# 一刀切り -One Complete Straight Cut- の理論と実践２

## １．はじめに

昨年の初めての作品展では、小さい頃からの自分の興味志向を振り返って、数学のパズル的な内容で取り組んでみたいという事からテーマを探し、折り紙数学の一分野である「一刀切り問題」に興味をそそられ取り組んだ。「一刀切り問題」とは昔から親しまれてきた紙切り遊びを題材に、どんな多角形が切り出し可能であるかを判定し、実際にそれを切る方法を見つける問題のことである。簡単な例では、正方形の折り紙を三度折りたたんで、ひと裁ちで星形を切り出すことを誰しも小さい頃にやってみたことがあるのではないかと思う。僕も七夕の飾りをこの方法で沢山作った記憶がある。折り紙は日本の文化と思っていたが、この問題に対して「折り紙をたたんで一度ハサミを入れるだけで、どんな形でも無限に作れる」という驚くような証明をしたのはカナダの Erik Demaine という若き天才数学者だ。僕は彼の記事[1)]を読んで非常にワクワクしたので、Erik の理論を理解することと、自分でもオリジナルの折り目設計に挑戦することをテーマに決め、昨年は「K」「E」「I」「O」と「ペンマーク」の切り出しに成功した。

今年は昨年の取り組みの中では自分にとっては難解で理解が中途だった部分や説明が足りなかった部分について、より詳しく学び解説を試みることと、昨年の１文字一刀切りより複雑な２文字一刀切り，更には４文字一刀切りの折り目設計に挑戦したい。そのために 昨年は Erik が運営するサイトや彼の著書の原書を主に参考にしていたが、日本の数学者による訳書等も読み込んで理解に努めたいと思う。また１文字の折り目設計は方眼紙を用いて手書きで行うことが出来たが、角度や辺の長さを精密にするには限界を感じていたので、今年は製図ソフトを使って取り組んでみたいと思う。

## ２．折り紙数理の基礎

### ２－１　数学上の１枚の紙とはなにか

数学的な紙では厚みは考慮されない。つまり普通の紙の厚みを０にしたものが数学的な紙である。任意の厚さのない平面であり、無限に多くの折り目を持つことが出来る。

### ２―２　折り紙の展開図

折り線は紙の上の線分であり、稜線が突き出るように折ることを山折りと呼び、へこむように折ることを谷折りと呼ぶ。複数の折り線が共有する点を頂点、頂点から伸びる折り線の数を次数と呼ぶ。折り線は頂点を境に山折りか谷折りのどちらかに割当てられなければならない。紙の上に描かれた折り線の集まりを展開図と呼ぶ。本研究では、目的の形を切り出す折り方の展開図を作ることを折り目設計と呼ぶこととする。そして出来た折り線と切り取り線を一刀切りの設計図と呼ぶことにする。

### ２－３　折り紙を平坦に折るということ

折り紙という遊びを考える時、鶴や風船といった立体的な作品を思い浮かべる人が多いと思う。１枚の２次元の紙から立体的な３次元の作品を作れるところが、折り紙の芸術的な素晴らしさだと思う。しかし、「一刀切り問題」に取り組むにはハサミを入れるために、紙は平坦に折られていなければならない。そして切り出される任意の多角形も平面である。

「一刀切り問題」とは「１枚の紙の上に描かれた平面グラフが与えられたとき、そのグラフの全ての頂点と辺が共通の線の上に乗り、しかもそれ以外のどの部分もその線の上には乗らないような紙の平坦折りを見つける」という命題に言い換えることが出来る。ここで平面グラフとは、平面上の任意の直線分の集まりである。この言い換えは、昨年「E」の折り目設計で苦労した失敗例の経験からよく理解できる。E の形の輪郭、つまりグラフ辺を無理に折り重ねた結果、グラフ辺以外の部分も共通線に重ねてしまい、その線上にハサミを入れることによって無残に切り刻まれた「E」になってしまった。

折り紙を平坦に折ることは、立体的な作品作りに比べると直感的には簡単なことのように思われる。しかし、与えられた展開図が平坦折り可能かどうかを判定する問題は NP 困難*であると Marshall Bern が示しているように、折り線が交わる点が複数存在する場合の紙全体の折りの可能性を考えることは難しい問題である。

但し局所的には以下の条件が必要であることがわかっている。
① 頂点の次数は偶数である　（偶数次数定理）
② 各頂点のまわりで、山折りと谷折りの本数の差は２である　（前川・Justin 定理）

＊NP 困難：問題に対する証拠 w が与えられた時、その証拠 w が本当に正しいかどうかを多項式時間すなわち現実的な時間では判定できない問題のこと。

## ３．一刀切りの設計手法

紙を折って、任意の多角形のグラフ辺を一つの共通線上に並べる方法は２種類報告されている。１つ目の方法は、昨年度の労作展でも使用した直線骨格法で、Erik が考案した。平面グラフの局所的な対称性を活用したものである。もう一方の方法は、Bern が考案したディスクパッキング法と呼ばれるものである。これは多角形を３角形と「良い」４角形に効率よく分解するものである。直線骨格法は、より単純な折り方になるが、ほとんど全ての場合を作図できるもので、万能な方法ではなかった。一方、ディスクパッキング法は全ての多角形に応用でき、この方法の考案によって一刀切り問題すなわち「紙を折りたたみ一回切ることで、いかなる多角形でも切り出すことができる」に対する完全なアルゴリズムが完成した。

### ３−１　木構造法

直線骨格法とディスクパッキング法という二つのアルゴリズムを考えるには、双方に密接な関連がある折り紙設計の木構造（tree method）をまずは理解しなければならない。一刀切りは求める多角形のグラフ辺のみを直線上に折り重ねていくので、その折りプロセスは蛇腹折りに近いものになる、ということは昨年の経験からも実感出来た。目的とする多角形の骨格構造を蛇腹折りを応用して上手く折り出すと、その骨格構造に対応するグラフ辺を同一平面に載せることができ、その時に全ての面がその平面に垂直になる、という特徴がある。このような性質を持つ基本形を作り出す手法は木構造と呼ばれ、その際に用いられる折り線の配置には、持ちたい枝の長さを半径とする円を敷き詰めるアルゴリズム（Circle packing）が必要となる。このようなアルゴリズムを持った Tree Maker という折り紙設計プログラムも木構造法の研究者である Lang によって無料で公開されていて、多くの複雑な折り紙作品が発表されている。
展開図と折った後に現れる木構造の対応の関係は以下の図がわかりやすい。折られた紙の輪郭線はそれぞれの色に対応した枝や幹になり、同一の平面上に乗っている。展開図の内部を見ると、枝の長さを半径とした円が敷き込まれていて、その円の中心が枝の先端になるように折り出すことが出来る。またその円の中心は紙の角である必要はなく、持ちたい枝の長さや位置に合わせて任意の位置から枝を作ることが出来る。またこのようにして作られた木構造は枝を平たくぶら下げることによって、平坦折りになる。

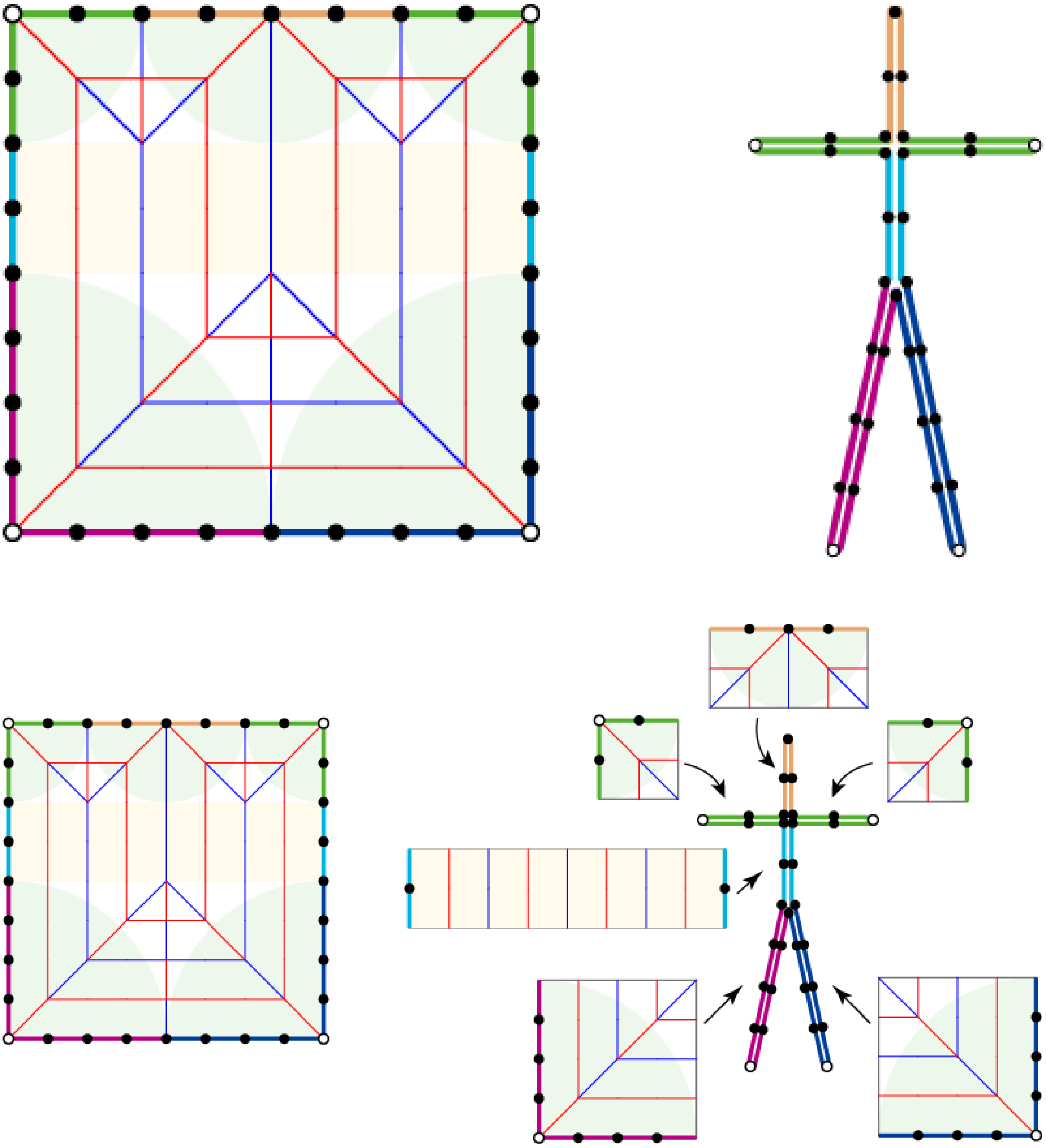

図１　木構造の一例

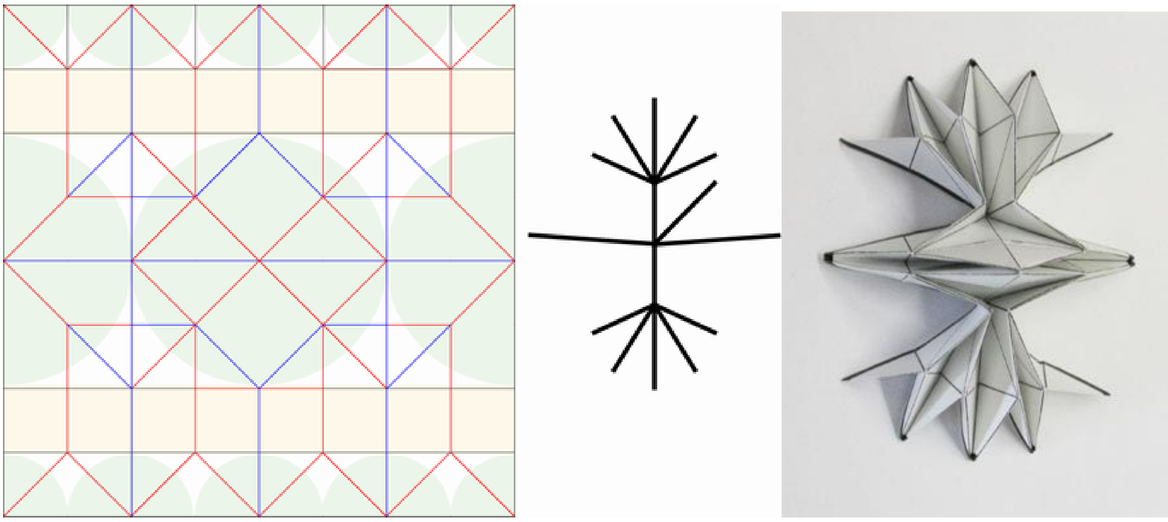

図２　複雑な木構造の一例

## ３−２　直線骨格法

直線骨格は多角形内の２つのグラフ辺の２等分線がいくつか組み合わさって構成される。それぞれの直線骨格辺は２本のグラフ辺を局所的に重ね合わせ、そして直線骨格全体は全てのグラフ辺を一直線上に重ね合わせることがイメージできると思う。
直線骨格の引き方は、角の２等分線を延長し、２等分線たちが最初に交わる交点に注目する。角の２等分線は、角を挟む辺からの距離が等しい。ひとつのグラフ辺の両端の２つの角の２等分線の交点は３つの辺からの距離が等しい。これら交点を結ぶ直線は、この対辺の延長の交点において、対辺のなす角の２等分線になっている。この線分のどの点からも対辺への距離は等しくなっている。
以上のことを一般化するには、任意の平面グラフのそれぞれの辺から垂直に等距離の点を取っていく。距離が０の時は多角形自身であり、距離が小さいうちは、元の多角形に似た少し小さい多角形になる。ある辺 e が縮小していって点になった時は、それ以降は e を無視して縮小を続ける。ある面が点になった時は、その面の縮小はそこで止める。図３-(d) のように頂点が辺に接触して、その結果ある面が単純でなくなった時は、その面を２つの成分に分割し、それぞれの部分ごとに縮小を続ける。この縮小のプロセスの間に頂点がたどる軌跡の和集合が直線骨格である。

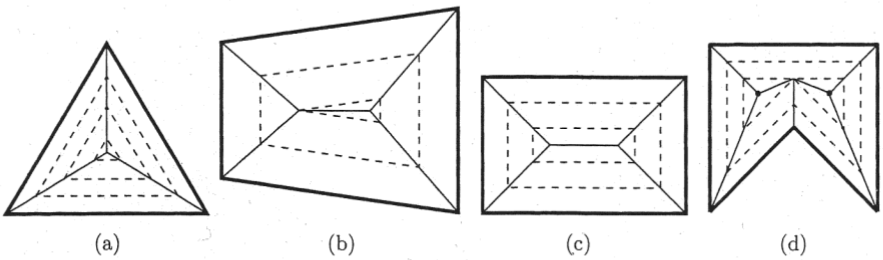


図３　グラフ面を縮小して直線骨格を構成する例

直線骨格はグラフ辺を一直線上に重ね合わせるのに有用であるが、そのままでは平坦折り可能な展開図とはならない。なぜなら、先に述べた平坦折り可能にする局所条件①②を満たすように仕向ける仕組みが全くないからである。そこで、平坦に折るために全ての骨格頂点から元の多角形のグラフ辺に向かって垂直に降ろす折り線を追加する必要が生じる。なぜ垂線かというと、直線骨格によって平面上に重ね合わせたグラフ辺にずれを生じさせないように平坦にするためにはその平面に対して垂直面で折らなくていけない為である。ここまでくると木構造と同じ状態になり、これらの垂線を鉛直線と呼ぶ。基本的に鉛直線はその骨格面の中に含まれている唯一のグラフ辺に対して垂直に引けば良い。鉛直線が骨格辺にぶつかった時は、その骨格辺を通過したところで鉛直線を反射させて新しい鉛直線の始点とする。この反射プロセスは鉛直線が骨格頂点にぶつかるか、無限遠に行くまで繰り返される。しかし、こうした鉛直線を実際には全て折る必要はなく、どの鉛直線を折り線として残すか選択するプロセスが必要となる。

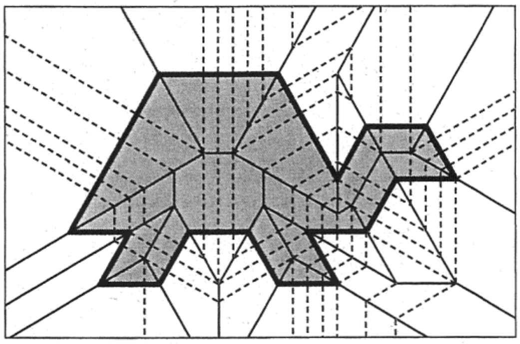

図４　直線骨格（細い実線）に対するすべての鉛直線（点線）の例

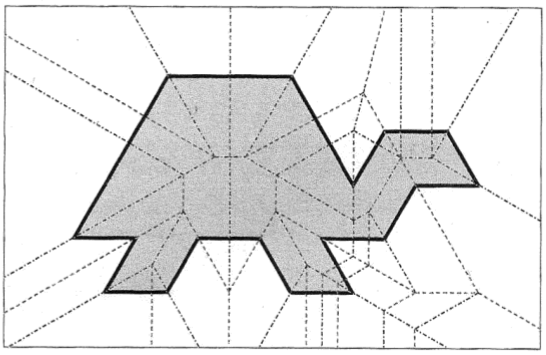

図５　図４に基づく実際の展開図

## ３−３　ディスクパッキング法

ディスクパッキング法は非常に複雑で、手作業でこのアルゴリズムを再現するのは無理そうなのだが、１つの多角形に対し主に次の６ステップの手続きから構成されている。円を敷き詰める手法は木構造で学んだ内容に似ていると感じた。

ステップ１　微小な値ｅだけグラフを平行にずらし、辺を幅２ｅのリボン状に太らせる。これをオフセット辺、太らせた全体をオフセットグラフと呼ぶ。

ステップ２　オフセットグラフにおいて
(a) 全ての頂点はディスク（円）の中心点
(b) 全ての辺はディスク（円）の半径の和集合
(c) ディスクのすき間は全て３本か４本の縁を持つ
という条件を満たすディスクパッキングを見つける

ステップ３　ディスクパッキングの円の半径を使って、オフセットグラフの面を３角形と「良い」４角形に分解する。　これらを元の多角形の内部か外部にあるかのラベル付けをする。

ステップ４　それぞれの３角形と４角形ごとに、それらの辺が共通の線上に並ぶように折って「分子」を作る。

ステップ５　個別の折り状態を繋ぎ合わせて、全体的な平坦折りを構成する。このとき外部分子の全ての境界辺は共通線上に並び、内部分子の全ての境界辺はその２ｅ分だけ上に並び、しかも全ての分子は線の上側に来るようにする。

ステップ６　それぞれの外部分子を、高さがｅよりも低くなるまで繰り返し沈め折りする。

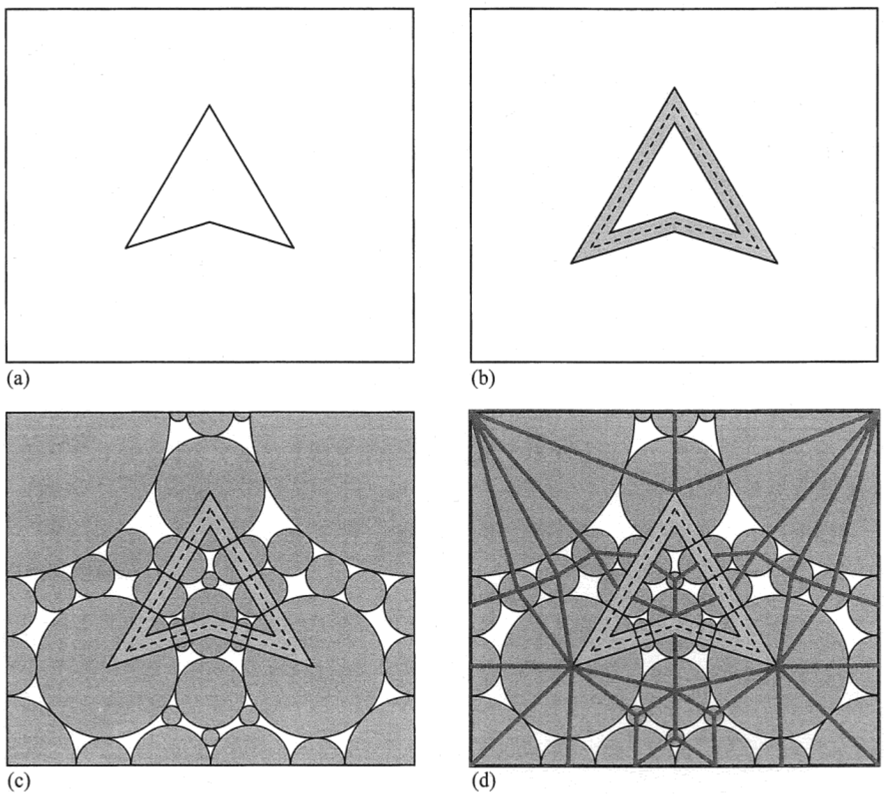


図６　ステップ１から３の様子

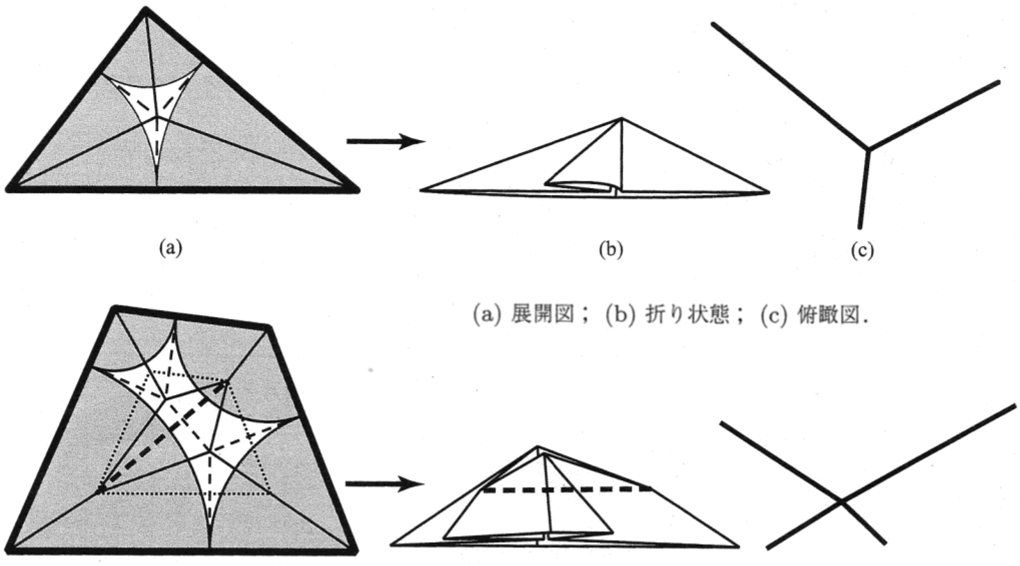


図７　ステップ４の３角分子と４角分子

一刀切り問題の数学的な証明という点では、これらのアルゴリズムがどのような図形に対しても万能に通用するのかもしれないが、実際には紙には厚みがあり、際限なく折り重ねることはできない。一般的なコピー用紙を４２回折ると月に届く高さになる、と聞いたことがある。僕は実現可能な折り数という制限を持って、オリジナルの設計に挑戦しなくてはならない。

## ４　CAD について

### ４−１　CAD の使用について

前回の労作展では方眼用紙に定規などを用いて作図をおこなったが、今年はより精密性を追求し、作図ソフトすなわち CAD を用いて、作図を行なった。初心者にも使いやすく、広く利用されているものとして「AutoCAD」という製品を選んだ。（AutoCAD を開発している AutoDesk 社では無料の学生版を提供していて、学生証のコピーを提出すれば最長３年間使用することが出来る。）

### ４−２　AutoCAD での作図について

AutoCAD でまずは、目的の多角形を作図した。CAD で精密に作図できるため、「K」「E」「I」「O」のそれぞれの文字の縦横比は、黄金比に近似した比とした。
折り線の作図については、さまざまな手法があるが、今回は折り線上の 2 点の座標を入力し、その２点を結ぶ線を折り線とした。

## ５　「KE」の設計

「KE」の設計は昨年の「K」「E」単体の設計図をまず取り込むところから始めた。Auto CAD に書き込んだ「K」「E」の中に色をつけて座標の変化量を打ち込んで「K」と「E」に設計した折り線を書き込んでいった。角の２等分線の入力の仕方を調べたが、線の長さを入力しなければならず、斜線の長さは三平方の定理により殆どの場合が無理数｛$\sqrt{a^2+b^2}$（a,b は任意の自然数）｝となってしまうので、座標の変化量を入力して行うことにした。山折り線は赤で、谷折り線は青で表した。二つの図形の間隔を適度にとり、図形内側から外側に伸びていた折れ線の端を他の線と交差するまで延長させた。だが延長させても既存の線とは交差しない線があったので、そこは放置しておき、後で手を使って実際に折り目を増やして結果を見てみようと考えた。そして、この仮の設計図を A4 用紙に印刷して折ってみることにした。僕の最初の設計図では、K を一旦半分に折ってから畳んでいたが、そうしてしまうと蛇腹折りのように畳む E と折り目が噛み合わなくなってしまった。そこで、K を最初に半分に折るのをやめ、E と同じように蛇腹折りのような折り方をすることで折り目が食い違う問題は解消された。次に E は図８の矢印の部分の枝を後ろに折り返すのではなく前に折り返す事で K の畳む方向と同じになる。この段階で K と E はそれぞれ一つの線になり、あとは二つを重ねれば完成する。今回は、図 11 のように二つの線は上からみると一直線上にあり、上下がずれているだけだったので、繋がりを持つ線を折る事で完成となった。最後の線は折り線を調べたところ、K の斜線と E の縦線を延長して出来た交点から引かれた延長線の二等分線だったと考えられる。なぜなら、角の二等分線には二つの線を同一線上に折り重ねることが出来るという性質があり、K と E は一つの線にす

でになっていたため、それぞれの重ねられた線の元となっている線同士の２等分線を折ることで完全な直線になるからだ。

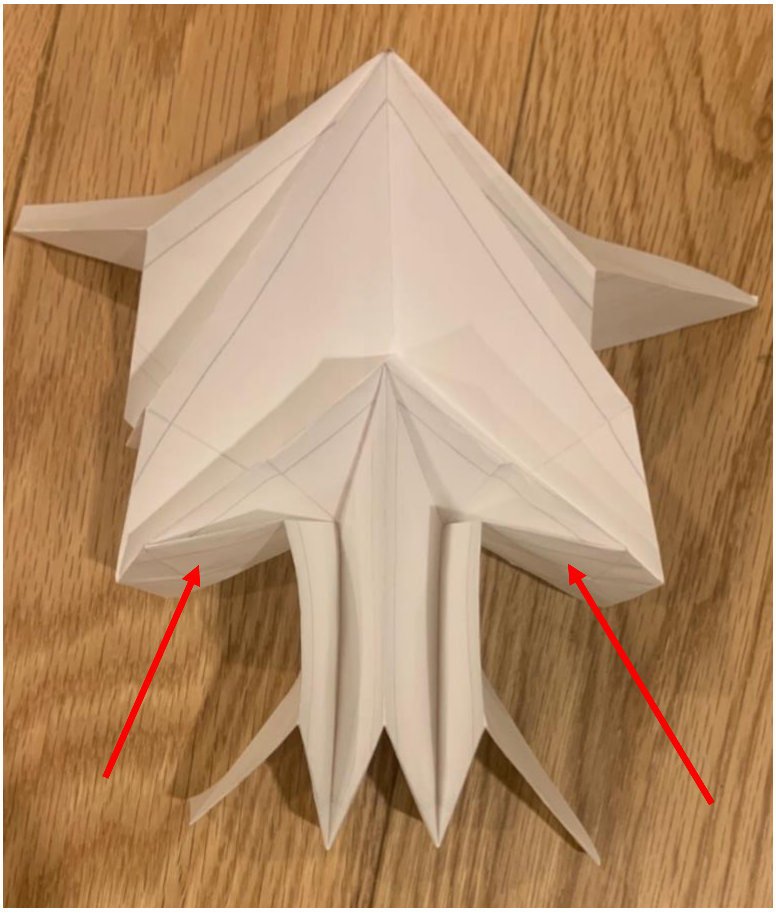
図８　「KE」を折り畳んで上から撮った図

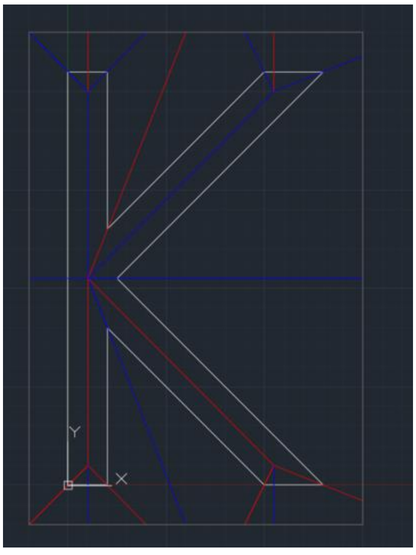
図９　「K」単体の設計図

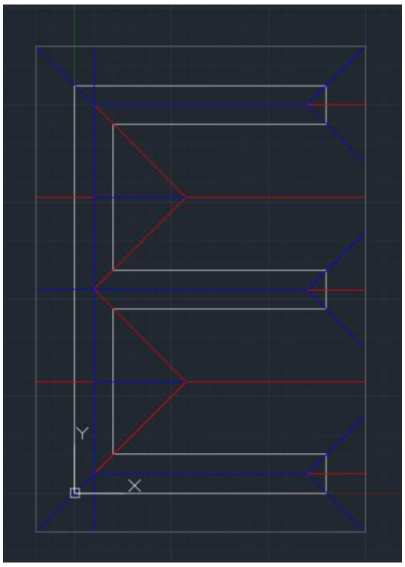

図 10　「E」単体の設計図

図 11　「K」と「E」の上下をずらした形

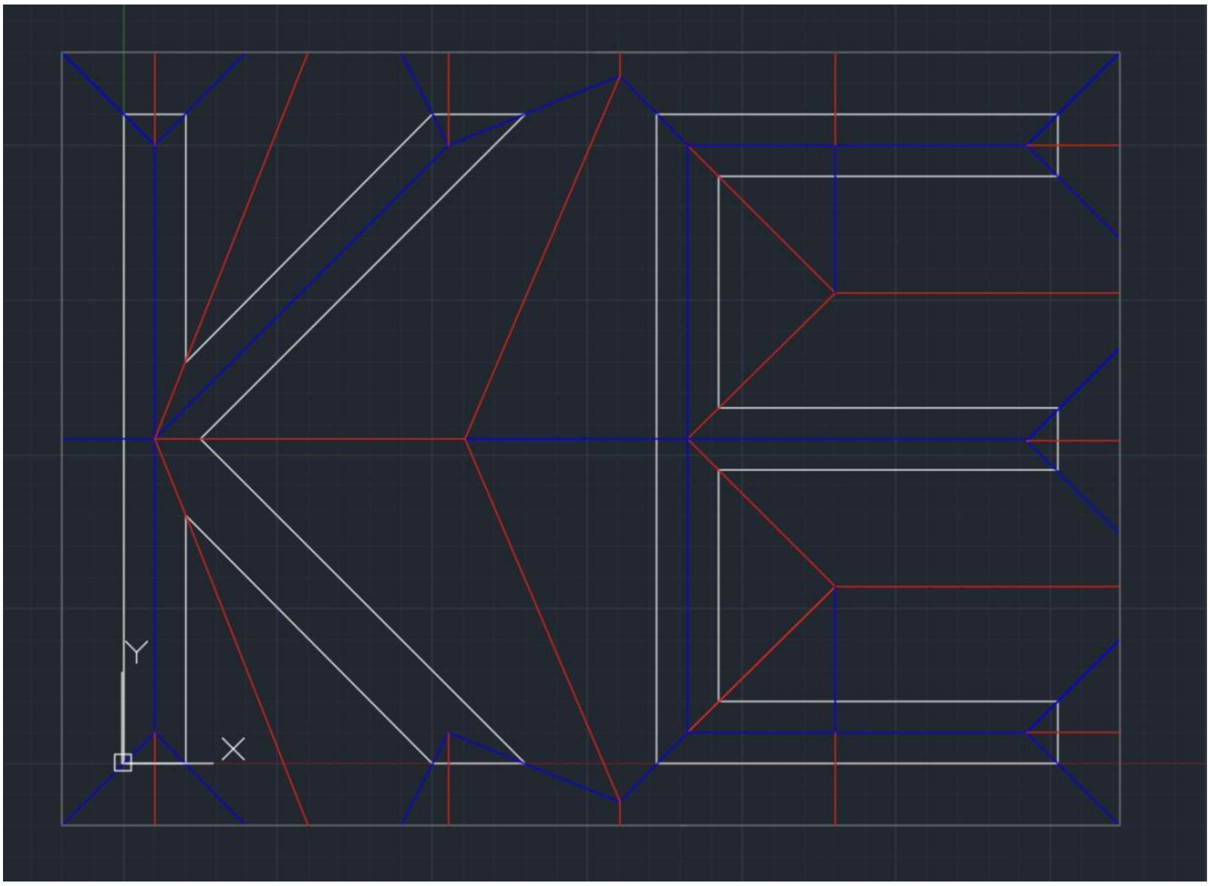


図 12　　完成した「KE」の設計図

## ６　「IO」の設計

「IO」の設計は「KE」の時と同じく、昨年の「I」「O」単体の設計図をまず組み込むところから始めた。その後も同様に色を変えて、折り線を入れていった。「I」「O」の間には直感で「K」「E」以上に間を開けておいた。今回は「K」と「E」にて試した折り線の入れ方、飛び出ている折り線を他の線とぶつかるまで延長する、が正しかったので延長できるだけ延長すると、図 17 のようになった。これを印刷して折ると、「I」「O」の折り線が唯一ぶつかる場所に紙が余って、盛り上がる部分ができてしまったので、図 13 の赤線が示すように内側に折り返すことで解決した。この時点で、「I」と「O」は一直線になっており、今回は図 16 のように二つの線の真ん中辺りを折れば一直線になる。「I」と「O」は比較的折るのは簡単だったが、折り線を CAD で入力するのがとても大変だった。最後に折った二等分線は平面グラフにすると、他の折れ線と当たる度に 90 度屈折していた。そして、全ての線は「I」と「O」の縦線と横線のどちらかに平行であった。角の二等分線はほとんどの場合が平行ではないはずなのに、なぜ全ての線がいずれかのグラフ辺と平行であったのかは「I」と「O」の両方が折り重ねていた線が平行だったからだ。二等分線はかならず中点を通る。今までは交点などを基点に二等分線を作図していたが、今回は二点の中間地点を指定しなければならない。Auto CAD による中点の作図には任意の

二点を指定して、その長さを直径とする円を作図して円の中心から線を引かなければいけない。盛り上がった部分を内側に折り返したと前述したが、その折り線も定規で測ったところ、図 17 の緑の線の中点などを通っており、作図は大変だった。

図 13　　「IO」を折っている途中の形

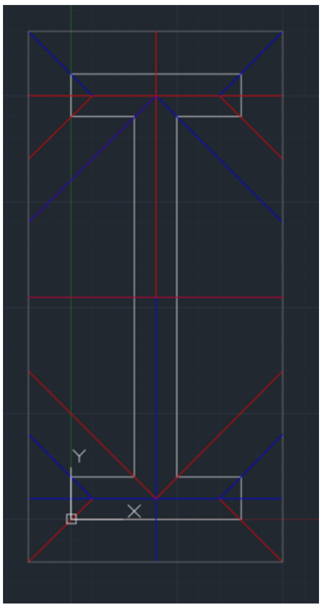


図 14　「I」単体の設計図

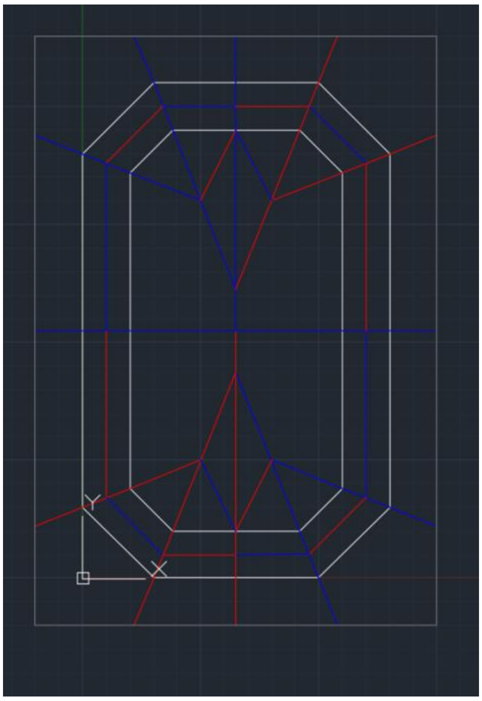

図 15　「O」単体の設計図

図 16　　「IO」の完成一歩手前

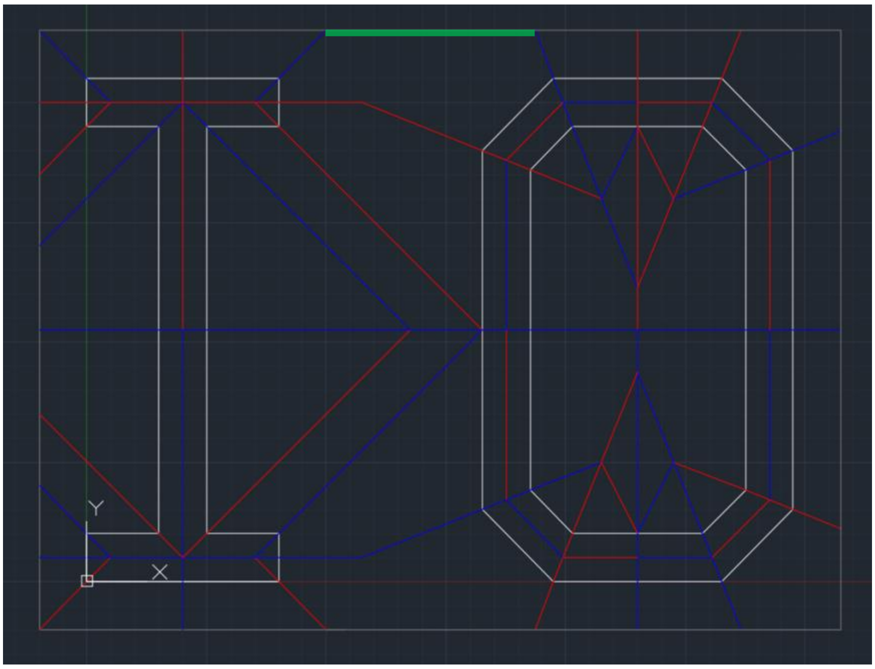


図 17　「IO」の初期の図

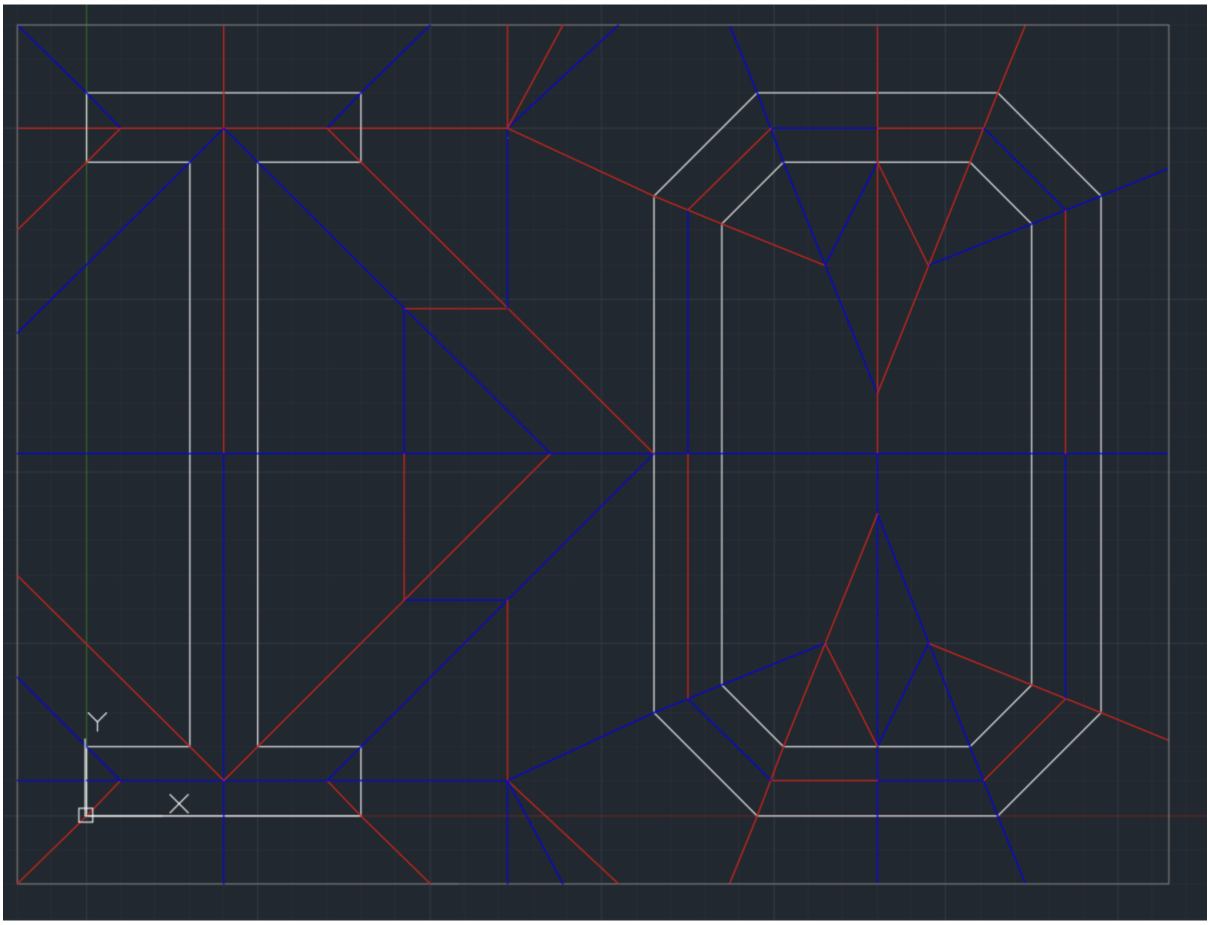


図 18　完成した「IO」の設計図

## ７　直線骨格法との結びつき

「KE」の設計図は後から見ると、全ての折り線がいずれかの角の二等分線になっており、これは直線骨格法に基づいていると言える。一方、「IO」の場合は設計図を見ただけでは線の意味がわからないものも多い。この違いが生まれる訳は何かと思い、「KE」と「IO」の折れ線の引き方を比べてみた。「K」と「E」は共に Erik のアルゴリズムに基づいて設計していた。Erik のアルゴリズムというのは直線骨格法のことで、先に述べたようにグラフ辺を内側に縮小していき、点になったら縮小してきた軌跡を結ぶというやり方である。「I」も同じく Erik の方法に則って作図していたが、「O」は急いで取り掛かった為うっかり昨年自分で考えた方法を使用していた。そのせいで、直線骨格法が当てはまらなくなり、二等分線以外の線が生まれてしまったと考えられる。よって、「IO」も「O」を直線骨格法に基づいて作図すれば、全ての折り線が二等分線となる一番美しい設計図が出来上がるのではないかと考える。

## ８　「EI」の設計

「KEIO」の設計の前段階として「KE」と「IO」をつなげるべく「EI」の設計をしてみる。
「EI」の設計は途中までは「KE」と「IO」同様、「E」と「I」の設計図をつなげて、間隔をあけて印刷してみた。まずは「E」と「I」を折り線通りに折ってみた。しかし、図 19 の黄色い部分が噛み合わず、混乱した。試行錯誤を繰り返し、数日考え続けた結果、折ることが出来た。図 19 の黄色い部分は図 16 のように同じ折り方をすることで解決した。「IO」は作図が大変だったが、「EI」は折ることがとても大変だった。Auto CAD に入力する際は、折り線が正しく交差するように適当に開けた間隔を調整した。それによって設計図に記されている折り線は全て、いずれかの角の二等分線になった。よって、「EI」の設計図は直線骨格法に基づいていると言える。理由としては「E」と「I」が Erik のアルゴリズムと同一の方法で製図したからだと考えられる。

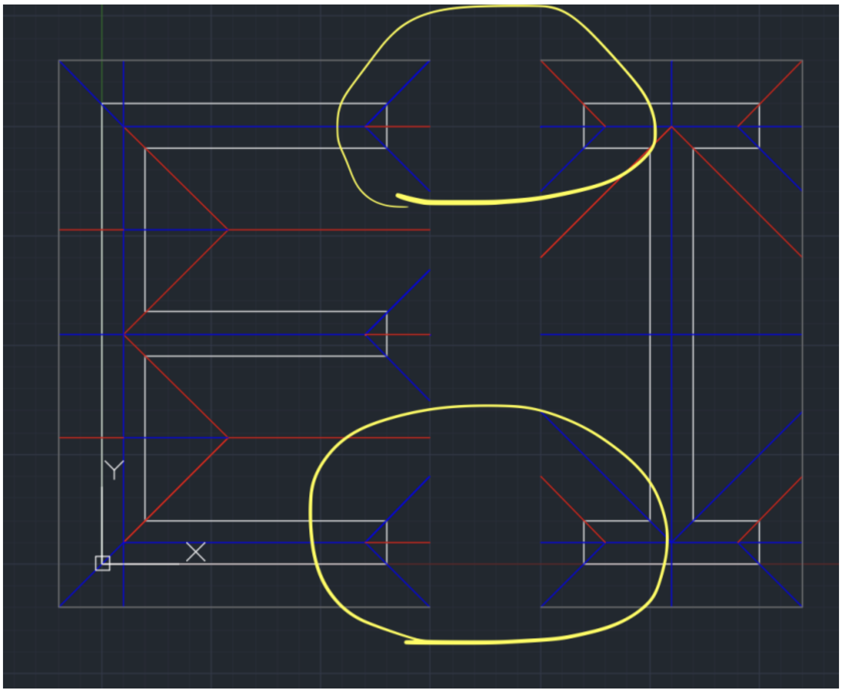


図 19　「IO」の初期の図

図 20　「EI」の突起を一直線にしている形

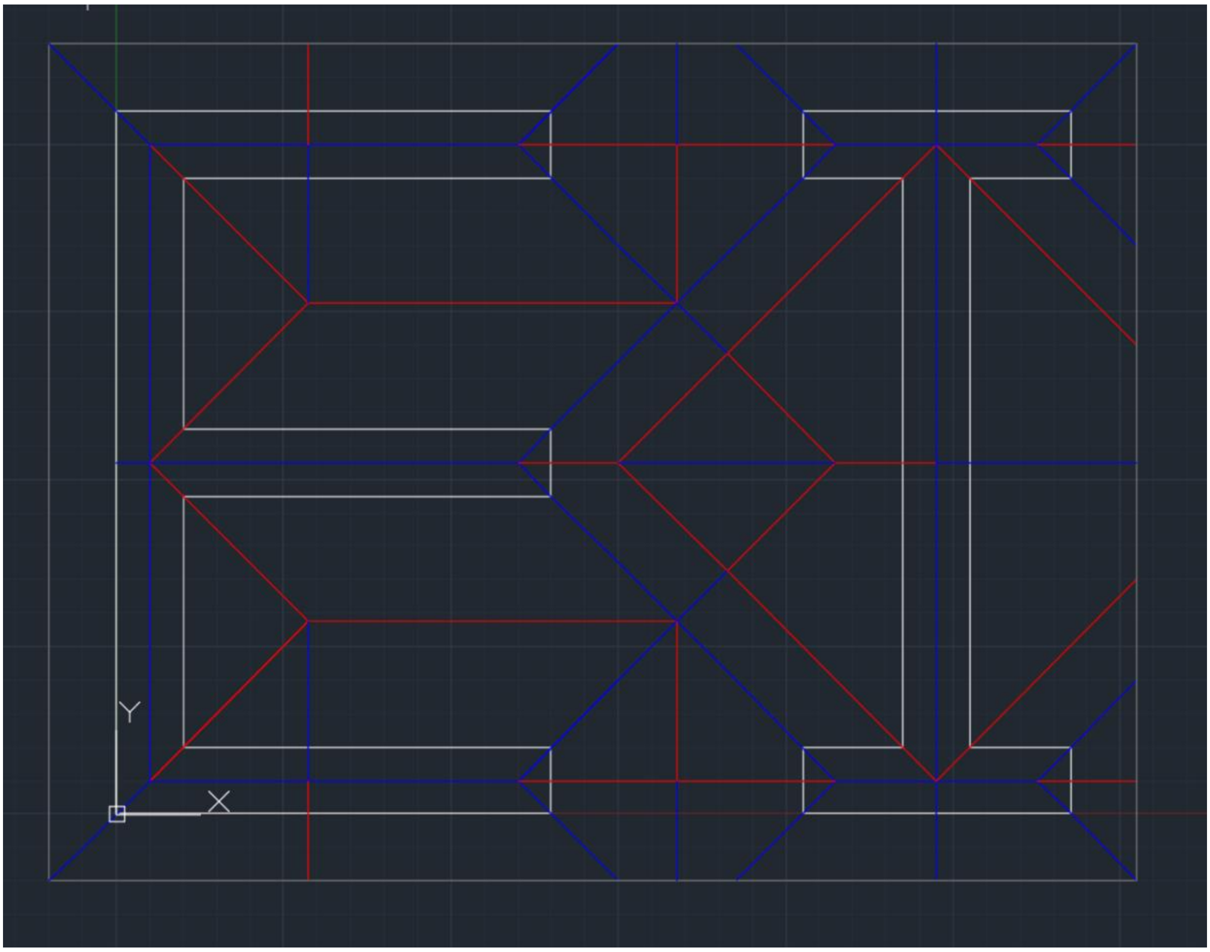


図 21　 完成した「EI」の設計図

## ９　「KEIO」の設計

今までに「KE」、「IO」、「EI」の設計図を作ってきた。これらが、繋がれば「KEIO」は完成する。折り線が食い違ってしまうのではないかと心配していたが、大丈夫だった。折り線から、全部繋がると判断できたので、連結して「KEIO」の設計図が完成した。「KEI」は直線骨格法で作ったのに、「O」だけ違うのは気持ちが悪い。そこで、「O」をアルゴリズムに則った設計にして再度「IO」を折ってみることにした。前に「IO」の折り線は折ってみるまで線の意味がわからないものがあると述べたが、「O」を直線骨格法で設計してもあまり変化は無かった。（図 22 は変更前の設計図、図 23 が変更後の設計図）そして直線骨格法で作った「IO」の設計図を「KEIO」の設計図に組み込んで終わりだと思われた。実際に印刷して折ってみるとちゃんと一直線にはなったものの、山谷の割り当ての間違いが多数あり、そこを修正して完成となった。
完成した「KEIO」を分割して A4 用紙 2 枚に印刷して横長になるようにくっつけたものを折ってみると見事に一直線になった。しかし、図 25 のように O だけは最後に半分に折る形でしか表せなかった。来年こそは「KEIO」が全て蛇腹折りのように綺麗に畳むことが出来るよう頑張りたい。

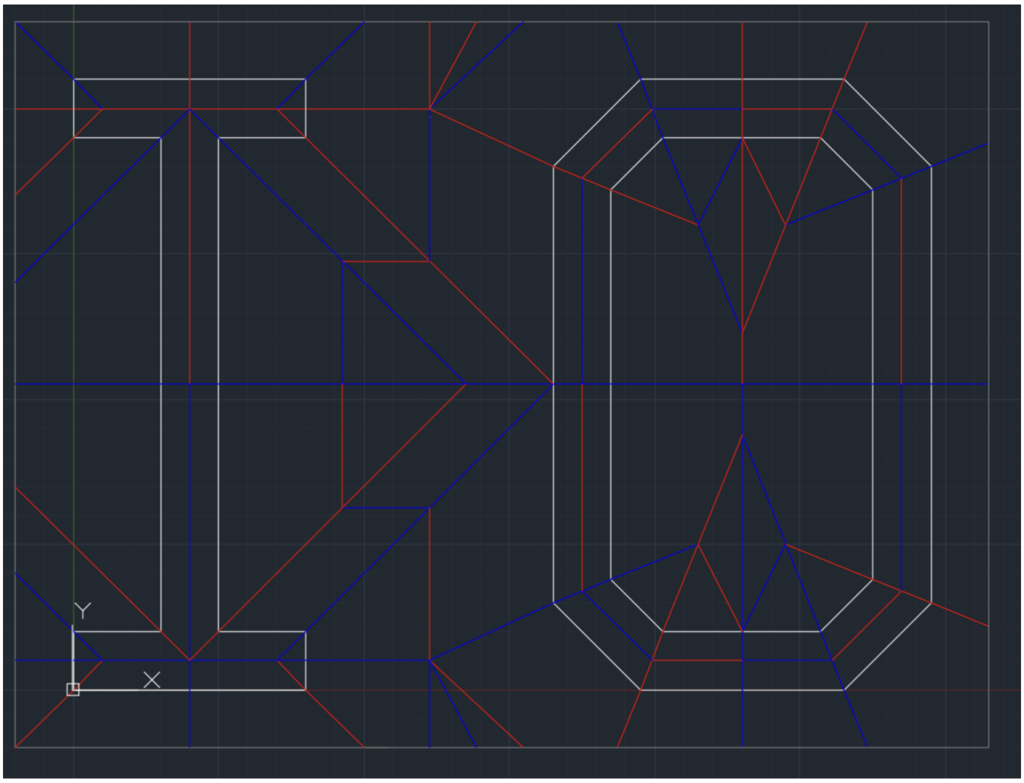

図 22　自分のオリジナルの「IO」設計図

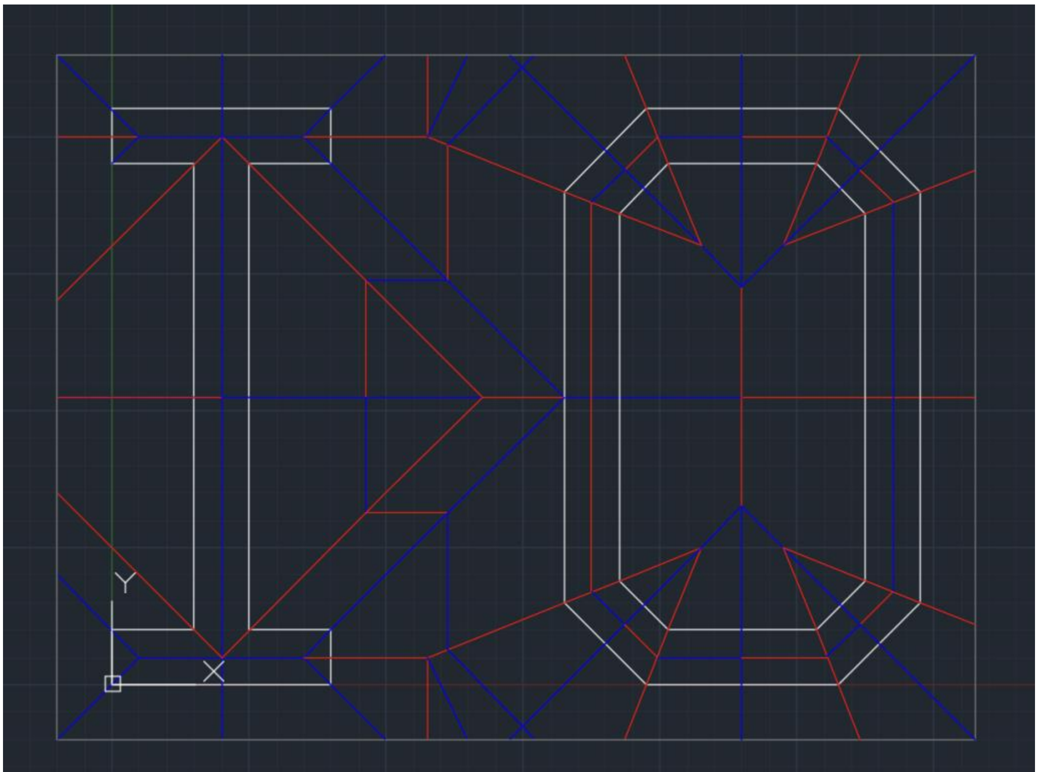

図 23　直線骨格法による「IO」設計図

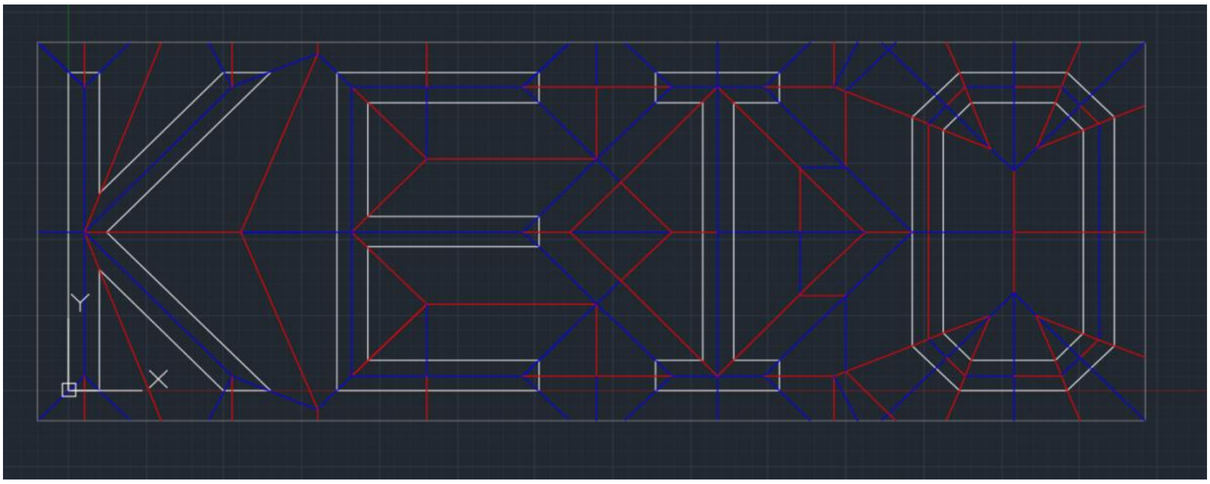

図 24　 完成した「KEIO」の設計図

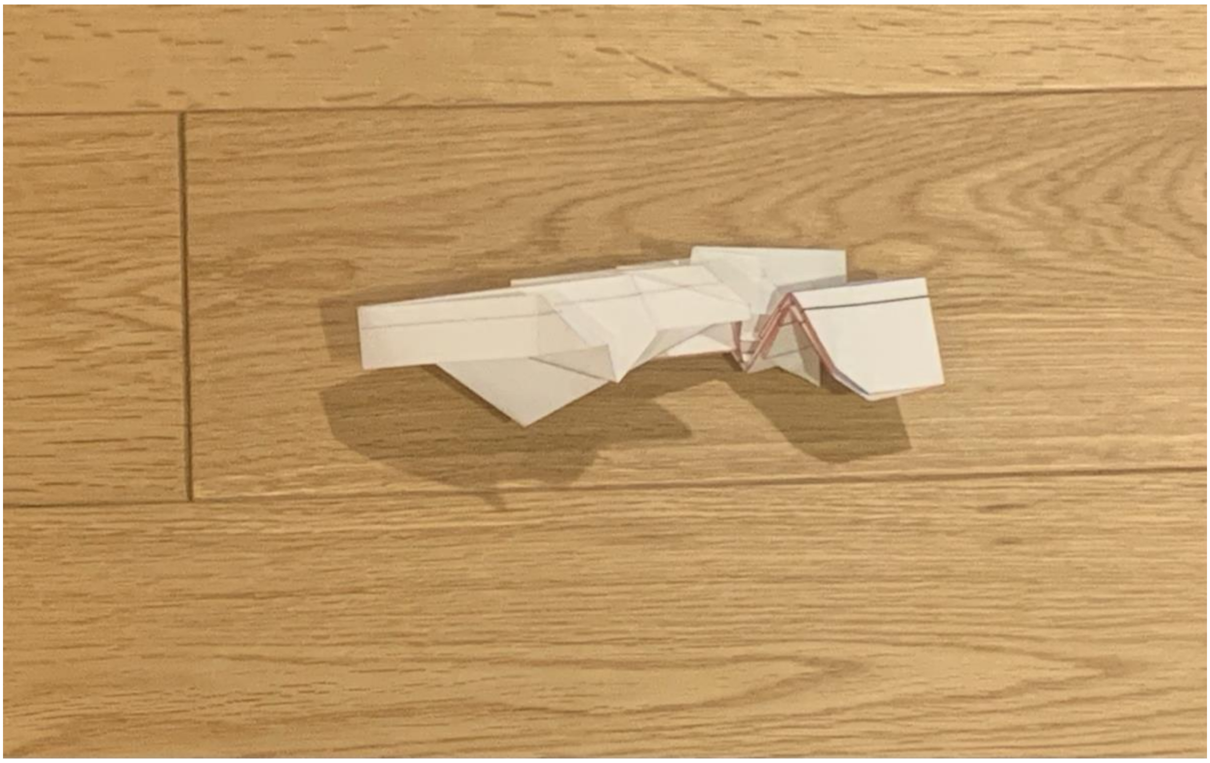

図 25　一直線に折られた「KEIO」

## 10 まとめ

今年は４文字を一気に切り出すことを最終目標に作業をした。同時に折り線の規則性を見出し、グラフ辺を見るだけで、折り線を設計できるようになるのも目標だった。しかし、直線骨格法とディスクパッキング法という二つのアルゴリズムの手法を昨年より深く学んだものの、その理論から生じる非常に多数の作図作業を手作業で追うのは AutoCAD を使っても大変困難で、結局のところ実際に手で折りながら設計を考え、それを作図に投影した後に出来た折り線の数理を考察するという流れになってしまった。ただ、手で折っていく上で、グラフ辺を重ねていく手法を先に理解できていることは作業をスムーズにしてくれた。今年の反省としては、夏休み中の予定の関係で作業時間が限られており、手元にあるものからと焦って作業を進めた結果、後から昨年の細かい振り返りや難しかった理論の理解をして先行作業が無駄になった部分が多々あったことがあげられる。やはり研究には計画性が大事だ。また、AutoCAD はもちろん方眼紙で作図した昨年よりずっと正確な作図を可能にしてくれたが、直接に角の２等分線を入力させることが出来ず、逐一座標の変化量を求めて入力しなければいけなかったことも困難の一因だった。
この研究を通して、人が直感に従い、手を動かしながら解に辿り着くプロセスの優れている点も再認識出来たし、アルゴリズム化することには万能で一般化されていなくてはならないがために複雑さと周りくどさが生じてしまうものなのだなとわかった。その部分を解決するには、来年はプログラミングを学び、いよいよ作業の自動化に取り組まなくてはいけないかもしれないと思っている。

最後に、昨年の研究への講評の中で今年の研究のヒントを下さった藤尾先生、手作業を手伝ってくれた家族、僕の研究を読んでくださった皆様、ありがとうございました。

**11 参考文献**


Erik D. Demaine and Joseph O’Rourke : Geometric Folding Algorithms, Linkages, Origami, Polyhedra.　Cambridge University Press, July, 2007

エリック・D・ドメイン、ジョセフ・オルーク：幾何的な折りアルゴリズムーリンケージ、折り紙、多面体　　近代科学社、2009 年

朝日新聞 GLOBE 数学という力　2010/2/1 号　（注１）

松川剛久、三谷純：平坦折り紙の数理　　日本応用数理学会論文誌 Vol.27, No.4 2017
https://www.jstage.jst.go.jp/article/jsiamt/27/4/27_333/_pdf
参照日 2022/8/30

三谷純：折り紙研究ノート
https://mitani.cs.tsukuba.ac.jp/origami/main.html
参照日 2022/8/30

坪井俊：講義ノート「折り紙と数学」
https://tsuboiweb.matrix.jp/files/musashino/4w-lecture/2020week4exp.pdf
参照日 2022/8/30

Robert Lang (吉田陽介　訳) : 5.0 The Tree Method of Origami Design
https://langorigami.com/wp-content/uploads/2015/09/TreeMkr40jp_s.pdf
参照日 2022/8/30

鈴木孝子：はじめて学ぶ AutoCAD 2023 作図・操作ガイド　　株式会社ソーテック社　2022 年