

論文名 Title	45°系格子パターンから作り出される 平坦折り形状の列挙	Enumeration of flat-folded shapes generated from the 45 degree grid system
著者 Author(s)	山本陽平 三谷純	Yohei Yamamoto, Jun Mitani
受理年月日 Date of acceptance	2015/2/19	
掲載 First publish	『折り紙の科学』（"Science of Origami"）2015/3/31 Vol. 4 No. 1 page 23-33	
備考 Note		

日本折紙学会
Japan Origami Academic Society
www.origami.jp

45° 系格子パターンから作り出される平坦折り形状の列挙

山本陽平 三谷純

筑波大学

305-8573,茨城県つくば市天王台 1-1-1

yamamoto@npal.cs.tsukuba.ac.jp

mitani@cs.tsukuba.ac.jp

Enumeration of flat-folded shapes generated from the 45 degree grid system

Yohei Yamamoto, Jun Mitani

305-8573, Tennoudai 1-1-1, Tsukuba, Ibaraki, JAPAN

要約 垂直および水平な直線で正方形の各辺を等分し，その結果得られる小さな正方形の各領域に対して対角線を追加したパターンを 45° 系格子パターンと呼ぶ．古典折り紙として広く知られる風車やだまし船などの展開図を見ると，その折り線は，この格子パターンに重ねることができる．このように 45° 系格子パターンには，基本的な折り紙の展開図が数多く含まれる．本論文では正方形の各辺を 4 等分して得られる 45° 系格子パターンの中に含まれる局所平坦折り可能条件を満たす折り線パターンの配置を列挙し，それを平坦に折った後に現れる形を列挙する方法を提案する．提案手法によって，13,451 通りの異なる形が得られたことを報告する．

Abstract: Vertical and horizontal lines which equally divide edges of a square make small square cells. We call the pattern which consists of diagonal lines of all the small square cells and the vertical and horizontal lines a 45 degree grid system. As we can find the crease patterns of the windmill and the trick boat, that are well known traditional origami, in this grid system, the 45 degree grid system contains a number of crease patterns of basic origami pieces. We enumerated all the patterns which satisfies the local flat-foldable conditions in the grid system whose size is 4x4. Further, we enumerated all shapes obtained by folding the patterns. The number of unique shapes we found is 13,451.

Keyword: 45 degree grid system, Flat-foldable origami, Enumeration

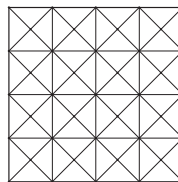


図 1 45° 系格子パターン．正方形の各辺を垂直および水平に等分する線分と，その結果得られる小さな正方形の各領域の対角線で構成される

1. はじめに

近年では折り紙の設計理論を計算機に実装することで、意図したものに近い形を1枚の紙から折り出すことが可能となった[1-3]. その一方で、折り紙設計以前の折り紙創作は、試行錯誤の中で独創的な形を多数生み出してきた事実があり、新しい作品の創出には、旧来の発見的なアプローチも不可欠であるといえる. 近年の計算機の性能の向上によって、さまざまな対象に対して、取り得る全ての組み合わせを現実的な時間で列挙することが可能となってきているため、折り紙においても、折り操作を限定すれば、その中で折り出すことが可能な形をすべて調べ上げることが可能である. 例えば、鶴田らは頂点と頂点、辺と辺を合わせるように折る操作を4回まで行った場合に得られる形を列挙した[4]. 本論文では、図1に示す 45° 系格子パターンに含まれる展開図に限定して、平坦に折った時に現れる形をすべて列挙する手法と、その結果について述べる. 本論文で提案する手法は、試行錯誤の部分を計算機に代行させるものであり、その結果として、今まで作品として認識されてこなかった形の発掘を目指したものである.

本論文では垂直および水平な直線で正方形の各辺を等分した線分と、その結果得られる小さな正方形の各領域に対して対角線を追加した線分の集合を 45° 系格子パターンと呼び、とくに正方形の各辺を4等分して得られるパターンから作り出される形に着目する. この格子パターンは辺と辺を合わせる折り操作によって容易に作り出すことができ、多くの折り紙作品の基本形が含まれるため、折り紙創作には重要なパターンである. たとえば、ざぶとんの基本形、風船の基本形、二艘船の基本形といった折り紙の基本形の他、家、二艘船、風車、だまし船など、すでに知られている様々な折り紙の展開図が、この格子パターン含まれており(図2)、これら以外にも多様な形を折り出すことができる. なお図2の例には見られないが、図5の右端に示すような4価頂点を含む折りも許容する. また、この格子パターンは直角三角形のパネルを繋ぎ合わせたロボットが、ヒンジ部で折り曲がることで変形するHawkesらの研究[5]にも見られ、本研究の成果はこのようなロボットの新たな変形パターンの構築にも関係する.

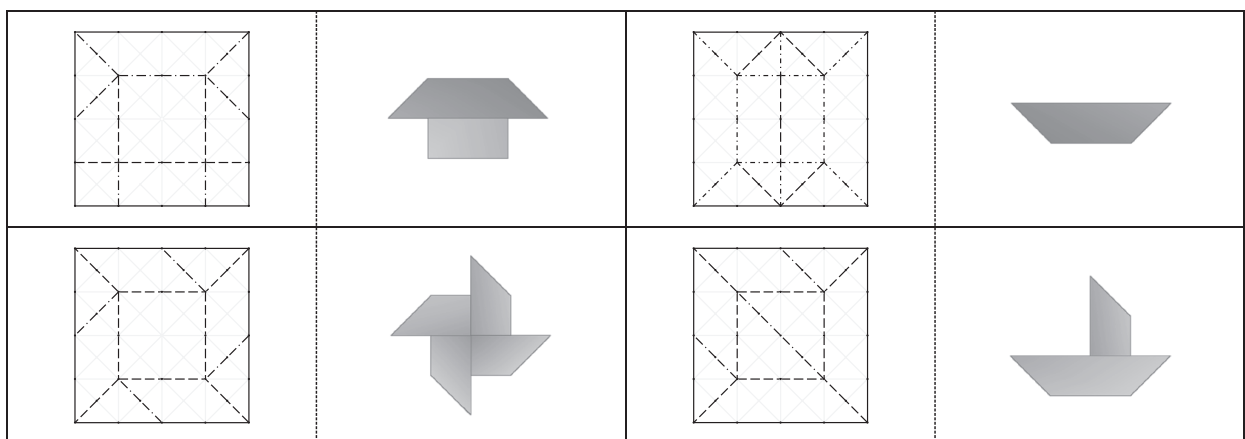


図2 45° 系格子パターンに含まれる展開図の古典折り紙の例
 左上) 家 ; 右上) 二艘船 ; 左下) 風車 ; 右下) だまし船

本研究では平坦に折り畳んだ後の形の列挙を行うために、まず平坦に折り畳み可能な展開図の列挙を行う。展開図が平坦折り可能であるための条件には、次に示すように内部頂点（展開図に含まれる折り線が交差する点のうち、展開図の輪郭上に含まれないもの）における折り線の山谷の割り当てに関する前川定理と、折り線間の成す角に関する川崎定理がある（図3）。

- ・「山折り」と「谷折り」の数の差は ± 2 （前川定理）
- ・1つおきの内角の和は 180 度である（川崎定理）

これらは1つの頂点周りに注目した局所的な定理であり、展開図全体が折り畳めるか否かについては言及していない。

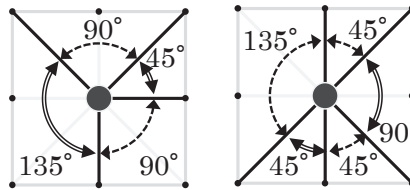


図3 局所平坦折り条件を満たす例(左)と満たさない例(右)

平坦に折り畳むことができる大域的な山谷の割り当てに関しては、川崎が文献[6]で詳しい考察を示している通りに難しい問題であり、それがNP困難であることがBernらによって示されている[7]。そこで本研究では山谷の割り当てについては議論せず、折り線の配置についてのみ議論する。これは川崎が「形式的折り線図」と呼んでいるものに相当する。つまり、以降で局所平坦折り条件について言及するときには、折り線間の角度に関する川崎定理のみを考慮し、山谷の割り当てに関する前川定理については考慮しない。そのため、本研究で列挙する折り線パターンは、次の2つのいずれかの理由により、実際には折ることができない可能性がある。

(1) 全ての内部頂点が同時に前川定理を満たすような山谷の割り当てができない。

(2) 紙を折る過程における大域的な自己干渉によって、平坦に折り畳むことができない。

しかしながら、対象とする格子パターンに含まれる折り線の数は少数に限定されることと、規則的な配置をしていることから、列挙されたパターンは全て平坦に折り畳めるのではないかと筆者らは予想する。これについては2.3節で詳しく述べる。

以降では、 45° 系格子パターンに含まれる局所平坦折り可能条件を満たす折り線パターン（形式的折り線図）を列挙し、それを折った後に現れる形を計算で求める手法について述べ、その結果と考察を述べる。なお列挙の際には、回転や反転の操作を行うことで一致する形は除外するものとし、単に展開図と記した場合には、山谷の区別がされていない折り線図を指すこととする。

2. 提案手法

以下の手順によって、格子パターンに含まれる展開図を平坦折りすることで得られる形の列挙を行うことができる。

- (1) 局所平坦折り条件を満たす展開図を列挙する。
- (2) 列挙した展開図を折り畳むことで得られる形を算出し、その結果を列挙する。

(3) 列挙した形の内，その形に折り畳まれる展開図の全てが（異なる展開図が同じ形に折り畳まれることがあることに注意されたい），次の2つの条件のいずれかを満たす場合，その形は折り畳むことができないと判断し，(2)の結果から取り除く．

(ア) 全ての内部頂点に対して，矛盾なく局所平坦折り条件を満たすような山谷の割り当てができない

(イ) 折る際に大域的な自己干渉を起こし，平坦に折り畳むことができない

本研究では，(1)および(2)をおこなうシステムを開発した．以降ではその手法について述べる．(3)に示した，実際には折れない場合の除外は行っていないため，折り畳むことができない形が列挙された結果に含まれる可能性がある．

2.1. 局所平坦折り条件を満たす展開図の列挙

局所平坦折り条件を満たす展開図を列挙するために，まず，その数の上界について議論する．格子パターンの内部には，線分が88本含まれ，各線分に対して折り線の有無を割り当てたものが局所平坦折り条件を満たす展開図の候補となるため，単純に考えると 2^{88} 通りが上界となる．この全てのパターンについて平坦折り可能性の判定をすることは現実的でないため，この値を小さくすることを検討する．

対象とする格子パターンには，内部頂点が25個だけ存在し(図4左)，内部頂点に接続する線分の本数によって，16個の4価頂点と，9個の8価頂点に分類できる．

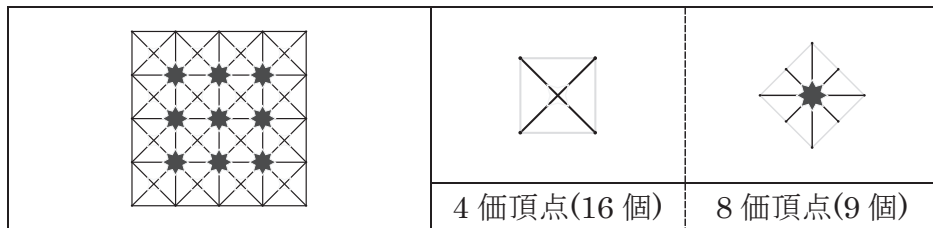


図4 左) 45° 系格子パターンの内部頂点；右) 内部頂点の種類

4価頂点に接続する線分に対して，局所平坦折り条件を満たす折り線の割り当ては図5に示す4通りしか存在しない．このことから，4価頂点においては，同一直線上に乗る2本の線分を連結して1本と見なすことができる．そうすると，88本あった線分を実質的には $56 (=88-16 \times 2)$ 本に減らすことができ，上界を 2^{56} 通りまで小さくできる．各線分に対する折り線の有無は1bitで表すことができるので，このように線分の本数を減らすことで，1つの展開図パターンを，一般的なPCで扱うことができる64bitの整数型の値として格納できるようになる．

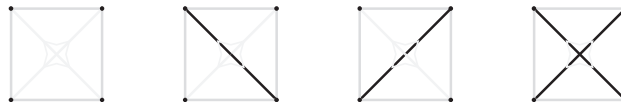


図5 4価頂点における局所平坦折り条件を満たす折り線の割り当て

8価頂点に接続する線分に対して，局所平坦折り条件を満たす折り線の有無の割り当ては，図6に示す36通りだけ存在する．

本論文で提案する手法では、すでに述べた 4 価頂点，および 8 価頂点における局所平坦折り可能なパターンを組み合わせることで，局所平坦折り条件を満たす展開図を列挙する．具体的には，次の手順に従う．

- (1) 格子パターン上に存在する 9 つの 8 価頂点に対して，図 6 に示した 36 通りの折り線の割り当てを順次当てはめていく．その際に，頂点間で共有される線分(図 7 に示す 26 本の点線)において，矛盾が無いことを制約とする．矛盾が生じた時点で以降の当てはめを省略する（枝刈りする）ことで，現実的な時間で全てのパターンを列挙できる．4 価頂点については，既に述べたように同一直線上に乗る 2 本の線分を連結して 1 本と見なすため，考慮しなくてよい．
- (2) 前項の処理では 4 隅の線分（図 6 の一点鎖線）が，対象外となっていたため，得られたパターンに対して，4 隅の折り線の有無（都合 16 通り）を組み合わせたものを，最終的な展開図とする．
- (3) 得られた展開図に対して ± 90 度， 180 度の回転，および反転の有無の組み合わせ操作を行い，その結果が既に見つけ出されていない場合のみ列挙する．

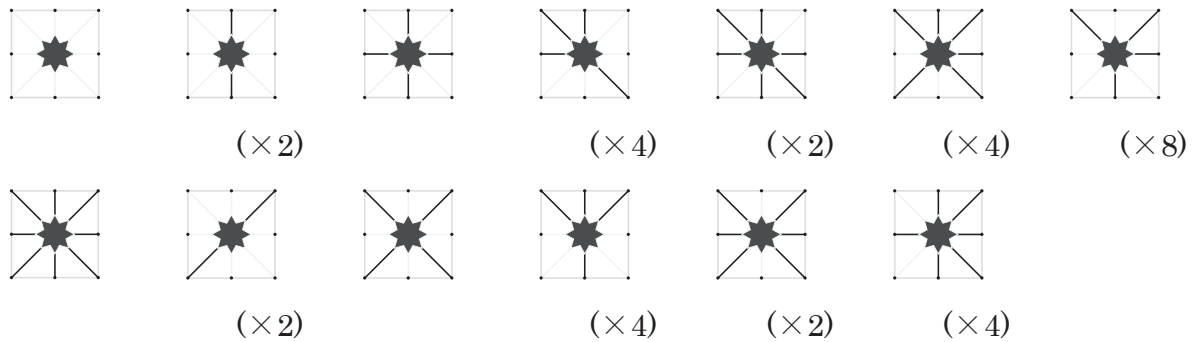


図 6 8 価頂点における局所平坦折り条件を満たす折り線の割り当て
(回転させて重なるものは省略．括弧内の数字は，それがいくつあるかを示している.)

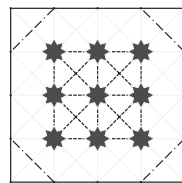


図 7 平坦折り可能な展開図の作成に用いるガイド．破線部で示した折り線は複数の内部頂点に連結する．一点鎖線で示した折り線はどの内部頂点にも連結しない．

2.2. 折り畳み後の形の算出

与えられた展開図を折り畳むことで得られる形の算出は，三谷が公開しているオープンソースプログラムである ORIPA[8]を用いた．ORIPA は与えられた展開図を折り線で区切られた多角形に分割し，折り線を軸とする鏡映変換によって重ねることで，平坦に折り畳んだ際の形を出力する．より具体的には，次のような処理を行っている．まず，多角形領域をノードとし，折り線を介して隣接するノード間をエッジとするグラフ構造から，極大木を作り，あるノードを始点として全てのノードを巡回する．その際に，ノードからノー

ドへ移るときに、移動先の多角形をエッジに相当する折り線で鏡映反転する。

本論文では、この機能に対して以下の2つの変更を加えることで、展開図を折り畳んだ形の算出を行った。

- (1) 座標値を 32bit の整数型で扱う
- (2) 折り畳んだ形を 64bit の整数型で表す

ORIPA は与えられた展開図の座標値を小数点で取り扱うことで、自由な展開図の入力を可能にしている反面、鏡映反転等の幾何演算で座標値に数値誤差が生じる可能性がある。本研究では得られた形が他の形と重複するか否かを判定するために、誤差の無い状態で形が得られることが望ましい。ここで、入力する展開図は格子パターンに含まれることに着目すると、各頂点座標は等間隔に位置するため、ORIPA が取り扱う座標値を整数型で扱うことが可能であり、誤差の無い形を得ることができる。

また、ORIPA は折り畳んだ形を描画処理によって出力するため、得られた形が既に算出された形と同一であるか否かを判定するために、形を記述するための方法が必要である。そこで、折り畳んだ形もまた格子パターンに乗る直角二等辺三角形の集合であることに着目し、64 個ある三角形の何番目の三角形を用いて構成されているかを調べ、形同士と比較に用いることとした。このようにすることで、任意の形を 64bit の整数型の値 1 つで表現できる。

位置のずれや回転、反転の違いを除くために、得られた形は格子の最も左上に配置する(図 8 左)。続いて、 ± 90 度、 180 度の回転、および反転の有無の組み合わせ操作を行い、それらを再び左上隅に配置することで、都合 8 種類の整数型の値を得る。これらの値の中で最大となるものを、その形を示す値とし、その結果が既に発見されていない場合のみ、新しく見つかった形と判定する。また、外形だけを見たいため、形に含まれる頂点が異なるものが複数ある場合(図 8 右)は、各形を示す値の内、最大となるものをそれらの形を示す値とする。

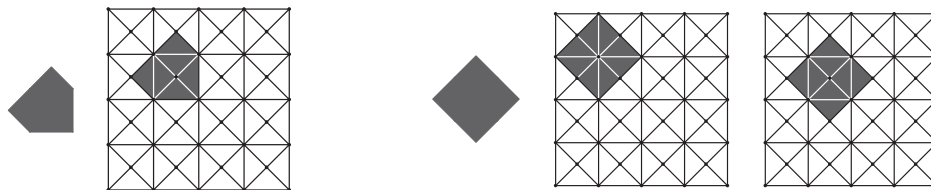


図 8 平坦に折り畳んだ形を格子パターンの左上に寄せて重ねる例；左) 形に含まれる頂点が一意に定まる例 右) 形に含まれる頂点が複数考えられる例

2.3. 折り畳めない形の除外と予想

2.1 節で得た展開図からは 2.2 節で述べたように折り畳み後の形を算出できるが、本章冒頭に示した手順(3)の判定処理を行わない限りは、実際にはその形を作りだせない可能性がある。

しかしながら、1 つの展開図に対しても山谷の割り当て方は折り線の数を 2 の指数とする膨大な数の組み合わせがあり、また次の章で示す通り、1 つの形を複数の異なる展開図から作ることもできることから、著者らは列挙された形はいずれも実際に折ることができるであろうと予想する。この予想が正しいことを示すには、列挙された形が大域的にも折り

畳めるような山谷の割り当て付きの展開図を、全ての形に対して少なくとも1つを示せばよいが、本研究の範囲外としており今後の課題である。この予想を否定するには、1つでも反例（実際には折れない形）を示せばよいが、今のところ見つかってない。

なお、川崎[6]によって9x12、館¹によって5x7の45°系格子パターン内に、(3)(ア)に記した条件（全ての内部頂点に対して、矛盾なく局所平坦折り条件を満たすような山谷の割り当てができない）を満たすような例（図9）が示されているが、それを本論文が対象とする格子パターン上に配置することはできない。

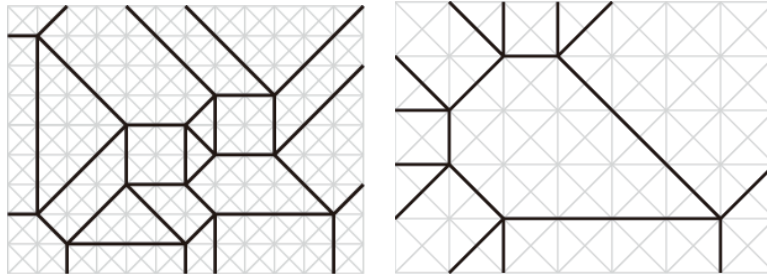


図9 全ての内部頂点に矛盾なく局所平坦折り条件を満たす山谷の割り当てができない展開図の例。左は川崎、右は館による。

3. 結果と考察

提案システムで列挙したデータの詳細を示す。システムの実装には局所平坦折り可能な展開図の列挙にはC++言語を用いてCore i7 @2.00GHz, RAM8GBを搭載したPC上で動作させ、折り畳み後の形状の列挙にはJavaを用いて、Core i7 @2.40GHz, RAM8GBを搭載したPC上で動作させた。得られたデータは、どちらも1つにつき64bitの整数型とした。

3.1. 展開図の列挙

展開図の列挙で得られた要素数は259,650,300通りであり、計算に要した時間は約5時間40分であった。展開図中の56本の線分の内、33本の線分を用いた展開図が最も多く、25,566,613通りだけ存在した。得られた展開図の一例を図10に示す。図10の左は、2本の線分を用いた全ての展開図であり、右は33本の線分を用いた展開図の一部である。

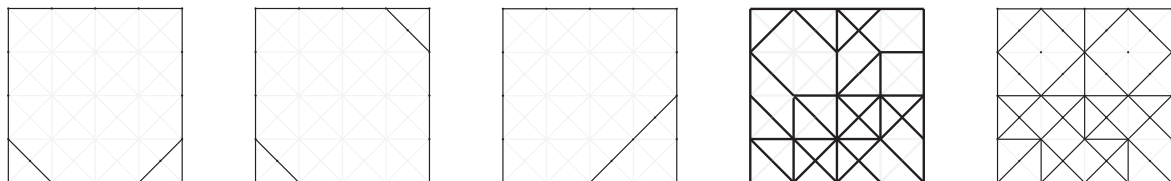


図10 2本の線分を用いた全ての展開図（左から3つ）と、33本の線分を用いた展開図の一例（右から2つ）

¹ 私信, 2014/12/14

3.2. 折り畳んだ形の列挙

実装したシステムで得られた形は 13,451 通りであり，要した時間は約 67 時間 40 分であった．展開図の数に対して，折りたたんだ形の数は，約 20,000 分の 1 に減少したことを確認できた．同じ形に折り畳まれる展開図が複数存在することがあり，その個数をみたときに，上位 10 種類の形を図 11 に示す．図 11 の最も左の形は 17,708,760 の展開図から作り出され，これは列挙された展開図の約 7% に相当する．対応する展開図の数が多いものから上位 10 種類の累積度数を図 12 に示す．上位 10 種類の形のいずれかに折りたたまれる展開図が，列挙された展開図の約 45% に相当することが確認できる．また，ただ 1 種類の展開図から折り出される形は 782 種類だけ存在することを確認した．その一部を図 13 に示す，全ての線を折って得られる形や，折る線の少ない形は自明であるが，ある程度の数の線を折るものの存在も確認できた．

17,708,760 種	15,414,564 種	14,390,242 種	13,009,613 種	12,624,178 種
11,945,801 種	9,636,341 種	8,097,514 種	5,732,605 種	5,489,934 種

図 11 展開図の種類数で見た上位 10 種類の形

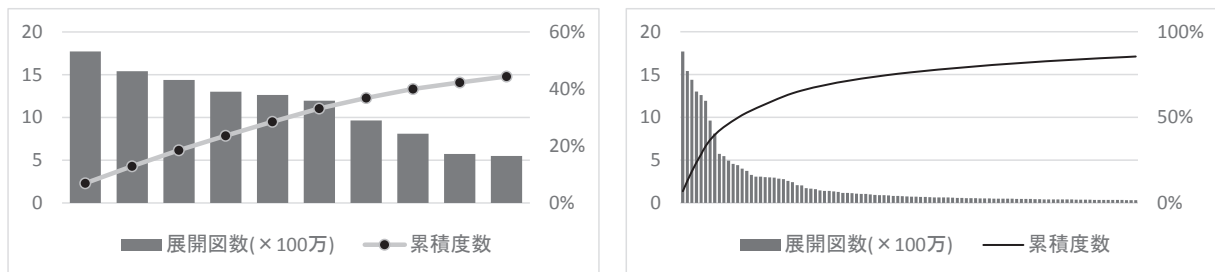


図 12 左)展開図の種類で見た上位 10 種類の形における展開図の累積度数，右)上位 100 種類の形における累積度数

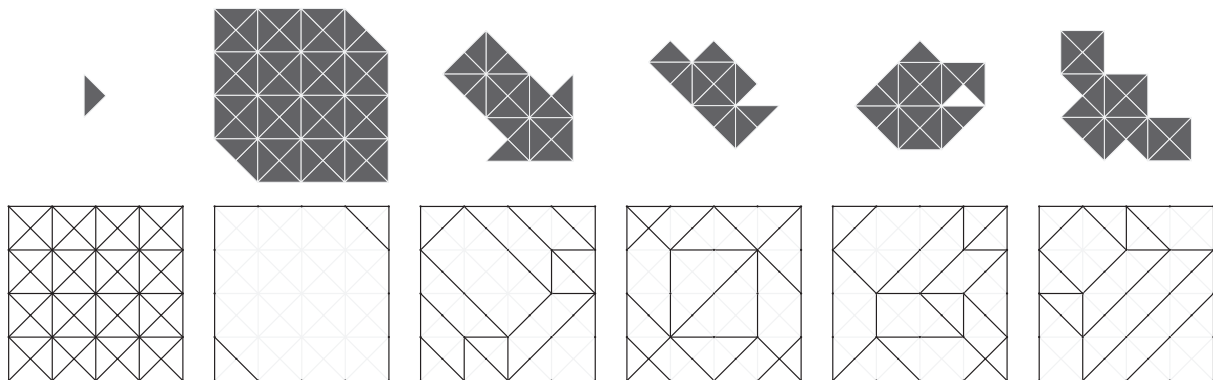


図 13 展開図が 1 種類しか存在しない形の一例と その展開図

3.3. 形の検索

列挙された形は1万を超えることから、全体を目視することは困難である。そのため、列挙された形の中に意図した形が含まれるか判定するために、形の検索を行うシステムの実装を行った。ここでは、 45° 系格子パターンに含まれる最も小さな直角三角形を基本三角形と呼ぶこととする。実装したシステムでは、基本三角形を組み合わせることで形を入力し、その形に折りたたまれる形が列挙した中に含まれるならば、その形に折りたためる全ての展開図を出力する。

検索に用いるデータは次のようにして構築した。まず、1つの形に対して、1つのバイナリファイルを作成した。ファイル名をその形を表す整数とし、ファイル内部には、その形を折れる展開図を64bitの固定長データとして並べた。検索システムは、入力された形を表すファイル名を検索し、ファイルが見つければ、その内部に格納されている展開図を先頭から出力する。

このシステムにより、意図した形を格子パターンから作ることができるか否かを判定でき、できる場合には、その展開図を出力することが可能となった。この検索システムを用いて見つけた、 45° 系格子パターンから作ることができるアルファベットと数字を図14に示す。これらは、正方形の紙から実際に折れることを確認した。

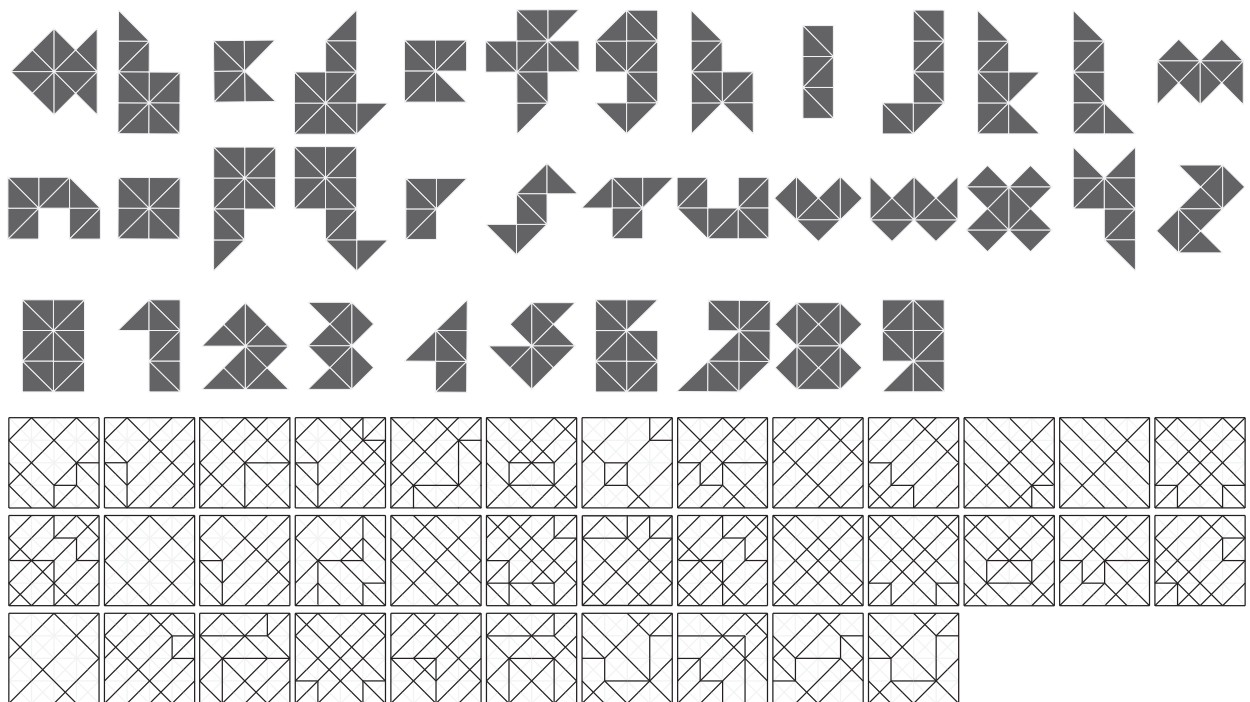


図 14 上) 実際に折りたためるアルファベットと数字, 下) その展開図

図14に示したものは全て円板と同相であるが、内部に穴が開いたものもいくつか確認できた。確認された形は、図15左のように、穴の一部が外部に頂点で接しているものに限られ、図15右のように、穴の頂点も完全に内側に存在するものは存在しなかった。図15右に示す形を折るには、穴を完全に囲うために（折り返しを考慮して）14枚より多くの基本三角形が直線に連結した形が必要であるが、今回対象とした格子パターンで直線に連

結する基本三角形は最大でも対角線上の 14 枚であるため、このような形は折ることができないものと考察される。



図 15 左) 出力された穴の開いた形の例 右) 存在しない形の例

また、検索システムを用いることで、列挙された中には含まれない形を見つけることも可能となった。その例を図 16 に示す。つまりこれらは、格子パターンからは折り出すことができない形である。提案手法では大域的な折り畳み可能性の判定をしていないため、検索で見つかった形は、厳密には「折れる可能性が高い形」に過ぎない。その一方で、検索で見つからなかった形は「明らかに折れない形」である。このように、本システムを用いることで、ある形が格子パターンから作り出せないことを示すことも可能となった。



図 16 展開図が存在しない形の例

4. まとめ

本研究は、 45° 系格子からなる展開図に着目し、 4×4 の格子を敷き詰めたパターンに含まれる、山谷を考慮しない局所平坦折り条件を満たす展開図の全てと、それらを折り畳んで得られる形の全てを列挙した。その結果、展開図は 259,650,300 通り、折り畳むことで得られる形は 13,451 通り得られた。また、得られた形を検索するためのシステムを開発することで、アルファベットや穴の開いた形など、研究前には予想していなかった形を得られた。また、このシステムを用いることで、格子パターンからは折り出すことができない形も得られることを示した。

今後の課題は、 4×4 の 45° 系格子パターンの展開図は大域的に全て折りたためることを証明する、もしくは反例を示し、大域的に折り畳めることが保証される形のみを列挙することである。

文献

- [1] Robert J. Lang , TreeMaker 4.0: A Program for Origami Design,
<http://www.langorigami.com/science/computational/treemaker/treemaker.php>,
(2015/3/9 アクセス)
- [2] Tomohiro Tachi, "Origamizing Polyhedral Surfaces", IEEE Transactions on Visualization and Computer Graphics, Vol. 16, No. 2, pp. 298-311, 2010.
- [3] Jun Mitani, "Column-shaped Origami Design Based on Mirror Reflections", Journal for Geometry and Graphics, Vol.16, No.2, pp.185-194, 2012.
- [4] 鶴田直也, 三谷純, 金森由博, 福井幸男, "形状列挙に基づく幼児向け折り紙作品の創

- 作支援システム", 折り紙の科学, Vol. 2, No. 1, pp. 33-44, 2012.
- [5] E. Hawkes, B. An, N. M. Benbernou, H. Tanaka, S. Kim, E. D. Demaine, D. Rus, R. J. Wood, "Programmable matter by folding", Proceedings of the National Academy of Sciences, DOI: 10.1073/pnas.0914069107, 2010.
- [6] 川崎敏和, "バラと折り紙と数学と", 森北出版株式会社, 1998.
- [7] M. Bern, B. Hayes, "The Complexity of Flat Origami", Proceedings of the Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 175-183, 1996.
- [8] 三谷純, "折紙の展開図専用エディタ(ORIPA)の開発および展開図からの折りたたみ形状の推定", 情報処理学会論文誌, Vol. 48, No. 9, pp. 3309-3317, 2007.