]	P [4]
interval graph	P [6]	$O(4^k k^2 n^3)$ [6]
proper interval graph	P [6]	$O(4^k k^2 n^3)$ [6]
caterpillar	P [6]	$O(4^k k^2 n^3)$ [6]
path/cycle	$O(n^2)$ [9]	$O(n^2)$ [9]

表 3.1: 各グラフクラスに対する Fixed Flood Filling Game の計算時間

Filling Game は，色の数に制限のある Free Flood Filling Game よりも難しい」という推測が可能である．すなわち，fixed/free の違いより，色の数の制限の有無の方が、計算量に与える影響が大きいのではないかと思われる．

また，既存研究や関連研究によると，caterpillar や interval gprah では，fixed であったり，free でも色の数が制限されている場合は多項式時間で解けることもわかっている．free で色の数に制限が無いことは，非常に問題を難しくすることがわかった．

今後は，free で色の数に制限のある場合や，fixed の場合に関しても，問題の難しくなるよりタイトな境界を求める事が課題である．また，必ずしも interval represetation を持たないグラフや，２人ゲームへの拡張なども課題である．

最後に，fixed, free それぞれについて，既存結果と本研究での結果を整理した表を掲載する．

グラフクラス	free	free, k 色以下
一般のグラフ	NP-C	NP-C ($k \geq 3$) [1] P ($k \leq 2$) [8, 9]
(□/△/hex.) grid	NP-C	NP-C ($k \geq 3$) [9]
tree	NP-C (本稿)	NP-C ($k \geq 3$) (本稿)
split graph	NP-C (本稿)	$O((k!)^2 + n)$ (本稿)
interval graph	NP-C (本稿)	$O(4^k k^2 n^3)$ [6]
proper interval graph	NP-C (本稿)	$O(4^k k^2 n^3)$ [6]
caterpillar	NP-C (本稿)	$O(4^k k^2 n^3)$ [6]
path/cycle	$O(n^3)$(本稿)	$O(n^3)$ (本稿)

表 3.2: 各グラフクラスに対する Free Flood Filling Game の計算時間

謝辞

本研究を進めるにあたり，日頃から御指導いただきました北陸先端科学技術大学院大学の上原隆平教授，大舘陽太助教には大変お世話になりましたことを，深く感謝いたします．

また，国立情報学研究所の宇野毅明准教授並びに大阪府立大学の宇野裕之准教授には，研究に関する有意義な議論をしていただきました．心より感謝いたします．

最後に，本学における生活を共に過ごし，生活等も支援していただいた，北陸先端科学技術大学院大学の浅野哲夫教授並びに浅野哲夫研究室，上原隆平研究室の皆様に，厚く御礼申し上げます．

参考文献

[1] D. Arthur, R. Clifford, M. Jalsenius, A. Montanaro, and B. Sach. The Complexity of Flood Filling Games. In *FUN 2010*, pages 307–318. Lecture Notes in Computer Science Vol. 6099, Springer-Verlag, 2010.

[2] K. P. Bogart and G. S. Lueker. A short proof that 'proper=unit'. *Discrete Mathematics*, 201:21–23, 1999.

[3] R. Clifford, M. Jalsenius, A. Montanaro, and B. Sach. The Complexity of Flood Filling Games. Theory of Computing System, DOI: 10.1007/s00224-011-9339-2, 2011.

[4] R. Fleischer and G. J. Woeginger. An Algorithmic Analysis of the Honey-Bee Game. In *FUN 2010*, pages 178–189. Lecture Notes in Computer Science Vol. 6099, Springer-Verlag, 2010.

[5] H. Fukui, A. Nakanishi, R. Uehara, T. Uno, and Y. Uno. the Complexity of Free Flood Filling Games. In *WAAC 2011*, pages 51–56, 2011.

[6] H. Fukui, Y. Otachi, R. Uehara, T. Uno, and Y. Uno. On Complexity of Flooding Games on Graphs with Interval Representations. arXiv:1206.6201v2, 2013.

[7] M. R. Garey and D. S. Johnson. *Computers and Intractability — A Guide to the Theory of NP-Completeness*. Freeman, 1979.

[8] A. Lagoutte. 2-Free-Flood-It is polynomial. Technical report, arXiv:1008.3091v1, 2010.

[9] A. Lagoutte, M. Naual, and E. Thierry. Flooding games on graphs. In *Latin-American Algorithms, Graphs and Optimization Symposium (LAGOS 2011)*, 2011.

[10] K. Meeks and A. Scott. The complexity of Free-Flood-It on $2 \times n$ boards. arXiv:11015518v1, Jan 2011.

[11] F. S. Roberts. Indifference graphs. In F. Harary, editor, *Proof Techniques in Graph Theory*, pages 139–146. Academic Press, 1969.