

Title	アンチスライドパズルの数理的特徴づけと計算量的複雑さ
Author(s)	南澤, 洸
Citation	
Issue Date	2021-12
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/17607
Rights	
Description	Supervisor: 金子 峰雄, 先端科学技術研究科, 修士(情報科学)

修士論文

アンチスライドパズルの数理的特徴づけと計算量的複雑さ

南澤 洸

主指導教員 金子 峰雄
審査委員主査 金子 峰雄
審査委員副査 上原 隆平
藤崎 英一郎
廣川 直

北陸先端科学技術大学院大学
先端科学技術研究科
(情報科学系)

2021(令和3)年12月

概要

アンチスライドパズルとは、ピースの集合 \mathcal{P} とフレーム F が与えられた時 \mathcal{P} の全てのピースが動けない配置（アンチスライド配置）になるように F の中に \mathcal{P} をしまうことが目的のパズルである。この「しまう」操作をパッキングと呼ぶ。アンチスライドパズルとよく似たパズルにパッキングパズルがあるが、この2つのパズルは以下に示すように困難性の根拠が異なる。

計算機科学の世界で既によく研究されているパッキングパズル（パッキング問題）は、多くの場合フレーム F 内部の面積とピース集合 \mathcal{P} の総面積がほとんど等しくデザインされていて「パッキングに困難性があるパズル」である。一方で、アンチスライドパズルには「パッキングは自明にできる」という前提条件が置かれている。すなわち、アンチスライドパズルは「パッキングは簡単だが、アンチスライド配置を作ることが困難なパズル」である。

よって、アンチスライドパズルの困難性を証明する時には「パッキングができることは簡単に（多項式時間で）判定できる」という前提条件が明示的に表現されたガジェットをデザインする。

また、アンチスライドパズルの特殊形として、フレーム F が与えられずに、ピース集合 \mathcal{P} だけで相対的にアンチスライド状態を作るパズルをインターロックパズルと呼ぶ。

本稿ではまずはじめにアンチスライド状態を数理的に特徴づけるモデルを複数提案し、アンチスライドパズルやインターロックパズルの先行研究で提案されているモデルと比較する。本研究ではアンチスライドパズルのピース形状とピース配置の組合せとして以下の3つを研究対象とする：ポリオミノの直交配置・ポリオミノの一般配置・一般多角形の一般配置。ここでポリオミノとは、複数の単位正方形に対して、隣接する2つが単位辺を完全に共有するように並べて得られるシルエットのことをいう。多角形 P の直交配置とは、 P のすべての辺が x 軸か y 軸に平行な配置のことである。

本稿では以下の3つのアンチスライド性を定義した：弱アンチスライド・強アンチスライド・ポリオミノのアンチスライド。先行研究では、本稿で定義した言葉を使うと「ポリオミノの直交配置に対する弱アンチスライド性」のみを扱っている。このモデルを前提条件としてIPソルバやOBDDを使って、ピース数最小のアンチスライド配置の解析などを報告している。しかし、このモデルでは、実際のアンチスライドパズルに必要な情報（例えば重力）を考慮していないので、計算機が出力した解が現実的な解と必ずしも一致しない。したがって本稿では、重力などの物理的な情報を加味し、先行研究のモデルよりも現実的なアンチスライド性に近いモデルとして「強アンチスライド」を提案した。

「弱アンチスライド」・「強アンチスライド」は「ポリオミノの直交配置」・「ポリオミノの一般配置」・「一般多角形の一般配置」の全てに対して定義される。一方で本稿で定義した3つ目のアンチスライド性「ポリオミノのアンチスライド」は「ポリオミノの直交配置」に対してのみ定義される。

本稿ではモデル「ポリオミノのアンチスライド」上で、アンチスライドパズルの計算困難性を証明する。このとき、一番扱いやすいポリオミノの直交配置を前提条件とすることにより、「この問題は条件に縛りを入れた単純な場合においても難しい」といえる。

まず、与えられたピース配置 \mathcal{A} のアンチスライド性の判定問題を取り上げる。この問題を解くためにアンチスライドパズルの与えられたピース配置を有向グラフで表現する。次に、「与えられたピース配置のアンチスライド性」と「そのピース配置を表現する有向グラフの強連結性」が同値であることを証明する。先行研究により、与えられた有向グラフの強連結性の判定問題は線形時間で解ける。よって「与えられたピース配置のアンチスライド性の判定問題は多項式時間で解ける」ということが証明できる。

次に、入力としてピース集合 \mathcal{P} とフレーム F が与えられたときにアンチスライド性を達成する配置が存在するかどうかを判定する問題を考える。以下に示す手順でアンチスライドパズルの NP 完全性を証明する。与えられたピース配置 \mathcal{A} のアンチスライド性は多項式時間で判定できるので、アンチスライドパズルの困難性は計算量クラス NP に入ることが示される。次は、NP 完全問題である 3-Partition 問題からの還元で NP 困難性を示し、アンチスライドパズルが NP 完全問題であることの証明が完了する。さらに、アンチスライドパズルのフレームを変形することにより、この結果をインターロックパズルへも拡張できることを示す。

最後に、与えられたピース集合 \mathcal{P} がインターロックできるための境界条件を解明する。すなわち、ピースが全て x 単調なときはインターロックする配置が存在するが、ピースが全て凸多角形の時は、いかなるピース配置に対してもインターロックできないことを証明する。

Abstract

For a given set \mathcal{P} of pieces and a frame F , an anti-slide puzzle asks us to arrange the pieces so that none of the pieces can slide (an anti-slide arrangement) in the frame F . This “putting in” operation is called packing. Packing puzzles are similar to anti-slide puzzles, but these two puzzles have different reasons for hardness.

Packing puzzles (packing problems) have been well investigated in computer science. Packing puzzles are often designed so that the area inside the frame F and the total area of the pieces in the set \mathcal{P} are almost equal. Thus, “the hardness of packing puzzles lies in the packing”. On the other hand, anti-slide puzzles are based on the following condition “packing is trivial to solve”. Thus, anti-slide puzzles are required to have the following properties “easy to pack, but hard to make an anti-slide arrangement”.

Therefore, when proving the hardness of an anti-slide puzzle, we have to design gadgets that explicitly express the properties that “it is easy to check (in polynomial time) that packing can be done”.

The puzzles that make a relative anti-slide arrangement with only a set of pieces \mathcal{P} without a frame F is called interlock puzzles. Interlock puzzles are a special form of anti-slide puzzles.

In this paper, We first propose three models that mathematically characterize the anti-slide states and compare them with the models proposed in previous research of anti-slide and interlock puzzles. In this research of anti-slide puzzles, we consider three possible combinations of piece shapes and piece arrangements: “orthogonal arrangements of polyominoes”, “general arrangements of polyominoes”, and “general arrangements of general polygons”. Here, a polyomino refers to the silhouette obtained from multiple unit squares by arranging them so that two adjacent squares totally share a unit edge. A polygon P is orthogonal when all edges of P are parallel to the x - or y -axes.

There are three types of anti-sliding properties defined in this paper: “weak anti-slide”, “strong anti-slide”, and “anti-slide for polyominoes”. Previous work has dealt only with “weak anti-slide properties for orthogonal arrangements of polyominoes” in terms of our paper. They reported analyse of anti-slide arrangements with the minimum number of pieces, using the tools IP solver and OBDD, under this model. However, this model does not take into account some information (e.g. gravity) necessary for real anti-slide puzzles. Therefore, the solution output by the computer does not always match the realistic solution. In this paper, we proposed the “strong anti-slide” as a model that is closer to the realistic anti-slide property than the models in previous research, taking into account physical information such as gravity.

“Weak anti-slide” and “strong anti-slide” are defined for all of “orthogonal arrangements of polyominoes”, “general arrangements of polyominoes” and “general arrangements of polygons”. On the other hand, the third anti-slide property defined in this paper, “anti-slide for polyominoes”, is defined only for “orthogonal arrangements of

polyominoes”.

In this paper, we investigate the computational complexities of the anti-slide puzzles on the model “anti-slide for polyominoes”. Based on the following condition the orthogonal arrangements of polyominoes, which is the simplest, we can see that “the problem is hard even in the restricted case with bounded conditions”.

First, we consider the problem of determining whether a given piece arrangement \mathcal{A} is anti-slide or not. To solve this problem, we represent a given arrangement \mathcal{A} of pieces in an anti-slide puzzle as a directed graph. Next, we prove that “the problem of determining whether a given piece arrangement \mathcal{A} is anti-slide or not” and “the problem of determining whether the directed graph representation of the piece arrangement is strongly connected or not” are equivalent. The problem of determining the strong connectivity of a given directed graph can be solved in linear time. Thus, we can prove that “the problem of determining whether a given piece arrangement is anti-slide or not can be solved in polynomial time”.

Next, for a given set \mathcal{P} of pieces and a frame F , we consider the problem of determining whether there exists an arrangement that achieves anti-slide or not. We prove the NP-completeness of anti-slide puzzles using the following procedure. Since it is possible to determine in polynomial time whether a given piece arrangement \mathcal{A} is anti-slide or not, it can be shown that the hardness of anti-slide puzzles falls into the computational complexity class NP. Next, we show the NP-hardness reduction from the 3-Partition problem, which is an NP-complete problem. We complete the proof that anti-slide puzzles is an NP-complete problem.

Finally, for a given set \mathcal{P} of pieces, we clarify the boundary conditions for interlocking. That is, we prove that there exists an interlocking arrangement even if all pieces are x -monotone, but that it is impossible to interlock for any piece arrangements when all pieces are convex polygons.

目次

第1章	序論	1
1.1	研究の背景	1
1.1.1	アンチスライドパズルの周辺のパズル	1
1.2	先行研究	2
1.2.1	アンチスライドパズルの周辺のパズルの先行研究	2
1.2.2	アンチスライドパズルの紹介とその先行研究	4
1.3	研究の動機・目的・意義	6
1.4	本稿の構成	8
第2章	準備	9
2.1	グラフ理論	9
2.2	アルゴリズム理論	11
2.3	離散・計算幾何学	11
2.4	計算量理論	13
2.5	本稿で扱う問題の定義	16
第3章	アンチスライドパズルの数理的特徴づけ	18
3.1	「支える」の定義	18
3.2	「アンチスライド性」の定義	20
3.3	「インターロック性」の定義	23
第4章	アンチスライド性判定問題の多項式時間アルゴリズム	26
4.1	ピース配置の有向グラフ表現と多項式時間判定アルゴリズム	26
第5章	アンチスライドパズルの困難性	30
第6章	インターロック可能な多角形の境界条件の解明	33
第7章	今後の課題	36
7.1	ポリオミノの直交配置	36
7.2	ポリオミノの一般配置	38
7.3	一般多角形の一般配置	40

外部発表	42
謝辞	43
参考文献	44

目次

1.1	タングラム (a) と清少納言知恵の板 (b) のピース分割	2
1.2	タングラムの問題例 [6]	3
1.3	アンチスライド配置の例	4
1.4	Anti-Slide(Wil Strijbos)[24]	5
1.5	Lock Device(山本浩)[26]	5
1.6	方向 \vec{d} に対して, P は現実的にはアンチスライドとは呼べない例	7
2.1	ペントミノ	12
2.2	x 単調なポリオミノ	12
3.1	Q が P を下から支えている場合	19
3.2	不等式 $x_d(q_{e'-}) < x_d(p) < x_d(q_{e'+})$ を満たす例, 満たさない例	20
3.3	多角形 P の下側エンベロップ	20
3.4	重心とバランス点	21
3.5	\vec{d} を下向きとしたときのピース P のアンチスライド性	22
3.6	(a): インターロックしていない例, (b): インターロックしている例	24
3.7	(a): ピース配置 \mathcal{A} , (b): ピースを融合して得られる配置 \mathcal{A}'	24
3.8	インターロック状態の構成手順の例	24
4.1	インターロックしていない例. (a): 配置 \mathcal{A} , (b): $D_S(\mathcal{A})$, (c): $D_E(\mathcal{A})$	27
4.2	インターロックしている例. (a): 配置 \mathcal{A} , (b): $D_S(\mathcal{A})$, (c): $D_E(\mathcal{A})$	27
4.3	図1.3(b) の有向グラフ表現. (a): 配置 \mathcal{A} , (b): $D_S(\mathcal{A})$	29
5.1	(a) フレーム F , (b) 穴に詰めるピース P_i のサイズ	31
5.2	図5.1(a) の F を変形する.	32
6.1	配置 \mathcal{A} に対するコンタクトグラフの構成手順	35
6.2	T に沿ったサイクル	35
7.1	2次元版ジェンガ	38
7.2	座標軸に依存するアンチスライド性	39
7.3	Rotation モデル	39
7.4	Coordinate Motion モデル (アンチスライド)	40
7.5	Coordinate Motion モデル (インターロック)	41

第1章 序論

1.1 研究の背景

1.1.1 アンチスライドパズルの周辺のパズル

パズルは大きくペンシルパズルとメカニカルパズルの2つに分類できる。ペンシルパズルとは、紙と鉛筆があればできるパズルである [9]。メカニカルパズルとは、手で操作できる、何らかの専用の器具が必要とされるパズルである [11]。有名なパズルを例にあげると、数独とルービック・キューブがそれぞれ対応する。本稿のようにパズルを通して計算量理論を論じることは、計算量理論における計算量クラスの理解の助けになる。例えば「数独とルービック・キューブは困難性が本質的に異なる」ということを計算量理論を使って証明できる。数独は「解くことは困難（効率的に解くアルゴリズムが知られていない）だが、解の正当性を検証することは容易である」という **NP** 完全性という特徴をもつ。一方でルービック・キューブは、（一般的に、解くことは困難であるが）解の検証は簡単であり、かつ解くための手順（アルゴリズム）が知られている。よって、計算量理論的な意味で、ルービック・キューブは数独に比べて簡単であるといえる [12]。

メカニカルパズルの分類の例として、シルエットパズルとパッキングパズルと組木がある。

シルエットパズルとは、与えられたピース集合 \mathcal{P} を平面的に並び替えて目的のシルエット S を作るパズルである。ピースを立体的に組む作品（例えばソーマキューブ [11]）は3次元的なシルエットパズルとみなせるが、本稿の中ではシルエットパズルというと2次元的な作品を指すものとする [14][15]。

パッキングパズルとは、与えられたピース集合 \mathcal{P} をすべて、与えられたフレーム F の中にしまうことが目的のパズルである [13]。本稿で扱うアンチスライドパズルはパッキングパズルと似た特徴を持つメカニカルパズルである。

組木とは、与えられたピースを技巧的に組むことによって1つの立体を作るパズルである。本研究を先に進めていくと関連でてくるものは「すべてのピースを同時に動かす事によって分解する・組む」という、Coordinate Motion[27]（とパズル業界で）呼ばれるテクニックを使う組木である。例えば Exploding Ball[30] や Kaleidos[28] がそれにあたる。Coordinate Motion を使う組木は（発表されている作品の多くは）対称性の高い作品であり、完成形が正十二面体や菱形十二面体などにデザインされているものもある。Exploding Ball は George Bell と Stephen Chin

が2012年に発表した作品である。似た作品に、Stephen Chinが2010年に発表した作品1 Pinko Ringo[29]もある。これらは球体にデザインされている。球体を回転させることによって発生する遠心力を利用して球体を分解させる、という作品である。George Bellが自身のYouTubeチャンネルでパズルを分解する様子を公開している[30]。これを観ればCoordinate Motionという動きについて理解の助けになる。一方で、Philippe Duboisが1980年代に発表したとされているKaleidosという作品は、単純な仕組みに見えるが技巧的なCoordinate Motionを使った組木である。この作品はペンローズ三角形[8]の平面図のようにデザインされている。

折り紙工学の研究においても、全体が同時に折り畳まれるという仕組みが度々採用されている[31]。これもCoordinate Motionと似たものであるので、これらの動画を観てもCoordinate Motionという動きについて理解の助けになる。

本稿はアンチスライドパズルについての研究であり、Coordinate Motionとは一般には関連が薄い。しかし、アンチスライドパズルの研究をさらに進めていくとCoordinate Motionというテクニックを考慮しなければならない場合が出てくる。本稿でこれは扱わないので、本研究とCoordinate Motionとの関連性については最後に7章で紹介する。

1.2 先行研究

1.2.1 アンチスライドパズルの周辺のパズルの先行研究

特によく知られたシルエットパズルとして、正方形を7つの多角形に分割したピースを使うタングラム[6]や、タングラムとは違う7つの多角形に分割する、日本生まれのシルエットパズル清少納言知恵の板がある(図1.1)。タングラムのピースの集合 \mathcal{P} を図1.1に、シルエット S の問題例を図1.2に示す。このパズルについて、与えられたピースで作ることができるシルエットの総数についての研究がある[16][14]。問題例(図1.2)をみれば分かるとおり、シルエット S は無限に作れる。これらのうち特に「凸配置」の総数を数えた研究がある。この研究によると、タングラムの凸配置は13種類、清少納言知恵の板の凸配置は16種類である。

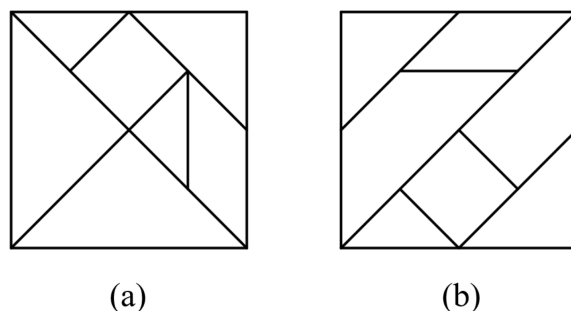


図 1.1: タングラム (a) と清少納言知恵の板 (b) のピース分割

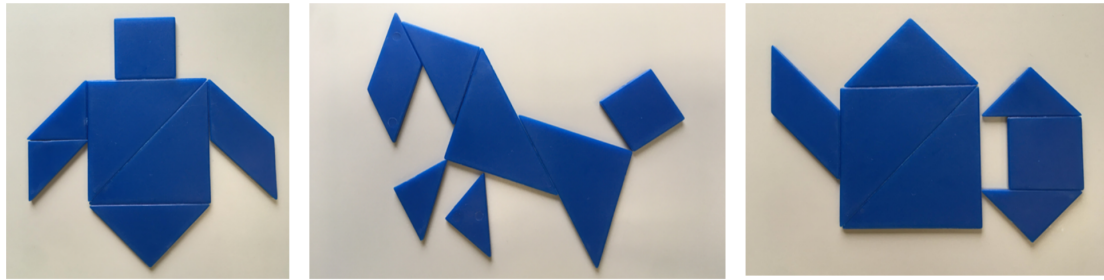


図 1.2: タングラムの問題例 [6]

近年は目的とするシルエット S が具体的には与えられずに、目的とするシルエットの条件のみが与えられるシルエットパズルが考案され、研究されている。例えば「線対称なシルエットを作れ」という線対称形パズルの計算複雑性を証明した研究 [17] と「シルエットの外形と内側にできる穴の形を相似形にせよ」という内外相似パズルの計算複雑性を証明した研究 [15] の 2 つがある。本稿で取り上げるアンチスライドパズルも、こうした「ゴールの性質は与えられるが、ゴールの形が明確に与えられない」という独特の難しさをもつパズルの 1 つである [10]。線対称形パズルの流れをくんだ 2 次元版アンチスライドパズルとして、Vladimir Krasnoukhov の Stop-Puzzle がある [25]。

パッキングパズルの先行研究の一部を紹介する。ドミノ（単位正方形を 2 個つなげた形状、 1×2 のポリオミノ）を多角形にパッキングする問題は多項式時間で解ける [20]。トロミノ（単位正方形を 3 個つなげた形状のポリオミノ。これは L 型と I 型の 2 種類あり、ピースとしてそれぞれ単独で使う問題を考える）を多角形にパッキングする問題は **NP** 完全である [21]。単位正方形を 4 個つなげた形状のうち 2×2 のテトロミノをピースとして使うパッキング問題も **NP** 完全である [22]。以上が 2 次元パッキング問題の困難性を証明した先行研究の一部である。本稿で扱う問題も 2 次元に限定したアンチスライドパズルの困難性の証明である。パッキング問題は長く研究されているので、3 次元以上の高次元に拡張した問題に対しても困難性の証明をした研究がある [23]。

文献 [19] ではもう少し一般化した問題として、長方形のパッキング問題の困難性を証明している。すなわち、正整数の多重集合 $\hat{A} = \{x_1, x_2, x_3, \dots, x_n\}$ に対して長方形ピースの集合 $\mathcal{P} = \{1 \times x_1, 1 \times x_2, \dots, 1 \times x_n\}$ を面積 $x_1 + x_2 + \dots + x_n$ のフレーム F にパッキングする問題は **NP** 完全であることを証明している。また、同じくこの論文では、正方形ピースの集合 $\mathcal{P} = \{x_1 \times x_1, x_2 \times x_2, \dots, x_n \times x_n\}$ を面積 $x_1^2 + x_2^2 + \dots + x_n^2$ のフレーム F にパッキングする問題は **NP** 完全であることも証明している。

1.2.2 アンチスライドパズルの紹介とその先行研究

アンチスライドパズルとは、ピースの集合 \mathcal{P} とフレーム F が与えられた時「 \mathcal{P} のすべてのピースが動けない配置になる」ように F の中にしまうことが目的のパズルである。図1.3にアンチスライド配置の例を示す。ここで、灰色の長方形ピース4つをピースの集合 \mathcal{P} とし、黒色の穴の空いた多角形をフレーム F とする。配置 (a) はアンチスライド配置ではなく、配置 (b) はアンチスライド配置である。この2つの違いを直観的に判定するならば「フレームを傾けたときにどのピースも動かないかどうか」を確かめれば良い。

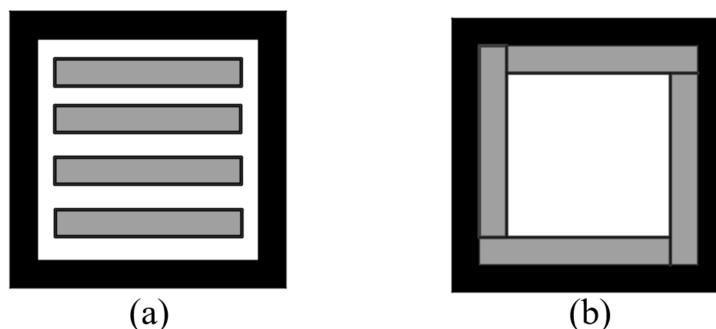


図 1.3: アンチスライド配置の例

アンチスライドパズルの元祖と呼べるものは、1994年にパズル作家 William Strijbs が考案し、2007年に「明治サイコロキャラメルパズル」(図1.4)として市販されたものである [24]。これは3次元版のアンチスライドパズルであり、 $4 \times 4 \times 4$ のケースの中に、 $1 \times 2 \times 2$ のピースをそれぞれ、15個、14個、12個、13個詰めてスライドしないようにするというパズルである。(難易度はこの順番で高くなる。また、ピースを斜めに詰める必要はない。)その後パズル業界では、特に2次元平面上で、様々なフレームと様々なピース集合によるアンチスライドパズルが数多く提案されている。例えば、Vladimir Krasnoukhov の Stop-Puzzle がある [25]。

本稿では2012年にパズル作家の山本浩氏が考案した Lock Device (図1.5) を特徴的なパズルとして紹介する [26]。このパズルでは、与えられるものはピース集合 \mathcal{P} だけであり、フレームが存在しないアンチスライドパズルである。本稿ではこれをインターロックパズルと呼ぶ。ピースはすべて x 単調で、それぞれが互いにロックして、どのピースも動かず、バラバラにならないようにすることがパズルの目的である。このパズルを計算量理論の研究として扱うのは本研究が初めてである。



図 1.4: Anti-Slide(Wil Strijbos)[24]

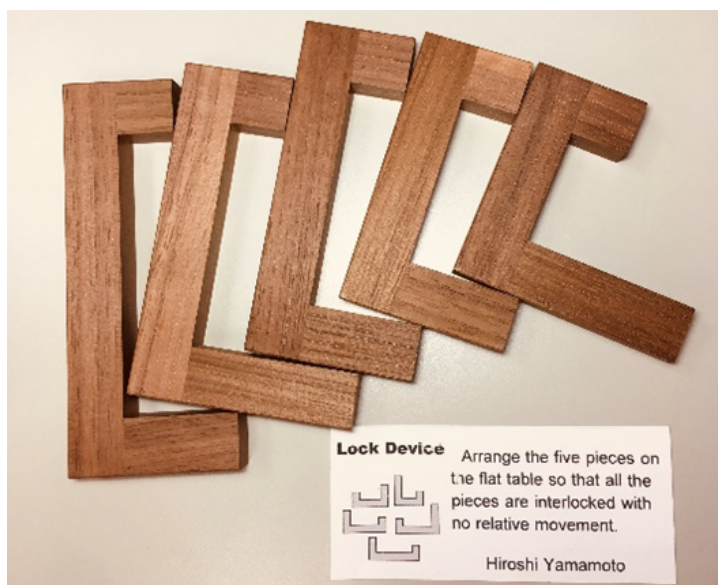


図 1.5: Lock Device(山本浩)[26]

3次元アンチスライドパズルに対して、2015年に天野らはIPソルバを用いて、ピース数最小のアンチスライド配置のピースの数を求める研究をした [1]。これは Strijbos のパズル (図1.4) を一般化した問題、つまり $2 \times 2 \times 1$ のピースの集合を $\ell \times m \times n$ のケースにいれる問題を扱っている。

一方2019年には、武永ら [3] が2次元のアンチスライドパズルの研究成果を報告した [3]。この研究では、ピースがペントミノの場合を定式化し、OBDDを用い

て解の列挙問題を解いた。しかし、これらの先行研究ではかなり限定的な問題を扱っており、アンチスライドという性質を一般の多角形上で定式化しているわけではない。そこで本稿ではまず、アンチスライドという性質に対する数理的なモデルを与える。この定式化はいくつかの段階に別れていて、それぞれのモデルに対して、既存の研究や自然なアンチスライドパズルを対応付けることができる。

2021年に天野らは2015年に行った3次元アンチスライドパズルの手法を使って、2次元アンチスライドパズルの解析をした[2]。この論文ではピースにT型テトロミノ（合同な正方形を4つ辺接触させたシルエット）を使ってピース数最小の2次元アンチスライド配置のピース数を求めている。

パズルのインターロック性判定問題を扱った先行研究[18]がある。しかし、この論文で定義しているインターロック性は、アンチスライド性を元に定義したものではないので、本稿で定義するアンチスライド性とは異なる概念であることに注意が必要である。[18]におけるインターロックの定義も本稿においても同様に「ピースの任意の部分集合に対して、それらが（十分遠くに）分離するものはない」としているが、この「分離」に関して[18, Figure 2]では「ピース同士が多少動いてもよいが完全に分離しないとき」としている。一方で本稿ではアンチスライド性の発展としてインターロック性を定義するので「どのピースも完全に動いてはいけない」とする。よって、[18, Corollary 3]の「多角形のインターロック性判定問題は **PSPACE** 完全問題である」という主張は、本稿とは違うインターロック性判定のモデルに関するものである。

1.3 研究の動機・目的・意義

アンチスライドパズルの先行研究は3報 ([1], [2], [3]) があるが、このうちどれも「アンチスライド」という現象に数理的に厳密な定義を与えておらず、アンチスライドパズルの計算困難性の議論もなされていない。またこれらの論文の中で定義されているアンチスライドモデルはとても直観的なものである。例えば[2]では、アンチスライドパッキングを「安定」と呼び「安定なパッキングとは、任意の方向に対して、どのピースも動けないパッキング配置である。」と定義している。[3]では、ピース P の下に Q が存在すれば（どんな配置かに関わらず）アンチスライドであるとしている。よって、図1.6に示すピース配置に対しても、方向 \vec{d} に対して、多角形 P は多角形 Q に対してアンチスライドであると判定されてしまう。このモデルは計算機上では扱いやすい。このモデルを本稿では「弱アンチスライド」と定義する。しかし、このモデルを使って計算機が出力した解と、現実的なアンチスライドパズルの解は必ずしも一致しない。よって、本研究の動機として、このモデルとは別に、現実のアンチスライドパズルに近い数理モデルを提案することがあげられる。そこで、本稿では重力を加味して、図1.6示すピース配置はアンチスライドではないと判定する2つめのモデル「強アンチスライド」を提案する。

また、本研究の目的として、先行研究にはない結果である、アンチスライドパズルの困難性を証明するということがあげられる。この証明をするときの条件としてポリオミノの直交配置を扱うために、3つめのモデル「ポリオミノのアンチスライド」を提案する。困難性（計算量の下限）を示すためには問題の本質をみきわめる必要があり、それが新しいアルゴリズムの発見への手がかりとなる可能性が高い [40]。よって、本稿でアンチスライドパズルの困難性を証明することは、計算量理論の発展のために意義がある。

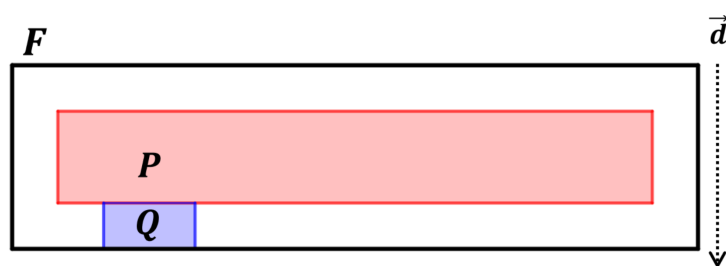


図 1.6: 方向 \vec{d} に対して、 P は現実的にはアンチスライドとは呼べない例

次に、パズル研究の応用事例を紹介することにより、本研究の意義を述べる。例えばシルエットパズルは食器のデザインに使われることもある [16]。パッキングパズルはその名の通り、商品の梱包作業に応用できる。このときの荷物の配置を「パッキング配置」と呼ぶことにする。ここで、アンチスライドパズルで行う「アンチスライド配置」とは、そのパッキング配置においてアンチスライド性を保証した配置のことである。実用的な応用を考えるとアンチスライド配置（特に、弱アンチスライド性だけではなく、強アンチスライド性も満たす配置）の方が通常のパッキング配置よりも安全性の面で優れているといえる。商品が箱に隙間なくパッキングされている場合はアンチスライドであるといえるが、一般にはそのように梱包できるとはいえない。上手い具合にパッキングすると詰めるために使う箱の数が減り、一度により多くの荷物をトラックに積み込めるので、全体として運送の回数が減って効率が良いといえるが、箱の中身のアンチスライド性が保証されていないと中身が動いて商品同士がぶつかり合ってしまうかもしれない。よって運送時にはパッキング性だけではなく、（強）アンチスライド性まで保証された配置がより安全であるといえる。箱の中に衝撃緩和剤を詰めればアンチスライド状態を作れるが、商品のみでアンチスライド状態を作ることができれば衝撃緩和剤は不要になるので、そういう意味でのコスト削減にもなる。

1.4 本稿の構成

1章では、本研究の動機と意義を述べ、アンチスライドパズルとその周辺のパズルについての先行研究を紹介する。

2章では、本稿で扱う知識、特に、グラフ理論、アルゴリズム理論、離散・計算幾何学、計算量理論の準備をして、「アンチスライドパズル」・「インターロックパズル」・「アンチスライド性判定問題」・「インターロック性判定問題」を定義をする。

3章では、先行研究で提案されているアンチスライドモデルを改善し、より現実的なパズルに近いモデルを複数定義する。

4章では、3章で定義したモデル上で、与えられたピース配置 \mathcal{A} のアンチスライド性判定問題は多項式時間で解けることを証明する。

5章では、アンチスライド配置を構成する問題の困難性を証明する。

6章では、アンチスライド性の特殊形であるインターロック性について、インターロック可能な多角形の境界条件を解明する。

7章では、本稿におけるアンチスライドパズル・インターロックパズルの研究で解明できなかったトピックを紹介する。

第2章 準備

2.1 グラフ理論

本稿で用いるグラフ理論の基礎事項について、文献 [38] より引用する。

有向グラフは、有限集合 V とその直積集合 $V \times V$ の部分集合 A の組 $D = (V, A)$ によって定義される。ここで、 $V = V(D)$ の元を D の頂点、 $A = A(D)$ の元を D の辺または弧と呼ぶ。

$$A(D) \subseteq V(D) \times V(D) = \{(u, v) \mid u, v \in V(D)\}$$

D の辺は2つの頂点 $u, v \in V(D)$ の順序対 (u, v) で表わす。このとき、 u を (u, v) の始点、 v を (u, v) の終点と呼ぶ。無向グラフの場合は辺を非順序対 $\{u, v\}$ で表わす。

有向グラフ D において、頂点 $u \in V(D)$ を始点、頂点 $v \in V(D)$ を終点とする有向歩道が存在するとき、 u から v へ到達可能であるといい、 u から v に至る有向道の長さの最小値 $d(u, v)$ を u から v までの距離と呼ぶ。また、 u から v へ到達可能でないときには、 $d(u, v) = \infty$ と定める。無向グラフのように $d(u, v) = d(v, u)$ になるとはかぎらない。さらに、 $d(u, v)$ が有限でも、 $d(v, u) = \infty$ となりえることに注意。

有向グラフ D のどの2頂点も半歩道で結ばれているとき、 D は弱連結であるという。これは D の辺の向きを無視して得られる無向グラフ (D の基礎グラフと呼ぶ) が連結であることに他ならない。無向グラフ G において、 $d(u, v)$ の最大値を G の直径と呼び、 $d(G)$ で表わす。特に、その直径 $d(G)$ が有限のとき、 G は連結であるといい、そうでないとき、非連結であるという。どの2頂点 $u, v \in V(D)$ に対しても、 u と v の間に少なくとも1方向の有向歩道 (u から v 、または v から u) が存在するとき、 D は片側連結であるという。さらに、双方向の有向歩道が存在するとき、すなわち、任意の2頂点 $u, v \in V(D)$ に対して $d(u, v)$ と $d(v, u)$ がともに有限であるとき、 D は強連結であるという。

$$\text{強連結} \implies \text{片側連結} \implies \text{弱連結}$$

有向グラフ D の頂点集合 $V(D)$ において、 u と v の間が双方向に到達可能であるとき $u \rightleftharpoons v$ と定めると、関係 \rightleftharpoons は $V(D)$ 上の同値関係になる。したがって、関係 \rightleftharpoons で D の頂点を分類すると、 $V(D)$ の分割 V_1, V_2, \dots, V_k が得られる。

$$V(D) = V_1 \cup V_2 \cup \dots \cup V_k, \quad V_i \cap V_j = \emptyset \quad (i \neq j)$$

その各同値類 $V_i (i = 1, \dots, k)$ が誘導する D の部分グラフ $\langle V_i \rangle$ を D の強連結成分と呼ぶ.

有向グラフ D の有向閉路に含まれる頂点は D の強連結成分であることがわかる. 有向グラフ D が有向閉路を含まないとき, D は無閉路的であるという.

命題 2.1 無閉路的な有向グラフには, 出次数が 0 の頂点と入次数が 0 の頂点が存在する.

強連結性の判定問題を考える. まず初めに, 強連結でない, といえるための十分条件を与える.

命題 2.2 与えられた有向グラフ D に, 出次数または入次数が 0 の頂点が存在するならば, D は強連結でない.

有向グラフ D において双方向に到達可能な 2 頂点は 1 つの有向閉歩道上にある. さらに, その有向閉歩道上のどの 2 頂点もその上で双方向に到達可能である. したがって, D の 1 つの強連結成分に属す 2 頂点はその強連結成分の中で双方向に到達可能である. この事実から, 強連結成分を以下のようにも定義できる.

定義 2.3 D を有向グラフとする. D が含む極大な強連結部分グラフを D の強連結成分という.

強連結でない有向グラフにはどの強連結成分にも含まれない辺が存在する. そこで, その辺全体が作る構造を明示するために, 有向グラフ D の各強連結成分 $\langle V_i \rangle$ を 1 点につぶして得られるグラフを D^* で表わし, D の凝縮グラフと呼ぶ. 凝縮グラフを次のように形式的に定義する.

$$V(D^*) = \{V_1, \dots, V_k\}, A(D^*) = \{(V_i, V_j) \mid \exists u \in V_i, \exists v \in V_j, (u, v) \in A(D)\}$$

次に示す命題は, 強連結成分の定義として定義 2.3 を採用することの正当性を与える.

命題 2.4 有向グラフ D の凝縮グラフ D^* は無閉路的である.

命題 2.5 $|V|$ をグラフの頂点数, $|E|$ を辺の本数とする. 頂点数 3 以上の平面的グラフ G に対して, 次の不等式が成り立つ.

$$|E| \leq 3(|V| - 2)$$

2.2 アルゴリズム理論

本稿で用いるアルゴリズム理論の基礎事項について、主に文献[37]より引用する。

深さ優先探索 (depth-first search; **DFS**) は、その名が示すように、可能ならば常にそのグラフの「より深い部分」を探索するという戦略にしたがう。深さ優先探索は、未探索の外向き辺が残る頂点の中で、最後に発見した頂点 v から出る辺を、探索する。 v の辺をすべて探索し終わると、 v を発見したときに通った辺を「バックトラック (逆戻り)」し、 v の直前の頂点を出る (未探索の) 辺の探索に戻る。この処理は始点から到達可能なすべての頂点を発見するまで続く。未発見の頂点が残されていれば、その1つを新たな始点として探索を繰り返す。アルゴリズムはすべての頂点を発見するまでこのプロセスを繰り返す。

深さ優先探索の実行時間を解析する。

定理 2.6 [37] グラフ $G = (V, E)$ が入力されたとき、深さ優先探索の総実行時間は $\Theta(|V| + |E|)$ である。

昔からよく知られている深さ優先探索の応用の1つに、有向グラフを強連結成分に分解する問題がある。2回の深さ優先探索を用いてこの分解を計算する方法を考える。

定理 2.7 [37] グラフ $G = (V, E)$ が入力されたとき、強連結成分は $\Theta(|V| + |E|)$ 時間で計算できる [37]。

このアルゴリズムは S. R. Kosaraju と M. Sharir の業績である。

このアルゴリズムとは別に、再帰アルゴリズムを用いた Tarjan の手法もある [33]。このアルゴリズムもまた強連結成分を線形時間で求めることができる。

2.3 離散・計算幾何学

本稿で用いる離散・計算幾何学の基礎事項について、主に文献[36]より引用する。

複数の単位正方形に対して、隣接する2つが単位辺を完全に共有するように並べて得られる多角形をポリオミノと呼ぶ [7]。面積に応じて、モノミノ、ドミノなどと呼ぶことがある。例として面積が5であるペントミノを図2.1に示す。その形状をアルファベットにたとえて、左上から順に、F, L, N, V, T, X, W, P, U, Z, I, Y と呼ぶこともある。

多角形 P が凸であるとは、 P 中の任意の2点 p, q に対して、線分 \overline{pq} が P に完全に含まれるときのことをいう。例えば、図2.1の多角形のうち凸多角形は唯一つである。多角形 P が単純多角形であるとは、 P が自己交差せず連結で、かつ穴がないときのことをいう。例えば、図2.1の多角形はすべて単純多角形である。直線

l に関して単調な単純多角形とは、 l に垂直などんな直線 l' に対しても、 l' と多角形との交差部分が連結であるものである。いい換えると、交差部分は線分か 1 点か、あるいは空でなければならない。図 2.2 に示すように x 軸に関して単調である多角形は x 単調と呼ばれる [36]。例えば、図 2.1 の多角形はすべて x 単調な多角形である。

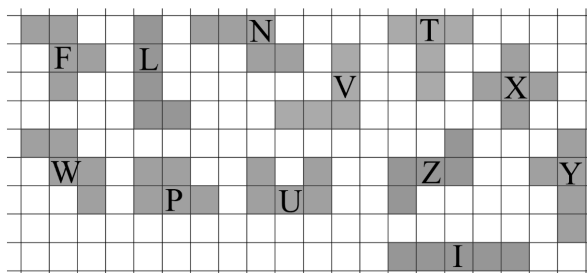


図 2.1: ペントミノ

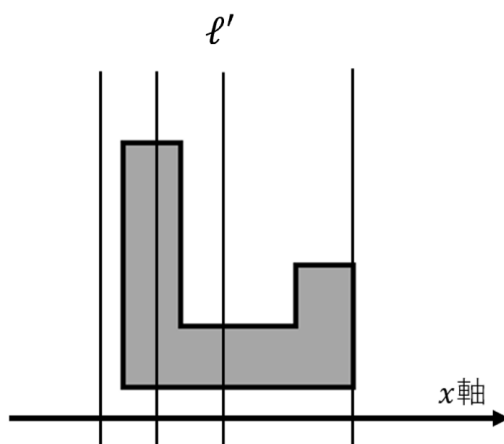


図 2.2: x 単調なポリオミノ

例えば線分の交差問題を考えるとき、実際的な状況においては、ほとんどの線分は他の線分とまったく交わらないか、あるいはほんの少数の線分としか交差しないので、すべての線分対について調べるのは効率的ではない。つまり、実行時間が入力線の数だけでなく交点の個数にも依存するようなアルゴリズムが欲しい。このようなアルゴリズムを出力サイズに敏感なアルゴリズムと呼ぶ。すなわち、アルゴリズムの実行時間が出力のサイズに依存して決まるものである。

定理 2.8 [36] P_1, P_2 をそれぞれ n_1, n_2 個の頂点をもつ多角形とし、 $n := n_1 + n_2$ とする。このとき、 P_1 と P_2 の共通部分は $O(n \log n + k \log n)$ 時間で計算できる。ただし、 k は出力の複雑度である。

2.4 計算量理論

本稿で用いる計算量理論の基礎事項について、主に文献 [39] より引用する。
まず、時間計算量の概念を定義する。

M を停止性チューリング機械、 Σ をその入力アルファベットとする。 M の入力文字列 x に対して、入力 x に対して M が停止する時刻を、 δ によって与えられる次の時刻の状態が q_{acc} または q_{rej} となる時刻と定め、それを $\text{time}_M(x)$ で表わす。

定義 2.9 $T(n)$ を \mathbb{N} から \mathbb{N} への関数、 M をチューリング機械とする。 M が、
すべての入力 x に対して、 $\text{time}_M(x) \leq T(|x|)$ である
という性質をもつとき、 M は $T(n)$ 時間限定であるという。

定義 2.10 $T(n)$ を \mathbb{N} から \mathbb{N} への関数とすると、 $T(n)$ 時間限定のチューリング機械によって受理される言語全体のクラスを $\text{TIME}[T(n)]$ で表わす。すなわち、

$$\text{TIME}[T(n)] = \{L(M) \mid M \text{ は } T(n) \text{ 時間限定である}\}$$

である。また、 $\text{TIME}[T(n)]$ に属する言語は、 $T(n)$ 時間判定可能であるという。

定義 2.11 \mathcal{F} を \mathbb{N} から \mathbb{N} への関数の集合とする。 \mathcal{F} に属するある関数 $T(n)$ に対して、チューリング機械 M が $T(n)$ 時間限定であるとき、 M は \mathcal{F} 時間限定であるという。

定義 2.12 \mathcal{F} を \mathbb{N} から \mathbb{N} への関数の集合とすると、 $\text{TIME}[\mathcal{F}]$ は \mathcal{F} 時間限定であるチューリング機械によって受理される言語全体の集合を表わす。すなわち、

$$\text{TIME}[\mathcal{F}] = \bigcup_{T(n) \in \mathcal{F}} \text{TIME}[T(n)]$$

である。また、 $\text{TIME}[\mathcal{F}]$ に属する言語は、 \mathcal{F} 時間判定可能であるという。

定義 2.13

1. **P** は多項式時間限定の決定性チューリング機械によって受理される言語のクラスである。すなわち、

$$\mathbf{P} = \bigcup_{f: f \text{ は多項式}} \text{TIME}[f(n)]$$

である。**P** に属する言語は、多項式時間判定可能であるという。

2. **NP** は多項式時間限定の非決定性チューリング機械によって受理される言語のクラスである。すなわち、

$$\mathbf{NP} = \bigcup_{f: f \text{ は多項式}} \text{NTIME}[f(n)]$$

である。**NP** に属する言語は、非決定的多項式時間判定可能であるという^a。

^a計算量クラス **NP** という名前も、非決定性チューリング機械で多項式時間判定可能 (Nondeterministic Polynomial time) からきている [40]。

定義 2.14 言語 $A \subseteq \Sigma^*$ が言語 $B \subseteq \Lambda^*$ に多項式時間多対一還元可能であるとは、FP に属する関数 $f: \Sigma^* \rightarrow \Lambda^*$ が存在して、すべての $x \in \Sigma^*$ に対して、

$$x \in A \iff f(x) \in B$$

が成り立つことをいう。つまり、 f は多項式時間計算可能関数で、 A の任意の要素を B の要素に、 \bar{A} の任意の要素を \bar{B} の要素にうつす。この性質をもつ f を、 A から B への多項式時間多対一還元と呼ぶ。 A が B に多項式時間多対一還元可能であることを、式 $A \leq_m^p B$ を用いて表わす。

ここで、本稿で扱う「問題」について改めて定義する [40]。

定義 2.15 本稿で扱う問題とは、与えられた入力に対して出力を計算する問題である。したがって、各問題はその入出力の対応関係を表わす関数によって規定される。なお関数はどんな入力にも値が定義されている全域的関数と仮定する。

ある集合 S に対し、 S の任意の要素 x を入力として与えたときに、 x がある性質 Q をもつかどうかを決定する問題を判定問題という。判定問題を決定問題と呼ぶこともある。言語クラス C の完全問題とは、そのクラスの中で本質的に判定がもつ

とも難しい判定問題である。

還元可能性を用いて完全問題の概念を導入する。

定義 2.16 C をクラス **NP** とする。言語 A が C 困難であるとは、 C のあらゆる言語 L に対して、 $L \leq_m^p A$ が成り立つことである。

定義 2.17 C をクラス **P, NP** のいずれかであるとする。言語 A が C 完全であるとは、 $A \in C$ かつ A が C 困難であることである。

NP 完全問題の大きな特徴のひとつは、ある **NP** 完全問題が多項式時間の解法をもてば、それを使って、あらゆる **NP** 完全問題を多項式時間で解くことができるということである。

命題 2.18 A が **NP** 困難かつ $A \leq_m^p B$ であるなら、 B は **NP** 困難である。

命題 2.19 A が **NP** 完全であり、 $A \leq_m^p B$ であり、 $B \in \mathbf{NP}$ であるなら、 B は **NP** 完全である。

命題2.19を用いて問題 B が **NP** 完全であることを証明するには、 $B \in \mathbf{NP}$ であることと、ある **NP** 完全問題が B に多項式時間多対一還元可能であることを示せば良い。

文献 [35] より、代表的な **NP** 完全問題を3つ紹介し、その問題例を示す。これら3つはすべて判定問題であるので、出力は Yes/No のみで十分であり、具体的な解を返す必要はないが、本稿では根拠となる解も記しておく。

問題 2.20 ナップサック問題

入力：多重集合 $\hat{A} = \{a_1, a_2, \dots, a_n\}$, B .

条件：各 a_i は正整数とする。

質問： $\sum_{i \in S} a_i = B$ となる添字の集合 $S \subseteq \{1, 2, \dots, n\}$ が存在するか？

例 2.21 $n = 10$, $B = 38$ の場合

入力： $\hat{A} = \{6, 7, 8, 9, 10, 18, 25, 26, 28, 28\}$, $B = 38$.

出力：Yes. (例えば, $\hat{A}' = \{10, 28\}$.)

問題 2.22 箱詰め問題

入力：多重集合 $\hat{A} = \{a_1, a_2, \dots, a_n\}$, B , k .

条件：各 a_i は正整数とする.

質問：添字の集合 $U = \{1, 2, \dots, n\}$ を U_1, U_2, \dots, U_k の k 個に分割し, 各 j で $\sum_{i \in U_j} a_i \leq B$ とすることは可能か?

例 2.23 $n = 20, B = 30, k = 10$ の場合

入力： $\hat{A} = \{2, 3, 4, 6, 7, 7, 8, 9, 9, 10, 10, 11, 12, 16, 17, 18, 25, 25, 28, 28\}$,
 $B = 30, k = 10$.

出力：Yes. (例えば, $\hat{A}' = \{\{25\}, \{28\}, \{2, 28\}, \{3, 25\}, \{10, 17\}, \{11, 18\}, \{4, 16, 10\}, \{9, 9, 12\}, \{6, 7, 7, 8\}\}$.)

問題 2.24 3-Partition 問題

入力：多重集合 $\hat{A} = \{a_1, a_2, \dots, a_{3m}\}$, B .

条件：各 a_i は正整数とし, $\frac{1}{4}B < a_i < \frac{1}{2}B$ を満たす.

質問：多重集合 \hat{A} を m 個の 3 つ組 A_1, A_2, \dots, A_m に分割できるか? ただしここで, $A_l = \{a_i, a_j, a_k\}$ としたとき, 各 l に対して $\sum_{s \in A_l} a_s = B$ が成立する.

例 2.25 $m = 5, B = 23$ の場合

入力： $\hat{A} = \{6, 6, 6, 6, 6, 7, 7, 7, 8, 8, 8, 9, 10, 10, 11\}$, $B = 23$.

出力：Yes. (例えば, $\hat{A}' = \{\{6, 6, 11\}, \{6, 7, 10\}, \{6, 7, 10\}, \{6, 8, 9\}, \{7, 8, 8\}\}$.)

2.5 本稿で扱う問題の定義

問題 2.26 アンチスライドパズル

入力：ピース集合 \mathcal{P} とフレーム F .

条件： \mathcal{P} の要素はすべて単純多角形で, F は穴が 1 つある多角形. そして F に \mathcal{P} の要素を詰め込めることが多項式時間で判定できる.

質問： F に \mathcal{P} の要素をすべて詰め込んで「アンチスライド配置」にできるか?

問題 2.27 アンチスライド性判定問題

入力：ピース集合 \mathcal{P} がすべてフレーム F の中に詰め込まれた配置 \mathcal{A} .

条件： \mathcal{P} の要素はすべて単純多角形で, F は穴が 1 つある多角形.

質問：配置 \mathcal{A} はアンチスライドか?

ただし「アンチスライド性」には複数のモデルがあり, アンチスライド配置はモデルにより異なる. アンチスライド性は3章以降で改めて定義する.

問題 2.28 インターロックパズル

入力：ピース集合 \mathcal{P} .

条件： \mathcal{P} の要素はすべて単純多角形である.

質問： \mathcal{P} の要素をすべて使って「インターロック配置」にできるか？

問題 2.29 インターロック性判定問題

入力：ピース集合 \mathcal{P} とその配置 \mathcal{A} .

条件： \mathcal{P} の要素はすべて単純多角形である.

質問：配置 \mathcal{A} はインターロックか？

ただし「インターロック性」については3章以降で定義する.

第3章 アンチスライドパズルの数理的 特徴づけ

3.1 「支える」の定義

多角形 P の外周部分を ∂P と書く。多角形 P と Q が接触しているとは、 $\partial P \cap \partial Q \neq \emptyset$ のときをいう。多角形の集合 \mathcal{P} の正当な配置とは、どの二つの要素 $P, Q \in \mathcal{P}$ をとっても ∂P 上の点と ∂Q 上の点以外には共有しない配置をいう。直観的には、どの二つの多角形も重ならない配置である。以下、配置といったときは基本的に正当な配置以外は考えないものとする。

多角形のスライドを考えるには、重力の向きを考える必要がある。適当な向きベクトル \vec{d} が与えられたとき、それが下を示すものと考えて、 $x-y$ 座標系を考える。このとき、 \vec{d} で定義される座標系に対して、点 p の座標を $(x_d(p), y_d(p))$ と書くことにする。

2点 p, q 間のユークリッド距離を $\text{dist}(p, q)$ で表わすことにする。平面 \mathbb{R}^2 上では、

$$\text{dist}(p, q) := \sqrt{(x_d(p) - x_d(q))^2 + (y_d(p) - y_d(q))^2}$$

である。

a を中心とする半径 r の円の円周上の点の集合を、

$$C_r(a) = \{x \in \mathbb{R}^2 \mid \text{dist}(a, x) = r\}$$

とする。

多角形 P が直交であるとは、 P のすべての辺が x 軸か y 軸に平行なときをいう。

多角形 P の直交配置とは、 P のすべての辺が x 軸か y 軸に平行な配置のことである。

定義 3.1 多角形 P と Q が接触しているとき、 ∂Q 上の点 q が、 ∂P 上の点 p を向き \vec{d} に対して支えているとは、以下の条件を満たすことをいう (図 3.1).

1. $\partial P \cap \partial Q$ が p と q を含む. $p = q$ としたとき以下が成り立つ.
2. ある実数 $\epsilon > 0$ が存在して、 $0 < \epsilon' < \epsilon$ を満たすどんな実数 $\epsilon' > 0$ に対しても以下を満たす:
 - (a) ∂Q と $C_{\epsilon'}(q)$ の 2 つの交点を $(q_{\epsilon'-}, q_{\epsilon'+})$ とし、 ∂P と $C_{\epsilon'}(p)$ の 2 つの交点を $(p_{\epsilon'-}, p_{\epsilon'+})$ とする. ここで、それぞれの対 (a, b) は $x_d(a) < x_d(b)$ をみたすものとする.
 - (b) $x_d(q_{\epsilon'-}) < x_d(p) < x_d(q_{\epsilon'+})$ または $x_d(p_{\epsilon'-}) < x_d(q) < x_d(p_{\epsilon'+})$ が成立する.

直観的には、点 p の近傍か点 q の近傍が、他方を支えるだけの十分な幅を持っていることを意味している.

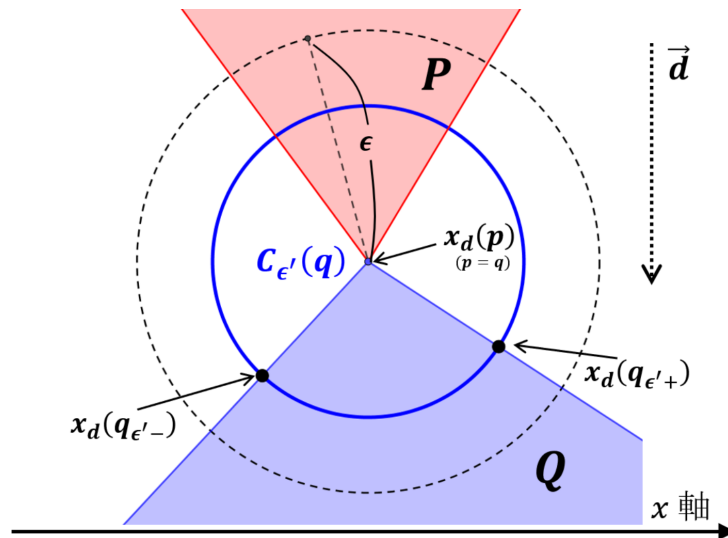


図 3.1: Q が P を下から支えている場合

向きによって支えられていたり支えられていなかったりする例を図 3.2 に示す. 向き \vec{d}_1 に対しては定義 3.1(b) の不等式 $x_d(q_{\epsilon'-}) < x_d(p) < x_d(q_{\epsilon'+})$ が成り立つが、向き \vec{d}_2 に対しては、 $x_d(q_{\epsilon'-}) < x_d(q_{\epsilon'+}) < x_d(p)$ となるので、成り立たない. したがって、 Q は P を、向き \vec{d}_1 に対しては支えているが、向き \vec{d}_2 に対しては支えていない.

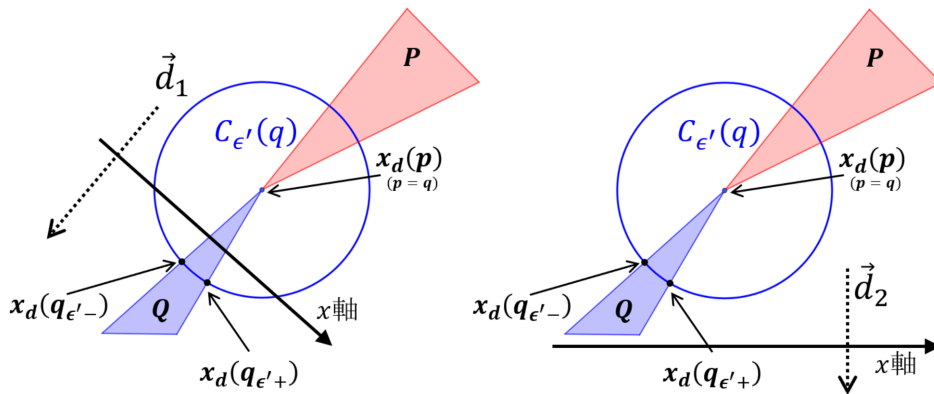


図 3.2: 不等式 $x_d(q_{\epsilon'-}) < x_d(p) < x_d(q_{\epsilon'+})$ を満たす例, 満たさない例
 Q は P を, 向き \vec{d}_1 に対しては支えているが, 向き \vec{d}_2 に対しては支えていない

3.2 「アンチスライド性」の定義

向き \vec{d} に対して, P の下側エンベロープとは, 任意の $\epsilon > 0$ に対して, $x_d(p) = x_d(p')$ かつ $y_d(p) > y_d(p')$ を満たす P の内部の点 p' が距離 ϵ 以内に存在しない ∂P 上の点 p の集合とする (図3.3). 上側エンベロープも同様に定義する.

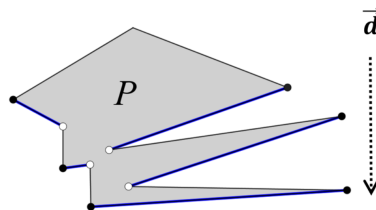


図 3.3: 多角形 P の下側エンベロープ
 黒頂点と青辺は多角形 P の下側エンベロープに含まれるが, 白頂点と黒辺は含まれない.

ここで弱アンチスライド性を定義する.

定義 3.2 多角形 P が向き \vec{d} について弱アンチスライドであるとは, ある多角形 Q が存在して, P の下側エンベロープのある点 p が, Q の上側エンベロープ上の点 q で支えられているときをいう. このとき, 簡単のため単に Q は P を支えているということにする (図3.5).

計算機上でモデル化する際に最も扱いやすいモデルは弱アンチスライド性である. いま, 与えられたアンチスライドパズルの配置 \mathcal{A} が平らな平面上に置かれているとする. このとき, 弱アンチスライド性をみたす配置に対して次の操作を行

なうと、現実的にはアンチスライドではないもの、現実的にもアンチスライドであるものに分けられる。

観察 3.3 平らな平面上に置かれた配置 \mathcal{A} に対して、アンチスライド性の現実的な判定方法は「フレームを手前に起こした時にどのピースも動かないかどうか」を確認することである。

この操作により、図1.6の例は、実際のパズルでは P が右に倒れてしまうが、図3.5(e)の例は、依然としてアンチスライドであるということを確認められる。観察3.3は直観的な主張なので数理的な言葉で言い換える必要がある。そのためには重力を考慮して、重心が支えられているかどうかを議論しなければならない。

武永らが [3] で扱っている問題は、本稿でいう弱アンチスライドモデルである。このモデル上でアンチスライドである図1.6の例は、現実的にはアンチスライドであるとは言いがたい。こうした現実に近い問題を扱おうとすると、多角形の重心を考える必要がある。本稿では多角形は一様な材料でできており、かつ厚みは一定であると考えられる。そうした多角形の重心は、通常の意味での重心として定義し、詳細は省略する。しかし例えばアルファベットの C のような多角形など、重心がその多角形の外部にある場合があることに注意する。そこでここでは次のバランス点を導入する (図3.4)：向き \vec{d} に対する多角形 P のバランス点とは、向き \vec{d} と平行で P の重心を通過する線分 ℓ と P の下側エンベロップの交点 p の中で、 $y_d(p)$ が最も小さい点をいう。多角形 P が厚みが一定の一様な材料でできていると考えると、以下の観察が得られる。

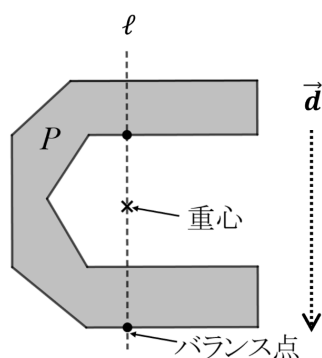


図 3.4: 重心とバランス点

観察 3.4 P が凸多角形のときは、バランス点は一意的に重心に決まる。任意の多角形 P と向き \vec{d} について、バランス点は必ず存在し、一意的に決まる。

次に強アンチスライド性を定義する。

定義 3.5 多角形 P が向き \vec{d} について強アンチスライドであるとは、 P の下側エンベロープ上に2点 p_r と p_l が存在し、以下が成立するときであるとする。

1. P のバランス点 p に対して $x_d(p_l) \leq x_d(p)$ かつ $x_d(p) \leq x_d(p_r)$ である。
2. 多角形 Q, Q' が存在し、 p_r は Q の上側エンベロープ上の点 q に、 p_l は Q' の上側エンベロープ上の点 q' によって支えられている。

ここで p_r と p_l は $p_r = p_l$ でもよいことと、 $Q = Q'$ でもよいことに注意する。直観的には、 P の重心が下から支えられていることを意味している。

強い意味で支えられている例と、支えられていない例を図3.5に示す。ここで特に図3.5(d) の場合が認められないことに注意する。実際には P に上下方向の厚みがあれば、 P は Q' によって支えられるため、下にスライドするとは考えにくい。実際のパズルでもこうした状況はアンチスライドであると考えられている。

以下、固定した座標系におけるポリオミノの直交配置を主な研究対象とする。そこで、定義3.2と定義3.5の中間として次を定義する。

定義 3.6 ポリオミノがアンチスライドであるとは、ポリオミノの辺に平行な4つの方向のそれぞれについて、弱アンチスライドであることをいう。

この定義によって特に図3.5(d)のように下と横から支えられている場合も扱えることに注意する。これは天野らの研究 [1] における3次元版のアンチスライドパズルでは正しく扱われており、また実際の2次元版のアンチスライドパズルでもよく見られる配置である。

	(a)	(b)	(c)	(d)	(e)
弱アンチスライド	×	○	○	○	○
強アンチスライド	×	×	×	×	○
ポリオミノのアンチスライド	×	未定義	×	○	○

図 3.5: \vec{d} を下向きとしたときのピース P のアンチスライド性 (黒点で重心を表わす)

3.3 「インターロック性」の定義

まず、個々のポリオミノがアンチスライドであるとしても、全体としてインターロックしていない場合があることを指摘しておこう。図3.6(a)は、4つのポリオミノはそれぞれが単体ではアンチスライドであるが、2ピースずつ組むと上下にスライドして分けることができる。一方で図3.6(b)の5つのポリオミノはアンチスライドである¹。

こうした場合を正しく把握するため、ポリオミノの配置に対するインターロックを定義する。まず、ある配置において2つのポリオミノが単位辺を1つ以上共有するとき、辺を共有するポリオミノ同士をつないだ1つのポリオミノで置き換えることを、ポリオミノの融合と呼ぼう。例えば、図3.7(a)には $\{u, v, w, x\}$ の4ピースがあるが、これを $\{u, x\}$ と $\{v, w\}$ の2ピースずつに分割し、それぞれを1ピースとみなす(図3.7(b))。すなわち、図3.7(a)の4ピースを2ピースに置き換える操作のことをピースの融合と呼ぶ。

これを使って、本稿におけるインターロック性を定義する。

定義 3.7 ポリオミノの集合 \mathcal{P} の直交配置 \mathcal{A} が与えられたとする。

1. \mathcal{P} の任意の部分集合 \mathcal{P}' によって導出される部分配置 \mathcal{A}' を考える。 \mathcal{A}' に含まれるポリオミノを可能な限り繰り返し融合する。これを \mathcal{P}' の融合と呼ぶことにする。ここで \mathcal{P} の任意の部分集合 \mathcal{P}' を融合したときに、これが1つのポリオミノ P' になったとする。このとき \mathcal{A}' がアンチスライドであるとは、 \mathcal{P}' を P' で置き換えた配置において、このポリオミノ P' がアンチスライドであるときをいう。
2. ポリオミノの集合 \mathcal{P} に対する配置 \mathcal{A} がインターロックであるとは、 \mathcal{P} の非空の任意の真部分集合 $\mathcal{P}' \subsetneq \mathcal{P}$ に対して、 \mathcal{P}' がアンチスライドであることをいう。

実際に、図3.6(b)には配置 \mathcal{A} を分離する部分集合 \mathcal{P}' が存在しないことが確認できる。

¹山本浩氏がデザインした Lock Device[26] は、図3.6(a)のようなJ型5ピースからなるパズルである。つまり図3.6(b)の正方形ピースを使わなくても、様々なサイズのJ型ポリオミノだけを5種類使えばインターロックできる。本稿では、作者に敬意を払ってパズルのピースの具体的なサイズは記載しない。

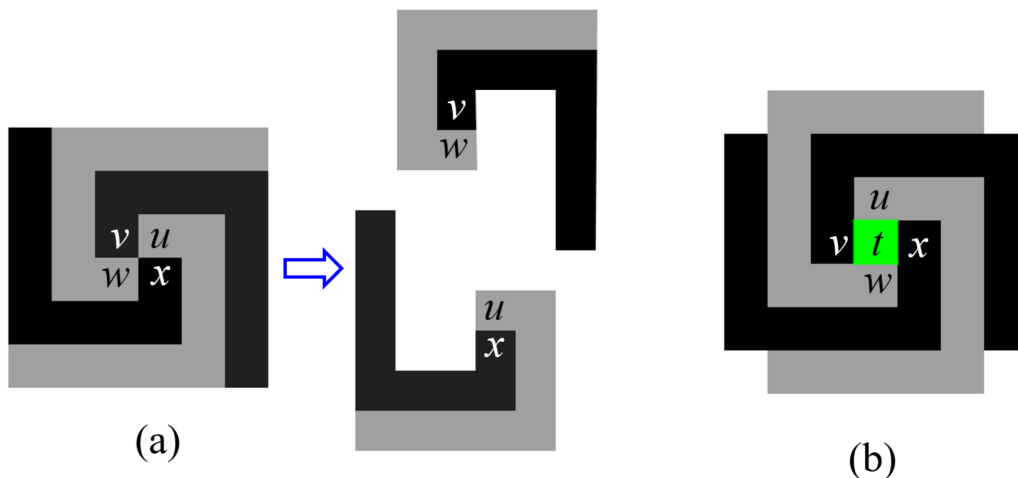


図 3.6: (a): インターロックしていない例, (b): インターロックしている例

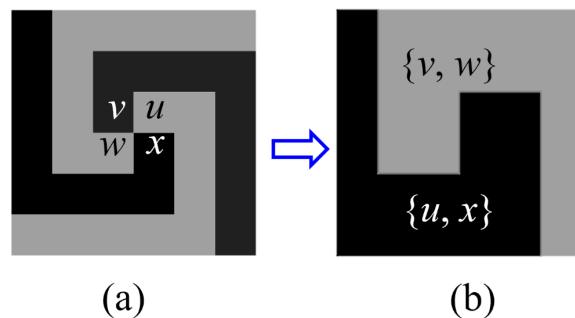


図 3.7: (a): ピース配置 \mathcal{A} , (b): ピースを融合して得られる配置 \mathcal{A}'

「スライド操作」とは2次元平面内における操作である。また、アンチスライド・インターロック状態とはスライド操作だけでは解除できない状態である。したがって、解除の逆操作である構成についても、2次元空間内だけでアンチスライド・インターロック状態をつくることはできない。図3.8に図3.6(b)の構成手順の例を示す。手順1から3までは2次元空間内のスライド操作のみで構成できるが、手順4をするためには2次元空間を出て3次元空間的な操作が必要である。



図 3.8: インターロック状態の構成手順の例

定義3.7にしたがうと、指数関数的に存在する任意の部分集合 \mathcal{A}' に対してアンチスライドであるかどうかを試す必要があるように見えるが、実はそうではない。次章では、与えられたポリオミノの配置がインターロックしているかどうかを多項式時間で判定するアルゴリズムを示す。

第4章 アンチスライド性判定問題の 多項式時間アルゴリズム

4.1 ピース配置の有向グラフ表現と多項式時間判定アル ゴリズム

本章では、ポリオミノの配置が与えられたとき、それがアンチスライドであるかどうかを多項式時間で判定できることを示す。ここでは、フレームは考えず、与えられたポリオミノの集合がインターロックしているかどうかを判定する問題を考える。本章の最後に、アンチスライド性判定問題に対してもこの手法を同様に適用できることを確認する。

ポリオミノの集合 \mathcal{P} の直交配置 \mathcal{A} が与えられたとする。ポリオミノの辺に平行な4つの方向を $\{N, E, S, W\}$ とする。本稿では地図の表記と同様、下を S とし、他の方向も同様に定義する。

定義 4.1 集合 \mathcal{P} の直交配置 \mathcal{A} と方向 $d \in \{N, E, S, W\}$ に対して、有向グラフ表現 $D_d(\mathcal{A}) = (\mathcal{P}, A_d)$ を次のように定義する： $P, Q \in \mathcal{P}$ に対して、有向辺 $(P, Q) \in A_d$ である必要十分条件は、向き \vec{d} に対して P が Q に支えられていることである。

図4.1と図4.2の (b),(c) に、それぞれ (a) に示す配置を表現する有向グラフ $D_S(\mathcal{A})$, $D_E(\mathcal{A})$ を示す。図4.1では u と x を融合すると v, w に対して乗っかっている構造になっていることがグラフに現れているが、図4.2のグラフではそうした頂点集合は存在しない。

図4.1(b) に示す $D_S(\mathcal{A})$ の強連結成分 (定義2.3) は $\{u, x\}$ と $\{v, w\}$ の2つである。図3.6(a) でみたように、図4.1(a) に示すピース配置 \mathcal{A} は方向 S に対して、ちょうどこの強連結成分 $\{u, x\}$ と $\{v, w\}$ の2つに分解される。頂点分割 $\{u, x\}$ と $\{v, w\}$ をそれぞれ1頂点に置き換えることによって得られる凝縮グラフ $D_S(\mathcal{A})^*$ は、確かに命題2.4でみたように強連結ではない。一方で図4.2の (b),(c) の $D_S(\mathcal{A})$, $D_E(\mathcal{A})$ はそれぞれ強連結であり、図4.2(a) の配置 \mathcal{A} は図4.1(a) の配置 \mathcal{A} のように分解されない。このことが一般に成り立つことを示す。

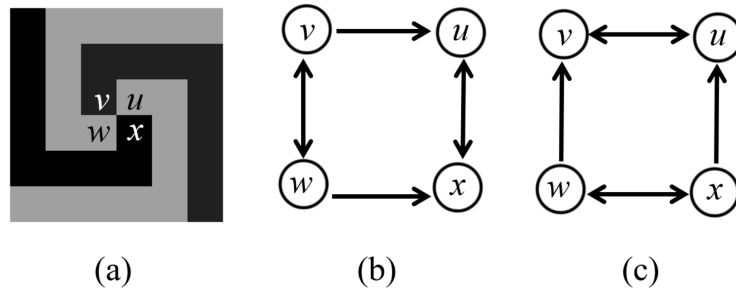


図 4.1: インターロックしていない例. (a): 配置 \mathcal{A} , (b): $D_S(\mathcal{A})$, (c): $D_E(\mathcal{A})$

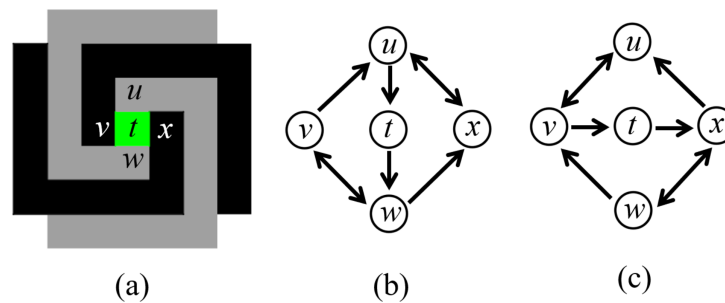


図 4.2: インターロックしている例. (a): 配置 \mathcal{A} , (b): $D_S(\mathcal{A})$, (c): $D_E(\mathcal{A})$

補題 4.2 ポリオミノの集合 \mathcal{P} の直交配置 \mathcal{A} が与えられたとする. このとき配置 \mathcal{A} がインターロックする必要十分条件は, 4 つのグラフ $D_N(\mathcal{A}), D_E(\mathcal{A}), D_S(\mathcal{A}), D_W(\mathcal{A})$ がどれも強連結であることである.

証明 一般性を失うことなく, 方向 $d = S$ のときのグラフ $D_S(\mathcal{A})$ を考える. ここで $D_S(\mathcal{A})$ は連結であると仮定してよい.

まず \mathcal{A} がインターロックするとき, $D_S(\mathcal{A})$ が強連結であることを示す. $D_S(\mathcal{A})$ が強連結でないと仮定して矛盾を示す. $D_S(\mathcal{A})$ が強連結でないとすると, \mathcal{P} を \mathcal{P}_1 と \mathcal{P}_2 に分割して, 任意の $P_1 \in \mathcal{P}_1$ と $P_2 \in \mathcal{P}_2$ に対して, P_1 から P_2 にはたどり着けるが, P_2 から P_1 にはたどり着けないようにできる. こうした \mathcal{P}_1 に対して, \mathcal{P}_1 の要素を融合する. すると, このポリオミノ (2 つ以上である場合はどれでも良い) は \mathcal{P}_2 に対して方向 N にスライドすることができる. これはインターロックしているという仮定に反する. したがって $D_S(\mathcal{A})$ は強連結である.

つぎに $D_S(\mathcal{A})$ が強連結であったときに, \mathcal{A} がインターロックしていることを示す. 支えることの推移性を考えれば, P_1 から P_2 への有向路があるときは, P_1 はいくつかのポリオミノを挟んで P_2 を支えている. 逆に P_2 から P_1 への有向路があるときには P_2 はいくつかのポリオミノを挟んで P_1 を支えている. したがって, 互いに有向路があるときは, 互いに支え合っており, P_1 を P_2 から S 方向にも N 方

向にも遠ざけることはできない。 $D_S(\mathcal{A})$ が強連結であったときは、どの2つのポリオミノも同様の関係にあるため、インターロックしている。

よって、配置 \mathcal{A} がインターロックする必要十分条件は、4つのグラフ $D_N(\mathcal{A}), D_E(\mathcal{A}), D_S(\mathcal{A}), D_W(\mathcal{A})$ が強連結であることである。 \square

なお、一連の定義は、向きに対して対称となっている。このため、 $D_N(\mathcal{A})$ の有向辺をすべて反転させれば $D_S(\mathcal{A})$ が得られる。 $D_E(\mathcal{A})$ と $D_W(\mathcal{A})$ も同様である。よって以下の系が得られる。

系 4.3 ポリオミノの集合 \mathcal{P} の直交配置 \mathcal{A} が与えられたとする。このとき配置 \mathcal{A} がインターロックするかどうかは、2つのグラフ $D_E(\mathcal{A}), D_S(\mathcal{A})$ がどちらも強連結であるかどうかを調べれば十分である。

系4.3において2つのグラフ $D_E(\mathcal{A}), D_S(\mathcal{A})$ のみを調べれば十分である、というのは定義3.6でみたとおり、本稿で扱う「ポリオミノの直交配置に関してのみ」という限定的な場合であることに注意が必要である。

定理 4.4 ポリオミノの集合 \mathcal{P} の直交配置 \mathcal{A} が与えられたとする。 \mathcal{P} の頂点数の総和を n としたとき、配置 \mathcal{A} がインターロックしているかどうかは、 $O(n^3 \log n)$ 時間で判定できる。

証明 与えられたポリオミノ集合 \mathcal{P} の配置 \mathcal{A} に対して、2つの方向 $\{E, S\}$ のそれぞれについてグラフを構成する。ここでは例として $D_S(\mathcal{A})$ を考える。

入力 n は \mathcal{P} の頂点数の総和である。ここでポリオミノの頂点数とは、例えば、ポリオミノ P がI型ペントミノであるとき頂点数は4で、Y型ペントミノであるとき頂点数は6である (図2.1)。

与えられたポリオミノ集合 $\mathcal{P} = \{P_1, P_2, \dots, P_k\}$ のそれぞれが頂点数 $\{n_1, n_2, \dots, n_k\}$ であったとする。つまり $\sum_{i=1}^k n_i = n$ とする。

P_i を P_j が方向 S に対して支えているかどうかは定理2.8より $O((n_i + n_j) \log(n_i + n_j))$ 時間で判定できる。

具体的には、 P_i の各辺 e と P_j の各辺 e' のペアに対して、 e と e' が正の長さの共通部分を持つかどうかということと、どちらが下にあるかを判定すればよい。

上記を \mathcal{P} すべてのペアに対して調べて $O(n^2)$ 時間かかり、定理2.8より、2つの多角形の共通部分を求めるために $O(n \log n)$ 時間かかるのでグラフ $D_S(\mathcal{A})$ を構成するための全体の計算量は $O(n^3 \log n)$ 時間となる。

P_i が P_j に対して弱アンチスライドであった場合、定義4.1にしたがって、グラフ $D_S(\mathcal{A})$ における頂点 v_{P_i} から頂点 v_{P_j} へ有向辺 (v_{P_i}, v_{P_j}) を張る。このようにして有向グラフ $D_S(\mathcal{A})$ を構成する。

ポリオミノ集合 \mathcal{P} の頂点数の総和が n なので、 $D_S(\mathcal{A})$ の頂点集合 \mathcal{P} は $\sum_{i=1}^k n_i = n$ より、 $|\mathcal{P}| = O(n)$ を満たす。配置 \mathcal{A} は平面的であるので、出来上がるグラフ $D_S(\mathcal{A})$

も平面的となり，命題2.5より辺の総数 $|E|$ は， $|E| \leq 3(|\mathcal{P}|-2)$ で抑えられる．したがって， $|E| = O(|\mathcal{P}|)$ かつ $|\mathcal{P}| = O(n)$ なので $|E| = O(n)$ である．よって， $D_S(\mathcal{A})$ の強連結性は線形時間 ($O(|\mathcal{P}| + |E|)$ 時間) で判定できる [33].

したがって，配置 \mathcal{A} がインターロックしているかどうかは， $O(n^3 \log n)$ 時間で判定できる. □

観察 4.5 図4.1, 図4.2に示すインターロックパズルの有向グラフ表現は，図4.3に示すアンチスライドパズルのインスタンスに対しても同様にして適応できる．図4.3(a)にはピース集合 \mathcal{P} とフレーム F があるので，アンチスライドパズルの定義 (問題2.26) よりアンチスライドパズルのインスタンスである．そのフレーム F を1つの頂点として有向グラフを作れば，インターロック性判定問題の議論に還元できる．

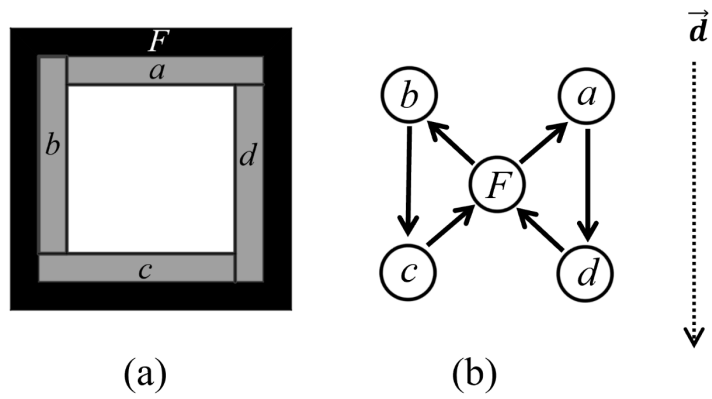


図 4.3: 図1.3(b) の有向グラフ表現. (a): 配置 \mathcal{A} , (b): $D_S(\mathcal{A})$

第5章 アンチスライドパズルの困難性

定理 5.1 \mathcal{P} と F をアンチスライドパズルのインスタンスとする。ここで、 \mathcal{P} を大きさ $1 \times k$ の長方形 $3m$ 個からなる集合（ただし k は正整数）とし、 F を1個の穴を持つポリオミノとする。このとき \mathcal{P}, F に対してアンチスライド配置を求める問題は、直交配置に限定しても **NP 完全** である。

証明 まず、 \mathcal{P} と F はどれもポリオミノであるので、定理4.4と観察4.5より、与えられた配置がアンチスライドであるかどうかは多項式時間で判定できる。したがってアンチスライドパズルは計算量クラス **NP** に入る。よって **NP 困難性** を示せばよい。3-Partition 問題（問題2.24）からの多項式時間還元を示す。

3-Partition 問題の入力の $\hat{A} = \{a_1, a_2, \dots, a_{3m}\}$ のそれぞれの a_i に対して、図5.1(b) に示すサイズ $1 \times k'a_i$ の長方形ピース P_i を用意する。（ただし、 k' は $4(3m+1)/B < k'$ を満たす正整数とする。）またフレーム F は、図5.1(a) に示すとおり内部に $m \times k'B$ の穴と、 $2m \times k'B/2$ の穴がつながって1つの穴になったポリオミノとする。

まず、 F に \mathcal{P} の要素をすべて詰めるパッキング問題は簡単に解けることを確認する。フレーム F の穴の面積の合計は少なくとも $3m \times k'B/2$ であり、 \mathcal{P} の要素である $3m$ 個のピース P_i はそれぞれ高々 $1 \times k'B/2$ である。したがって、それぞれの i に対してピース P_i をフレーム F の高さ $i+1$ の場所にパッキングすることができる。

次に元の3-Partition問題のインスタンス \hat{A}, B が解を持つことの必要十分条件が、この問題のインスタンス \mathcal{P}, F に対して、 \mathcal{P} が F の中でアンチスライドにできることであることを示す。

\hat{A}, B が解を持つとき、 m 個の3つ組 $A_l = \{a_i, a_j, a_k\}$ で、どの3つ組も $a_i + a_j + a_k = B$ を満たす \hat{A} の分割が存在する。この3つ組に対応するピース $\{P_i, P_j, P_k\} \in \mathcal{P}$ を図5.1(a) の F のサイズ $m \times k'B$ の場所に詰めると、穴の中で3つのピースがアンチスライドになる。逆に、 \mathcal{P} が F の中でアンチスライドになったとき、 a_i は正整数であるという条件より、 $1 \leq \frac{1}{4}B < a_i < \frac{1}{2}B$ である。よって、任意の2つ組 P_i, P_j に対して $P_i + P_j > k'B/2$ なので、 F の $2m \times k'B/2$ の場所にピースを2つ詰めることはできない。よって、 $2m \times k'B/2$ の場所に詰めることができるピースの数はちょうど1つであるが、 $a_i < \frac{1}{2}B$ より、それに対応するピース P_i は $2m \times k'B/2$ の場所

でアンチスライドにはなりえない。よって「 \mathcal{P} が F の中でアンチスライドである」という仮定を満たすためには、 \mathcal{P} のすべてのピース P_i を $m \times k'B$ の穴の中に詰めなければならない。

このとき、 $a_i + a_j < 2 \times \frac{1}{2}B = B$, $a_i + a_j + a_k + a_l > 4 \times \frac{1}{4}B = B$ より、 $m \times k'B$ の場所に詰めるピースの数が2つのときアンチスライドにはなりえず、4つのときは入らない。したがって、アンチスライド配置を作るためには、 $m \times k'B$ の場所にはそれぞれちょうど3つのピースを詰めなければならない。これらのピースは、 $\frac{1}{4}k'B < k'a_i < \frac{1}{2}k'B$ を満たすので、ピースの3つ組 $\{P_i, P_j, P_k\} \in \mathcal{P}$ から、 $a_i + a_j + a_k = B$ を満たす正整数の3つ組 $A_l = \{a_i, a_j, a_k\}$ が得られる。

したがって3-Partition問題のインスタンス \hat{A}, B が解を持つことの必要十分条件は、アンチスライドパズルのインスタンス \mathcal{P}, F に対して、 \mathcal{P} が F の中でアンチスライドにできることである。よってアンチスライドパズルはNP完全問題である。□

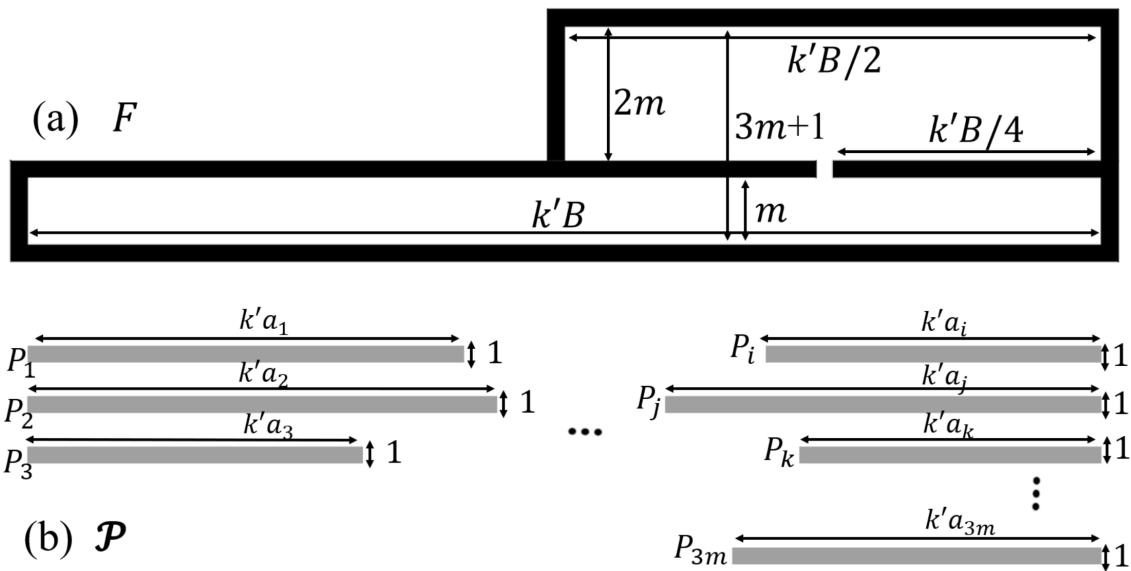


図 5.1: (a) フレーム F , (b) 穴に詰めるピース P_i のサイズ (ただし、 k' は $4(3m+1)/B < k'$ を満たす正整数とする.)

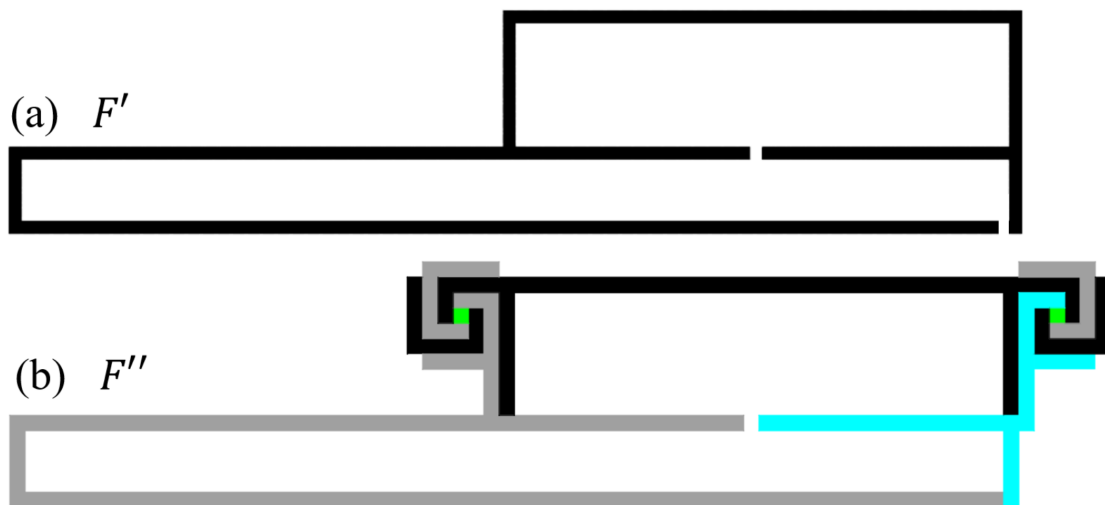


図 5.2: 図5.1(a) の F を変形する.

(a) 穴を 0 個にすることによってインターロックパズルのインスタンスにしたもの,
 (b) x 単調なポリオミノ 9 つでインターロックにしたもの

定理5.1の \mathcal{P} の要素は, すべて凸なポリオミノである. しかし, フレーム F は単純多角形ではないので, インターロックパズルの定義 (問題2.28) より, これをインターロックパズルのインスタンスとみなすことはできない. したがって, 定理5.1のフレーム F を単純多角形に変形することによって, インターロックパズルの困難性を証明する.

定理 5.2 \mathcal{P} をインターロックパズルのインスタンスとする. このときインターロック配置を見つける問題は, \mathcal{P} が以下のいずれかの条件を満たしているとき, 直交配置に限定しても **NP** 完全である.

1. \mathcal{P} のポリオミノ 1 ピースを除いて他はすべて長方形ポリオミノであるとき.
2. \mathcal{P} のすべての要素が x 単調なポリオミノであるとき.

証明 (1) フレーム F' の外周の一部を単位長だけ欠けさせることにより, 穴を 0 個にできる. これを F' と呼ぶことにする. この操作によって F を単純多角形に変形できるので, これとピース集合 \mathcal{P} とを合わせて, インターロックパズルのインスタンスとみなすことができ, 定理5.1の議論がそのまま成立する.

(2) 定理5.1の F を図5.2(b) に示す x 単調な 9 ピースに分割すればよい. これを F'' と呼ぶことにする. ここでフレーム F'' の左上と右上の部分を \mathcal{P} の要素に比べて十分小さな単位にすれば, インターロックにする方法は一意的に定まり, 定理5.1の議論がそのまま成立する. □

第6章 インターロック可能な多角形の境界条件の解明

これまでの章では多角形をポリオミノに、配置を直交配置に限定した議論であったが、本章では一般多角形の一般配置について言及する。

本章ではインターロック配置を構成不可能な多角形の境界条件を解明する。具体的には、すべてのピースが凸であるとき、いかなる配置に対してもインターロックしないことを示す。これは一見自明に思えるが、多角形の配置に関して細かい条件分岐がたくさん発生し、証明を短くまとめることは難しい。そこで、本稿では、Edelsbrunnerらの「growing テクニック [34]」を応用してエレガントな証明を与える。

まずこれまでに以下のことを指摘した。

観察 6.1 図3.6(b)に示すとおり、すべての多角形が「 x 単調なポリオミノ」であるときには直交配置に対してインターロックする配置が存在することがある。

この図3.6(b)は山本浩氏のパズル作品「Lock Device」[26]が「5ピースがすべて x 単調なJ型ポリオミノであり、インターロックできるという事実」にヒントを得たものである。

また、以下の命題が成り立つことは、計算幾何学における限界長方形[36]の議論を使って容易に確かめられる。

命題 6.2 [18, Theorem 4(1)] 多角形を凸なポリオミノ（つまり、長方形ポリオミノ）に限定すると、いかなる直交配置に対してもインターロックできない。

本稿では「ポリオミノの直交配置」を主な研究対象としてきたので、観察6.1と命題6.2によってインターロック可能な多角形の境界条件は解明できた。しかし、本章ではより強い結果として、命題6.2を「凸多角形の一般配置」に拡張する。

定理 6.3 任意の自然数 k に対して k 個の凸多角形集合 $\mathcal{P} = \{P_1, \dots, P_k\}$ を考える. このとき \mathcal{P} はいかなる配置に対してもインターロックしない.

証明 $k = 1, 2$ の場合はインターロックしないことは容易に確かめられる. $k > 2$ を \mathcal{P} のある配置 \mathcal{A} がインターロックする最小の整数と仮定する.

この最小配置 \mathcal{A} を [34] と同様の方法で, 多角形を膨らませて \mathcal{P} をすべて含むように三角形 T で周りを囲む. この膨らませる操作によって多角形のいくつかの幾何学的情報 (特に凸性) を崩さないようにする. 膨らませ方のルールは [34, Fig. 5] にしたがう. 図6.1に例を示す. まず初めに与えられた配置 \mathcal{A} をすべて含む三角形 T を描く (図6.1(A)). 次に多角形を膨らませた配置 \mathcal{A}' を構成する (図6.1(B)). 次に \mathcal{A}' について, \mathcal{P} の2つの多角形が正の長さだけ辺を共有するときに限り, その多角形同士を辺でつなぐ, というルールにしたがってコンタクトグラフ $G_C(\mathcal{P}, E)$ を定義する (図6.1(C)). (図6.1(B) に示す b のように「角」にある多角形はスライドすることができないということ, c, d のように「角」にある多角形はユニークには決まらないということに注意する.)

コンタクトグラフ $G_C(\mathcal{P}, E)$ は平面的であり, 配置 \mathcal{A} から自然な方法によって得られることが知られている [34, Lemma 1].

図6.1(B) に示す多角形 a の配置は $G_C(\mathcal{P}, E)$ において切断点となる. もし $G_C(\mathcal{P}, E)$ が切断点を持つとき, その頂点に対応する多角形は \mathcal{A} を2つに分離する. よって, \mathcal{A} はインターロックできないことになり k の最小性に反する. したがって, $G_C(\mathcal{P}, E)$ は切断点を持たない. このとき, T に沿って時計回りに \mathcal{P} の多角形の列を得ることができる. $C = (P_1, P_2, \dots, P_{n'-1}, P_{n'} = P_1)$ を T を巡回することによって得られるサイクルとする (図6.2).

方向 \vec{l}_i を C 上の多角形 P_i と P_{i+1} が (辺を共有する場合) 共有する辺に沿って内側から外側への向きと定義する. $G_C(\mathcal{P}, E)$ は切断点を持たないので方向 \vec{l}_i はユニークに決まる. このとき, \vec{l}_i から \vec{l}_{i+1} への回転角が非負のときに限り P_{i+1} は T の外側へスライドできることを容易に確かめられる.

$\vec{l}_1, \vec{l}_2, \dots, \vec{l}_n$ の回転角の合計は 360° なので, ある i が存在して \vec{l}_i, \vec{l}_{i+1} の間の回転角は正であり, このとき P_i は \vec{l}_i と \vec{l}_{i+1} の間の任意の方向にスライドできて, T の外側へスライドできる. これは \mathcal{A} がインターロックする最小配置であるという仮定に反する. したがって, 凸多角形集合はインターロックできない. □

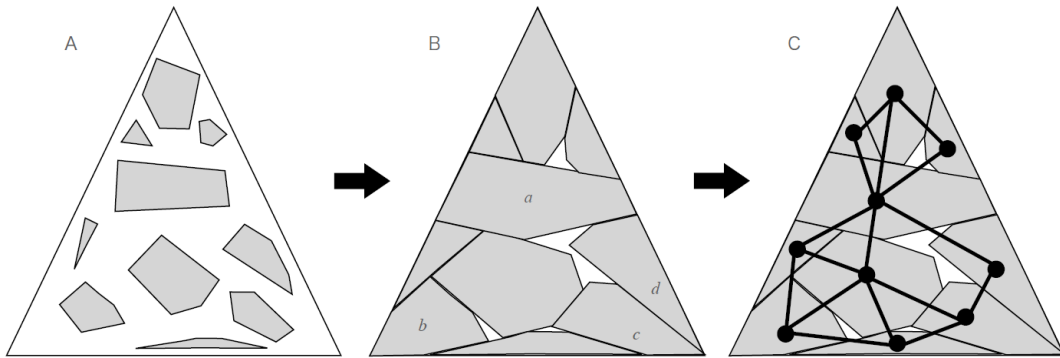


図 6.1: 配置 \mathcal{A} に対するコンタクトグラフの構成手順

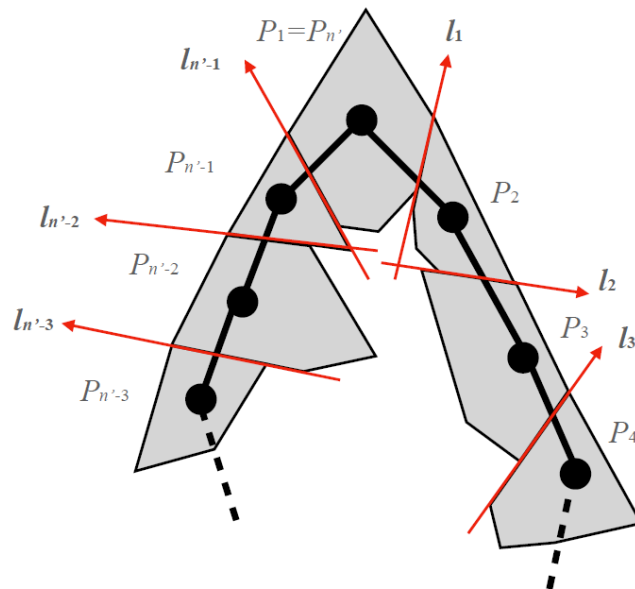


図 6.2: T に沿ったサイクル

第7章 今後の課題

アンチスライドパズルの研究は始まったばかりなので、多くの課題が残されている。思いつく限りの課題を列挙する。本稿のほとんどの箇所では「ポリオミノの直交配置」というとても限定的な場合を扱った。よって次にやるべきの課題は自然に考えると、「ポリオミノの一般配置」や「一般多角形の一般配置」という条件に対してアンチスライド性に関する問題を解くことである。しかし、次にあげるとおり、「ポリオミノの直交配置」に限っても解決すべき問題がまだ残されている。

7.1 ポリオミノの直交配置

補題4.2では、ポリオミノの直交配置 \mathcal{A} に対して、 $\{N, E, S, W\}$ の4方向に対して弱アンチスライド性をみたすならば、 \mathcal{A} がインターロックであるということを証明した。この結果を以下のように拡張できるのかどうかは、証明も反例も与えられなかったので今後の課題とする。

予想 7.1 ポリオミノの直交配置 \mathcal{A} が与えられたとき、 \mathcal{A} が4方向に対してアンチスライドであるならば、全方向に対してもアンチスライドである。

観察6.1では、ポリオミノの直交配置に対して、すべての多角形が「 x 単調なポリオミノ」であるとき、インターロックできることを実例をあげて指摘した。これに対して命題6.2と定理6.3では、すべての多角形が「凸」であるとき、インターロックできないことを証明した。この結果により、インターロックできるための多角形の境界条件を解明できたが、多角形がインターロックできないための境界条件をさらに探求することも今後の課題とする。具体的には以下の予想に対して証明または反例を与えることを今後の課題とする。

予想 7.2 ポリオミノの直交配置に対して、すべての多角形が「 x 単調かつ y 単調なポリオミノ」であるとき、インターロックできない。

例えば、図2.1において「ポリオミノの直交配置に対して、 x 単調かつ y 単調なポリオミノ」に該当しないのはU型ペントミノだけである。図3.6(b)に、インターロックする例として示したJ型ポリオミノも該当しない。

ポリオミノの直交配置に関して、以下に示す問題を提案する。

問題 7.3 2次元版ジェンガ

入力：ピース集合 \mathcal{P} とそのアンチスライド配置 \mathcal{A}

質問： \mathcal{A} の中で $\mathcal{P}' \subseteq \mathcal{P}$ を取り除いた配置 $\mathcal{A} \setminus \mathcal{P}'$ が依然としてアンチスライド性を保つピースの部分集合 \mathcal{P}' は存在するか？

問題 7.4 問題7.3の最大値

入力：ピース集合 \mathcal{P} とそのアンチスライド配置 \mathcal{A}

質問： \mathcal{A} の中で $\mathcal{P}' \subseteq \mathcal{P}$ を取り除いた配置 $\mathcal{A} \setminus \mathcal{P}'$ が依然としてアンチスライド性を保つピースの部分集合 \mathcal{P}' は存在するか？ さらにその最大値はいくつか？

問題7.3では、多角形のアンチスライド配置が与えられたとき、「アンチスライド性を保つように」特定のピースを取り除く。この盤面の遷移中の操作「アンチスライド性を保つように」というのは、いわば遷移中に多角形の組を崩してはいけないという3次元ゲーム「ジェンガ」を2次元版にしたものとみなせる。しかし、ジェンガは3次元空間内で重力の働くただ1方向にのみアンチスライド性を確かめるものであるのに対して、問題7.3や本稿で扱ってきた問題は、2次元空間内で4方向のアンチスライド性を確かめる問題である。

この問題は、入力として既にパッキングされたアンチスライド配置 \mathcal{A} が与えられて「狙ったピースを取り除くことができるかどうか」といったものなので、「狙った隙間をつくる」という意味では、内外相似パズルの困難性の証明 [15] が近い。しかしこれでは順番が逆転して本問には適さないので注意が必要である。「アンチスライドパッキングである」という条件が効いているのでこの問題の困難性の証明には一工夫が必要である。

問題7.4が通常の、ピース数最小のアンチスライド配置探索問題 ([1],[2]) と異なる点は、入力がピース集合 \mathcal{P} だけではなく、その配置 \mathcal{A} まで与えられているという点である。さらに、最初に与えられたアンチスライド配置 \mathcal{A} に依存して取り除けるピースが決まる、ということにも注意が必要である。

例えば図7.1に示すアンチスライドパッキング配置に対して「アンチスライド性を保ったまま取り除くことができるピースはどれか？」という出題 (問題7.3) がなされたとき、答えの一例としては青いN型ペントミノである。これに加えて、例えば赤いP型、Y型ペントミノも取り除くことができる。問題7.4は、このようなピースの部分集合 $\mathcal{P}' \subseteq \mathcal{P}$ の最大値を求める問題である。この問題は本研究の成果の1つである、アンチスライド性に対する有向グラフの強連結性の特徴づけ (補題4.2) を応用すると見通しがよくなる。

アンチスライドパズルで問われるアンチスライド配置は、図7.1に示す密な配置ではなく、図1.3(b)に示す疎な配置であることが多い。このことを踏まえて、上に

示した問題とは少し異なる，次のような問題も考えられる．与えられたアンチスライド配置 A から，あるピース P_i を取り除いた配置 A' について， A' も依然としてアンチスライド配置であったとする．さらに，ピース P_i を配置 A' の空いている場所に詰めた配置 A'' もアンチスライドスライド配置であったとする．この操作を続けて得られるアンチスライド配置に関して何らかの研究を行うことも今後の課題とする．この，ある規則に従って盤面が遷移する問題に対して，「組合せ遷移問題」という名前をつけて研究を行っている研究グループも存在する [32].

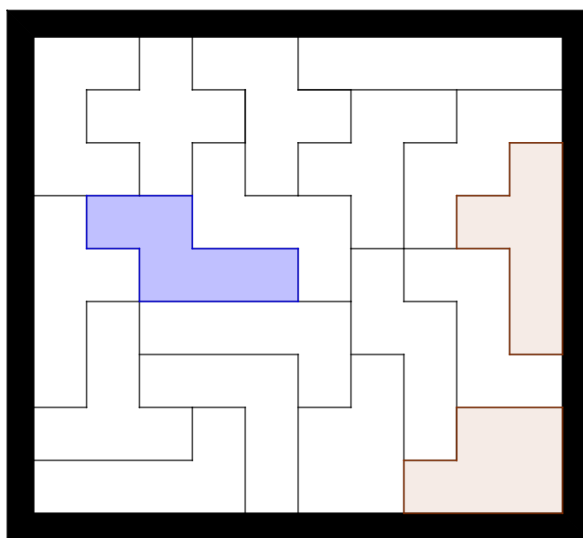


図 7.1: 2次元版ジェンガ

与えられたパッキング配置から，「アンチスライド性を保つように」特定のピースを取り除く問題

7.2 ポリオミノの一般配置

本稿では大部分が直交配置を扱ったので，定義3.1の条件1から2.(a)–(b)まで満たしていれば十分であった．しかし，ポリオミノの一般配置を考えると，図7.2(b)のピース配置も扱わなくてはならない．そのためには定義3.1に以下の条件を加える必要がある．

観察 7.5 $y_d(q_{e^-}) \leq y_d(p)$ かつ $y_d(q_{e^+}) \leq y_d(p)$ が成立する．等号成立は $\partial P \cap \partial Q$ が直線で交わり，かつその直線が x 軸に平行なとき，すなわち直交の場合．

図7.2(b)は，図7.2(a)の配置を，座標軸を固定したまま時計回りに少し回転させた配置である．図7.2(b)は定義3.1の条件1から2.(a)–(b)までは満たしているが，静

止摩擦係数が0の世界では右下に滑り落ちてしまうのでアンチスライドではない。よって、静止摩擦係数が0の世界において、直交配置でない場合を正しく判定するためには、観察7.5にあるように y 座標値に関する不等式評価も必要になる。なお、図7.2(b)は静止摩擦係数が1の世界においてはアンチスライドであるので、定義3.1の条件1から2.(a)–(b)まで満たしていれば十分である。

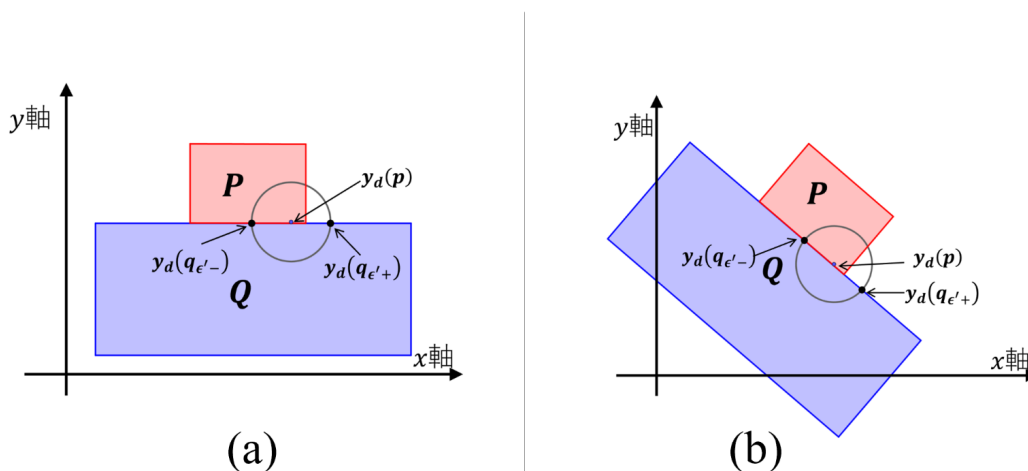


図 7.2: 座標軸に依存するアンチスライド性

しかし観察7.5の判定条件を加えても解決できない問題がある。その1つが図7.3に示す「Rotation モデル」である。図7.3(a)は回転でき、図7.3(b)は回転できない。すなわち、図7.3(a)はアンチスライドではなくて、図7.3(b)はアンチスライドである。このように「回転」に対して正しくアンチスライド性を判定できるモデルを考えたい。ポリオミノの直交配置においては離散的にアンチスライド性を考えればよかったが、この問題は連続的な動きに対してアンチスライド性判定をしなければならないことに注意が必要である。

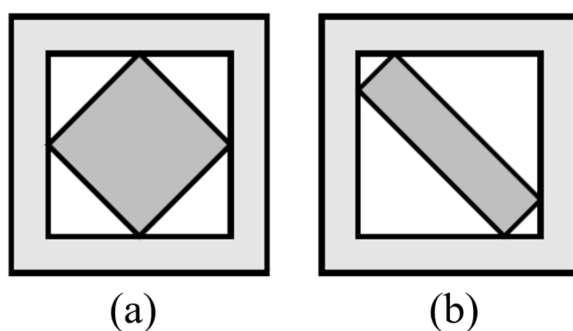


図 7.3: Rotation モデル

アンチスライド状態とは、フレームを傾けた時に「どのピースも動かない」状態のことであった(観察3.3)。すなわち、本稿で提案したアンチスライド性判定の

モデルにおいて、アンチスライド状態の否定は「あるピースが動く」という条件である。

ところで、パズル業界で **Coordinate Motion** と呼ばれる、組木の解体・構成テクニックがある。これは「すべてのピースを同時に動かすことによって、ピース集合を分解する・組むことができる」というものである。すなわち、あるピースが単体で動くことは起こらない。本稿ではすべてのピースが同時に動く場合、すなわちピースの全体集合になって初めてアンチスライドでなくなる場合は、そのピース配置はアンチスライドである、と判定していたので、Coordinate Motion もアンチスライドとなってしまった。ところが現実的には、Coordinate Motion テクニックを使って解体できるピース配置は（本来的には）アンチスライドとは呼べないので、本稿で提案したアンチスライドモデルとは別のモデルを提案する必要がある。

Coordinate Motion には、内側に萎んでいくタイプと外側に広がっていく2パターンがある。このうち、図7.4(a)に示す「内側に萎んでいくタイプ」は、本稿のモデルのままでは誤ってアンチスライドであると判定され、図7.5(b)に示す「外側に広がっていく」タイプは誤ってインターロックであると判定されてしまう。

図7.4(a)の4ピースすべてが同時に時計回りに回転することによって図7.4(b)の配置になる。すなわちこのモデルを考える前に **Rotation** モデルを定義する必要がある。

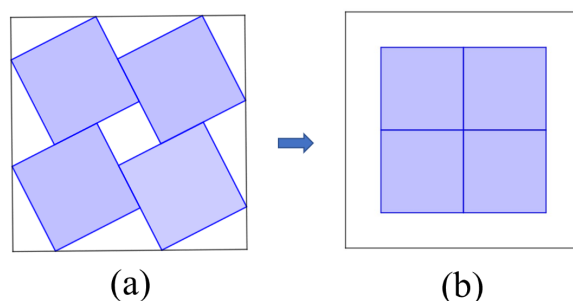


図 7.4: Coordinate Motion モデル（アンチスライド）

7.3 一般多角形の一般配置

1.1.1節で紹介したように、パズルの分類には組木というジャンルがある。いくつかの有名なパズルの分類を示した Web サイトでは組木の分類種別を「Interlocked」としている。おそらくこれは（組木の多くは3次元作品であることを考慮すると）組木を3次元空間的な意味でインターロックである」としているようであるが、本稿の立場からすると上で論じたように「3次元空間内で構成できるのであれば、そのオブジェクトは3次元的なインターロックとは呼べない。なお、組木作品の中で Coordinate Motion と呼ばれるテクニックを使って解体する・構成する作品は本

稿の立場からしてもインターロックであると判定されるので、ここには研究の余地がある。

図7.5(a)は定義3.6でみたポリオミノのアンチスライド性を定義した議論と同じく、辺と平行な方向にスライドするので、例えば赤いピースが単独で動くため、本稿のモデル上でも、方向を正しく見つければ「インターロックではない」と判定できる。一方で図7.5(b)に示す配置は、図7.5(a)を少し変形したもので、3ピースすべてを同時に動かすこと（Coordinate Motion テクニック）によって初めてピース配置を分解できる。本稿においてインターロックの定義は「すべてのピースが相対的に動かない」なので、インターロックの否定は「ある動くピースが存在する」となる。図7.5(b)に示す配置は「すべてのピースを相対的に同時に動かすと分解できる」ので、本稿におけるインターロックの否定になっていないにもかかわらず、実際にはインターロックではない。本稿の文脈では「ある止まったピース配置 \mathcal{A} 」に対してインターロック性の判定を行っていたので、Coordinate Motion モデルの連続的な動きに対しては必ずしも正しく判定ができなかったのであった。

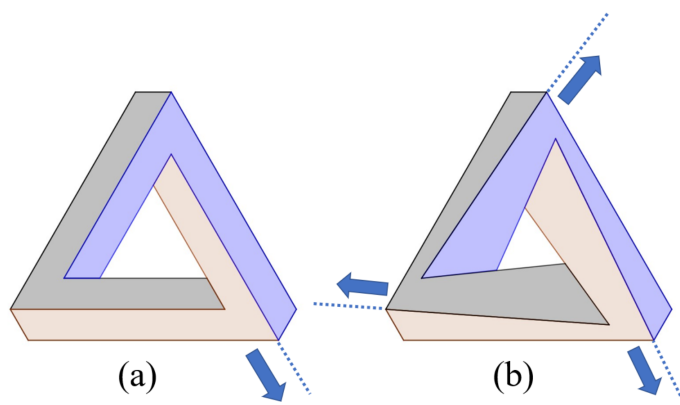


図 7.5: Coordinate Motion モデル（インターロック）

外部発表

1. 南澤 洸, 上原 隆平, 原 正雄. アンチスライドパズルの数学的特徴付けと計算量的複雑さ, IEICE Technical Report, COMP2020-13, pp. 10-17, 2020/10/23.
2. Ko Minamisawa, Ryuhei Uehara, and Masao Hara. Mathematical Characterizations and Computational Complexity of Anti-Slide Puzzles, The 15th International Conference and Workshops on Algorithms and Computation (WALCOM 2021), LNCS Vol. 12635, pp. 321-332, 2021/02/28-03/02, Yangon, Myanmar (Online). DOI:10.1007/978-3-030-68211-8_26
詳細 : <https://www.uit.edu.mm/walcom-2021/> or <http://www.jaist.ac.jp/~uehara/walcom-2021/>
論文 : https://link.springer.com/chapter/10.1007/978-3-030-68211-8_26
発表動画 : <https://youtu.be/xiauJ2I7bBs> (Accessed December, 2021)
3. Ko Minamisawa, Ryuhei Uehara, and Masao Hara. Mathematical Characterizations and Computational Complexity of Anti-Slide Puzzles, Theoretical Computer Science, submitted, June 2021.

謝辞

まず第一に指導教員の上原隆平教授には、論文執筆を通して形式的な文章の書き方、国内の研究会・国際会議での研究成果の発表を通して初見の人にも伝わる発表指導をしてもらいました。パズルの国際会議 IPP39 と金沢パズル博覧会、理論計算機科学の国際会議 WALCOM 2021 では上原教授が指揮を取ったプロジェクトの運営の補助を経験させていただきました。また研究以外でも、JAIST で管理する、約 1 万点に及ぶ世界的なパズルコレクションである NOB コレクションの整理を、謝金業務として研究室のメンバーと JAIST 広報係に任せてもらったので、実際のパズルに触れて愛着が湧き、パズルの計算複雑性の論文も読みやすくなりました。研究活動を通して特筆すべき指導として「2次元版アンチスライドパズルの議論において、なぜ重力の議論が必要なのか？（観察3.3）」について、私が納得するまで根気強く（2年間）説明し続けてくれたことがありました。最後に、このアンチスライドパズルの研究は、上原教授がパズル業界と理論計算機科学業界の両方に精通しており、膨大な知識があったからこそできたものでした。深く感謝申し上げます。

金子峰雄教授と藤崎英一郎教授は、修士論文中間発表会において、本稿の主結果の1つである定義3.1「『支える』の定義」と定義3.7「『インターロック性』の定義」に有益な指摘をしてくださいました。私の学部時代の指導教員である東海大学理学部情報数理学科の原正雄准教授は、私に（主に）グラフ理論の指導をして文献 [38] を教えてくれたり、セミナーを通して学んだことの延長として上原研究室を紹介してくれました。私が JAIST に進学してからも付き合いは続き、最終的には上原教授と私との共著論文を書いてくれました。電気通信大学大学院情報理工学研究科 情報理工学域情報・ネットワーク工学専攻 I 類（情報系）の岡本吉央教授は、定理6.3「凸多角形集合はいかなる配置に対してもインターロックできない」ことのエlegantな証明のアイデア [34] を教えてくれました。

JAIST 永井研究室の博士後期課程の谷口俊平氏は、私が研究で扱うパズルの制作において CAD, 3D プリンタ, レーザーカッターなどの技術的指導をしてくださいました。Twitter 上での交流において、谷井（たに~, @y_tanii1970）氏はパズルに関する私のどんな質問にも答えてくれて、幾多のパズルの知識を与えてくれ、ロボティクスの研究者 Alexey V. Ivchenko (@AlexeyIvchenko1) 氏は Coordinate Motion について興味深い GIF ファイルを作ってくれました。

数多くのパズル作家ら（特に、Wil Strijbos 氏, Vladimir Krasnoukhov 氏, 山本浩氏, 勝元甫氏, Philippe Dubois 氏, Stephen Chin 氏, George Bell 氏, 上原研究

室の神崎勝彰氏)は数理的に魅力のあるパズルを考案してくれたので、本稿を書く動機になりました。

セミナーで議論をしてくれた上原研究室の皆様(特に、英文添削をしてくれた上原研究室研究補助員の谷口智子氏、2021年度の上原教授の(半)サバティカル休暇期間中に研究室のセミナー運営をしてくれた **Giovanni Viglietta** 助教)にも感謝申し上げます。

様々な研究室の中国人留学生たちは、研究の息抜きとして卓球の相手をしてくれました。JAISTの外にいる学部時代からの友達は、オンライン越しに私の発表練習を聞いてくれたり、他分野の研究をしている彼からの視点で議論をしてくれました。また、大学院修士課程修了まで長年の面倒をみてくれた家族にも同様に感謝申し上げます。

皆様のおかげでよりよい修士論文が完成しました。

参考文献

- [1] Kazuyuki Amano, Shin-ichi Nakano and Koichi Yamazaki. Anti Slide, *Journal of Information Processing*, Vol. 23, No. 3, pp. 252–257, 2015.
- [2] Kento Kimura, Kazuyuki Amano and Tetsuya Araki. On the Minimum Number of Pieces for Two-Dimensional Anti-Slide Using T-Tetrominoes, *IEICE Transactions on Information and Systems*, Vol. E104-D, No. 3, pp. 355–361, 2021.
- [3] Yasuhiko Takenaga, Xi Yang, and Asuka Inada. Anti-Slide Placements of Pentominoes, *The 22nd Japan Conference on Discrete and Computational Geometry, Graphs, and Games (JCDCGGG)*, pp. 121–122, 2019.
- [4] Ko Minamisawa, Ryuhei Uehara, and Masao Hara. Mathematical Characterizations and Computational Complexity of Anti-Slide Puzzles, *The 15th International Conference and Workshops on Algorithms and Computation (WALCOM 2021)*, LNCS Vol. 12635, pp. 321–332, 2021.
- [5] Ko Minamisawa, Ryuhei Uehara, and Masao Hara. Mathematical Characterizations and Computational Complexity of Anti-Slide Puzzles, *Theoretical Computer Science*, submitted, 2021.
- [6] Jerry Slocum. *The Tangram Book: The Story of the Chinese Puzzle with Over 2000 Puzzle to Solve*, Sterling Publishing, 2004.
- [7] ソロモン・ゴロム (著), 川辺治之 (訳). 箱詰めパズル ポリオミノの宇宙, 日本評論社, 2014.
- [8] 杉原厚吉. ペンローズの不可能図形, 数学セミナー 2021年3月号 pp.25–29, 日本評論社, 2021.
- [9] 上原隆平. ペンシルパズル, 数学セミナー 2021年6月号 pp.78–79, 日本評論社, 2021.
- [10] 上原隆平. アンチスライドパズル, 数学セミナー 2021年12月号 pp.88–89, 日本評論社, 2021.

- [11] マーティン・ガードナー (著), 岩沢宏和, 上原隆平 (監訳). 完全版マーティン・ガードナー数学ゲーム全集2 ガードナーの数学娯楽, 日本評論社, 2015.
- [12] 徳山豪. 計算下界の解明 – その意義とシナリオ<前編>, 情報処理, Vol. 54, No. 4, pp.374–384, 2013. https://researchmap.jp/read0065631/misc/31636571/attachment_file.pdf.
- [13] 後藤新. Hoffman パズルの解の列挙と一般化に関する研究, 北陸先端科学技術大学院大学修士論文, 2011.
- [14] 岩井仁志. シルエットパズルの凸配置の個数の研究, 北陸先端科学技術大学院大学修士論文, 2016.
- [15] 樋本喬. 内外相似パズルにおける困難性の証明と効率的なアルゴリズム, 北陸先端科学技術大学院大学修士論文, 2019.
- [16] Eli Fox-Epstein, Kazuho Katsumata, and Ryuhei Uehara. The Convex Configurations of “Sei Shonagon Chie no Ita,” Tangram, and Other Silhouette Puzzles with Seven Pieces, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, Vol.E99-A, No.6, pp.1084-1089, 2016.
- [17] Erik D. Demaine, Matias Korman, Jason S. Ku, Joseph S. B. Mitchell, Yota Otachi, Andre van Renssene, Marcel Roeloffzen, Ryuhei Uehara, and Yushi Uno. Symmetric Assembly Puzzles are Hard, Beyond a Few Pieces, *Computational Geometry: Theory and Applications*, Vol. 90, pp. 101648:1–11, 2020.
- [18] Erik D. Demaine, Martin L. Demaine and Ryuhei Uehara. Any Monotone Function Is Realized by Interlocked Polygons, *algorithms* 5, pp.148–157, 2012.
- [19] Erik D. Demaine and Martin L. Demaine. Jigsaw Puzzles, Edge Matching, and Polyomino Packing: Connections and Complexity, *Graphs and Combinatorics*, 23(Supplement): pp.195–208, 2007.
- [20] Jhon E. Hopcroft and Richard M. Karp. An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs, *SIAM J. Computing*,2(4): pp. 225–226, 1973.
- [21] Takashi Horiyama, Takehiro Ito, Keita Nakatsuka, Akira Suzuki and Ryuhei Uehara. Packing Trominoes is **NP**-Complete, **#P**-Complete and **ASP**-Complete, *The 24th Canadian Conference on Computational Geometry (CCCG 2012)*, pp. 219–224, 2012.
- [22] Dania El-Khechen, Muriel Dulieu, John Iacono and Nikolaj van Omme. Packing 2×2 unit square into grid polygons is **NP**-complete, *The 21st Canadian Conference on Computational Geometry (CCCG 2009)*, pp. 33–36, 2009.

- [23] Yiping Lu, Danny Z. Chen, Jianzhong Cha. Packing cubes into a cube is **NP**-complete in the strong sense, *Journal of Combinatorial Optimization*, 29, pp. 197–215, 2015
- [24] Wil Strijbos. Anti-Slide, 1994.
https://johnrausch.com/PuzzleWorld/puz/anti_slide.htm, (Accessed December, 2021) (商品版：明治サイコロキャラメルパズル, ハナヤマ, 2007.)
- [25] Vladimir Krasnoukhov. Stop-Puzzle, 2014.
https://puzzles.schwandtner.info/puzzle_1518.html,
http://torito.jp/shopping/item.cgi?_stop-puzzle,
 (Accessed December, 2021)
- [26] 山本浩. Lock Device, 2012.
<https://puzzleworld.org/DesignCompetition/2012/>, (Accessed December, 2021)
- [27] Stewart T. Coffin. The Puzzling World of Polyhedral Dissections: Chapter 12 - Coordinate-Motion Puzzles, 1990–2012.
<https://puzzleworld.org/PuzzlingWorld/chap12.htm>,
 (Accessed December, 2021)
- [28] Philippe Dubois. Kaleidos, 1980s.
<https://www.puzzlemuseum.com/catalogue/index.html>で E976 を入力して検索, (Accessed December, 2021)
- [29] Stephen Chin. 1 Pinko Ringo, 2010.
<https://johnrausch.com/DesignCompetition/2010/default.htm>,
<https://www.boxesandbooze.com/boxesandbooze/2017/04/fruits-of-labor.html>,
 (Accessed December, 2021)
- [30] George Bell and Stephen Chin. Exploding Ball, 2012.
<https://www.youtube.com/watch?v=E4Y9rU-EVt0&t=0s>,
<https://www.youtube.com/watch?v=SiGyp4JdSZc&t=0s>,
 (Accessed December, 2021)
- [31] 田中浩也, 館知宏. コンピューテーショナル・ファブリケーション 「折る」「詰む」のデザインとサイエンス, 彰国社, 2020.
 FabCafeTV, CONNECTING ARTIFACTS つながるかたち展01 クロージングトーク, 2021. https://youtu.be/QHq_np4g76I?t=832,
 Tomohiro Tachi. Miura & Tachi Rigid Foldable Cylinder Tessellation, 2010.
<https://youtu.be/e4EErkJInHg>, (Accessed December, 2021)

- [32] 組合せ遷移プロジェクト, <https://core.dais.is.tohoku.ac.jp/>, (Accessed December, 2021).
- [33] Robert E. Tarjan. Depth First Search and Linear Graph Algorithms, *SIAM Journal on Computing*, Vol.1(2), pp.146–160, 1972.
- [34] Herbert Edelsbrunner, Arch D. Robinson and Xiao–Jun Shen. Covering Convex Sets with Non-Overlapping Polygons, *Discrete Mathematics* 81, pp.153–164, 1990.
- [35] M. Garey and S. Johnson. *COMPUTERS AND INTRACTABILITY: A Guide to the Theory of NP-Completeness*, FREEMAN, 1979.
- [36] M. ドバーク, O. チョン, M. ファンクリベルト, M. オーバーマーズ (共著), 浅野哲夫 (訳). コンピュータ・ジオメトリ 第3版, 株式会社近代科学社, 2010.
- [37] T. コルメン, C. ライザーソン, R. リベスト, C. シュタイン (共著), 浅野哲夫, 岩野和生, 梅尾博司, 山下雅史, 和田幸一 (共訳). アルゴリズムイントロダクション 第3版, 株式会社近代科学社, 2013.
- [38] 根上生也. 離散構造, 共立出版株式会社, 1993.
- [39] 荻原光徳. 複雑さの階層, 共立出版株式会社, 2006.
- [40] 渡辺治. 計算可能性・計算の複雑さ入門, 株式会社近代科学社, 1992.