

エッシャー風タイリング問題に対する局所探索法

今堀 慎治^{1,a)} 酒井 翔平¹

概要: 1種類の図形によって平面を隙間なく、かつ、重なりなく敷き詰めたものをタイリングと呼ぶ。エッシャー風タイリング問題とは、入力図形が与えられ、その図形にできるだけ近い図形によるタイリングを求める問題である。本研究では、この問題に対する小泉と杉原の解法 (H. Koizumi and K. Sugihara: Maximum Eigenvalue Problem for Escherization, Graphs and Combinatorics, Vol. 27, 2011, 431-439) について考察を行い、局所探索にもとづく改善手法を提案する。また、提案手法の性能を数値実験により検証する。

1. はじめに

タイリングとは、様々な図形によって、平面を隙間も重なりもなく敷き詰めたものである。タイリングは古くから建築物の装飾などに多く用いられており、現在においてもさまざまな場面で広く使われている。タイリングによって作られる模様は単純な幾何学図形から、複雑な図形に至るまで多岐に及ぶ。タイリングには芸術的な側面の他に、数学的な側面も多くあり、繰り返しの中に現れる法則や性質、タイリングの種類等について深く考察されている [3], [5]。オランダの版画家 M. C. Escher (エッシャー) は数学的な見地からタイリングを研究し、タイリングに関する芸術的な作品を数多く残した [2]。エッシャーは動物などの形を用いた芸術的なタイリングを作成しており、1種類の図形を用いた作品や、2種類以上の複数種類の図形を用いた作品を残している。

エッシャーの作品のようなタイリングを実際に作ろうとすると、芸術的なセンスや膨大な時間が必要となり非常に難しいことがわかる。そこで、誰でも簡単にタイリングを作ることができるよう、計算機を使ってタイリングを生成する問題が考えられる。この問題はエッシャーの名前と作品にちなんで Escherization Problem (エッシャー風タイリング問題) [6] と呼ばれる。

エッシャー風タイリング問題

ある入力図形 S が与えられたとき、

- (1) 図形 T は平面を敷き詰めることができる、
 - (2) 図形 T はできるだけ S に近い形である、
- という2つの条件を満たす図形 T を求める (問題の詳細は2章および3章で説明する)。

この問題に対し、Kaplan と Salesin はアニーリング法を用いた発見的解法を提案した [6]。彼らの解法は、入力図形が凸多角形や凸に近い形状の場合は精度のよい解を得られるが、複雑で入り組んだ入力図形に対しては効果的に機能しないという問題点があった。また、解 (タイリング) を求めるために多くの計算時間を要するという課題もあった。そこで小泉と杉原は、Kaplan らとは異なるアプローチでエッシャー風タイリング問題に取り組んだ [10]。彼らは、入力図形を多角形で近似し、図形同士の近さを定める基準としてプロクラステス距離を導入することで、エッシャー風タイリング問題を線形制約のもとでの2次関数の最小化問題として定式化を行い、その最適化問題の解を解析的に求めることに成功した。小泉と杉原の手法は、複雑な入力図形に対しても、しばしば質の良いタイルを得ることができ、さらに Kaplan らの手法よりも短時間で動作するという特徴をもつ。しかし、この解法には入力図形を多角形で近似する際に、点の配置をどのようにするかによって得られるタイルの質が大きく変わってしまうなど、いくつかの課題が残った。

そこで本論文では、小泉らの手法の性質と問題点について再考察を行い、各問題点に対して改善するアイデアを提案する。提案手法の特徴として、数多くのタイルを効率的に生成するために、局所探索法の枠組みを用いる点がある。提案手法を実装し、計算実験による性能評価を行った結果を報告する。

2. タイリング問題

この章では、まずタイリングの基礎的事項についてまとめる。続いて、本研究で対象とするエッシャー風タイリング問題について述べる。

¹ 名古屋大学, Nagoya University

^{a)} imahori@na.cse.nagoya-u.ac.jp

2.1 タイリングの基礎事項

平面を有限種類の図形で隙間も重なりもなく埋め尽くしたパターンをタイリングといい、タイリングに使われる図形をタイルという。2つのタイルが交わる時、交わりは曲線となり、その曲線をタイリング辺 (tiling edge) という。また、3つ以上のタイルが交わる時、交わりは点となり、その点をタイリング頂点 (tiling vertex) という。1つのタイルに対し、タイリング頂点をつないでできる多角形をタイリング多角形 (tiling polygon) と呼ぶ。なお、タイルの形状が多角形の場合は、タイリング辺やタイリング頂点と区別するために、多角形としての辺を図形の辺、多角形としての頂点を図形の点と呼ぶ。

1種類のタイルからなるタイリングを考える。タイリング上のタイルを1つ選ぶ。同じタイリング上のあるタイルに対し、元のタイルと重なるような合同変換のことを対称 (symmetry) という。タイリング上の任意の2つのタイルに関して対称が存在し、もしその変換を行ったときにタイリング全体も重なるのであれば、そのときその2つのタイルは同値 (equivalent) であるという。この同値関係によってタイリングのタイルは同値類に分類される。つまり、2つのタイルが同値であるとき、その2つのタイルは同じ同値類に分類されることとなる。タイリングが1つの同値類しか持たないとき、isohedral タイリングと呼ばれる。

isohedral タイリングは、数学的に扱い易く、一方で様々な形を表現するだけの柔軟性も持っている。実際に、1種類のタイルから成るエッシャーの作品のほとんどが、isohedral タイリングで出来ていて、さらにエッシャーの作成した2種類の図形によるタイリングでは、異なる2つのタイルを一つと見なすと isohedral タイリングとなることが知られている [6], [7]。このような状況から、本研究では、isohedral タイリングのみを考えることとする。

isohedral タイリングの性質は、topological type と incidence symbol によって決定される。topological type とは isohedral タイリングの中の一つのタイルを取って、そのタイリング頂点において交わっているタイルの個数 (タイリング頂点の次数) を並べたリストのことを言い、isohedral タイリングの定義より、タイリングの中の全てのタイルでそのリストは同じである。topological type は、isohedral タイリングの場合、全部で11種類であることが知られている [3]。また、incidence symbol はタイルの隣接関係を制限するものであり、tile symbol と adjacency symbol を合わせたものである。incidence symbol は次のようにして得られる。

まず初めに、タイリングの中の任意のタイルを選んで、そのタイルの1つのタイリング辺に向き付きのラベル a を付ける。そして、今決めた向きにタイリング辺をたどり、ほかのタイリング辺にも同様に b, c, \dots とラベルを付ける (図1参照)。ただしこのとき、タイリングが、あるタイル

をそのタイル自身に移すような対称を持つ場合は、移る前と移った後のタイリング辺には、向きを含めて同じラベルを付ける。このようにして付けたラベルを並べたものが tile symbol である。図1 (左) は $[a^+b^+c^+d^+e^+f^+]$ 、図1 (右) は $[ab^+c^+dc^-b^-]$ となる。ラベルの右肩に付いている符号は、初めに決めた向きとラベルが同じ向きなら $+$ 、逆向きなら $-$ として決められた符号である。ただし、ラベルが両方の向きを持つ場合は、何も付けないこととする。

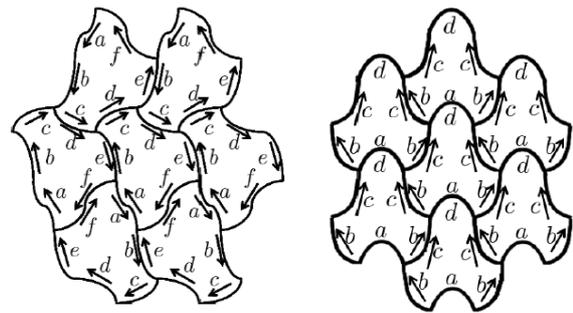


図1 incidence symbol (左: 標準, 右: 鏡映を含む)

次に、得られたラベルを隣接する他のタイルにも付ける。そして、元のタイルのラベルを並べた順番に、元のタイルのラベルと隣接するラベルを並べたものが adjacency symbol である。ただし、対称性のある部分は省略することとする。図1 (左) の場合は $[a^+e^+d^-c^-b^+f^+]$ となり、図1 (右) は $[dc^-b^-a]$ となる。ここで、ラベルの右肩に付いている符号は、タイリング辺の両側にあるラベルが違う向きなら $+$ 、同じ向きなら $-$ として決められた符号である。tile symbol と adjacency symbol を並べたものを incidence symbol と呼ぶ。図1 (左) の場合は $[a^+b^+c^+d^+e^+f^+; a^+e^+d^-c^-b^+f^+]$ となり、図1 (右) の場合は $[ab^+c^+dc^-b^-; dc^-b^-a]$ となる。topological type と incidence symbol によって、isohedral タイリングは、93種類に分類されることが知られており [3]、それぞれ IH01, IH02, ..., IH93 と書いて区別される。

次に、タイルを変形することを考える。タイルの変形とは、タイリング頂点の位置を変えることと、タイリング辺を変形することであり、許される変形は incidence symbol によって制限される。まず、タイリング辺の変形について考える。タイリング辺の変形に対する制限は、incidence symbol から得られ、次の4種類に分けられる:

S型 タイリング辺の両側のラベルが同じ名前でも向きが異なるとき。この場合は、そのタイリング辺の中点に関して点対称でなければならない。その条件を満たす範囲で変形できる。

U型 タイリング辺の両側の名前が異なり、少なくとも一方が向きを持たないとき。この場合は、そのタイリング辺の垂直二等分線に関して線対称でなければならぬ

い. その条件を満たす範囲で変形できる.

J型 両側のラベルの名前が異なり, どちらも向きを持つとき. この場合は, そのタイリング辺は自由に変形できる.

I型 両側のラベルの名前が等しく, どちらも向きを持たないか, どちらも同じ向きを持つとき. この場合は, S型とU型の両方の条件を満たさないといけないため, 変形できない. すなわち, このタイリング辺は直線分でないといけない.

同じラベルが付いたタイリング辺は, 全て同じ形になるということにも注意する.

タイリング頂点に関する制限も同様に, incidence symbolによって得られる. タイリング頂点を動かすことは, タイリング多角形の形を変えることを意味するが, incidence symbolによって, タイリング多角形の辺の長さや, 内角が制限される. 詳細は文献 [5], [6] において考察されている.

isohedral タイリングには 93 種類のタイプが存在するが, 対称性の高いものは変形の自由度が少なく, エッシャー風タイリングを生成する際に, タイルを変形するという観点から見れば, その実行可能解がほかのタイプの実行可能解にすべて含まれてしまう場合がある. 例えば, 図 2 の場合は, IH10 は 1 種類の J 型の辺しか存在しないのに対して, IH07 の方は 3 種類の J 型の辺が存在している. つまり, IH10 でタイリング可能なタイルは, 必ず IH07 でもタイリング可能であると言える. また, 図 3 の場合は, IH38 で

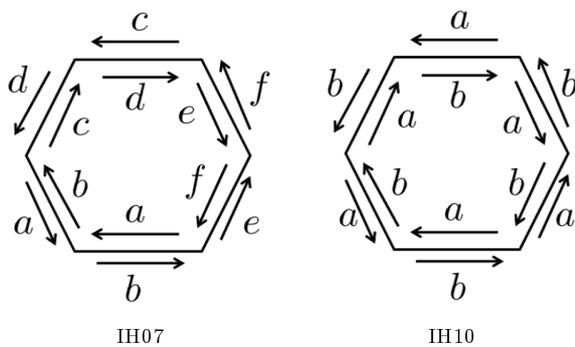


図 2 IH07 と IH10 の関係

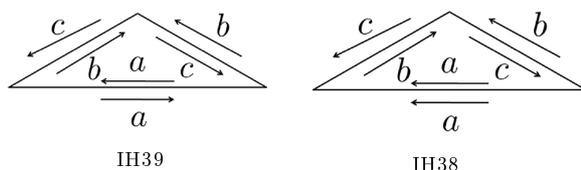


図 3 IH39 と IH38 の関係

I型となっている辺が, IH39 では S 型となっている. I型は変形できないのに対して, S型は中点に関して点対称であるという条件を満たす範囲で変形できる型であるため, IH38 でタイリング可能なタイルは, IH39 でもタイリング

可能である. したがって, エッシャー風タイリングを生成する際には, 対称性の高いものについては省いて考えることができる. IH_i の全ての実行可能解が IH_j の実行可能解に含まれるとき「 IH_i は IH_j に含まれる」と呼ぶ. いずれかの辺が変形可能であり, 他どのタイプにも含まれないものは 29 種類であることが知られているため [5], [9], 本論文では 29 種類のタイリングタイプを考慮する.

2.2 エッシャー風タイリング問題

エッシャー風タイリング問題とは, ある入力図形 S が与えられたとき,

- (1) 図形 T は平面を敷き詰めることができる,
 - (2) 図形 T はできるだけ S に近い形である,
- という 2 つの条件を満たす図形 T を見つける問題である. ここで, この問題を最適化問題として書き換える. まず, 1 つ目の条件は, 本稿では isohedral tiling のみを扱うので, 「図形 T は incidence symbol の定める制約に従う.」と書き換えることができる. 次に, 2 つ目の条件は, 二つの図形を引数に取り, 距離の公理をみたすような関数 $d(S, T)$ を用いて, 「図形同士の距離 $d(S, T)$ が出来るだけ小さい.」と書き換えることができる. よって, エッシャー風タイリング問題は以下のような最適化問題といえることができる.

$$\begin{aligned} & \text{最小化} && d(S, T), \\ & \text{制約条件} && \text{incidence symbol の定める制約.} \end{aligned}$$

次章では, この最適化問題における目的関数と制約条件について詳しく述べ, エッシャー風タイリング問題に対する既存解法を説明する.

3. エッシャー風タイリング問題の既存解法

本稿ではエッシャー風タイリング問題に対する, 小泉と杉原の定式化および解法 [9],[10] を紹介する. また, この手法の特徴についての考察を行う.

3.1 図形同士の距離

図形同士の距離 d は, 図形の表現方法や, 比較の仕方によって様々なものが考察されている. タイリングを扱う際は, 回転・伸縮・平行移動によって互いに重なる図形は同じ形であるとみなすため, 回転・伸縮・平行移動に関して不変な距離が望ましい. 小泉らはそのような特徴をもつ距離の一つである, プロクラステス距離 [8] を用いている.

プロクラステス距離を用いるために, まず図形を n 角形で近似し, その n 個の頂点の x, y 座標を順番に並べた $2 \times n$ 行列 U によって図形を表現する. すなわち n 角形の頂点の x 座標を $P_{1x}, P_{2x}, \dots, P_{nx}$, y 座標を $P_{1y}, P_{2y}, \dots, P_{ny}$ としたとき, その図形は,

$$U = \begin{pmatrix} P_{1x} & P_{2x} & \cdots & P_{nx} \\ P_{1y} & P_{2y} & \cdots & P_{ny} \end{pmatrix}$$

という行列で表現される。ただし、行列 U は、第一頂点 P_1 を n 角形のどの点とするかによって変化することに注意する。また、以下では、 n 角形の頂点を P_1, P_2, \dots, P_n としたとき、その位置ベクトルも同じ記号 P_i で表すこととし、図形とその図形から得られる行列も、同じ記号で表すこととする。

重心が原点に重なるように配置された図形 U, W に対して、プロクラステス距離 $d(U, W)$ は次式で定義される。

$$\begin{aligned} d^2(U, W) &= \min_{s, \mu} \left| sR(\mu) \frac{U}{|U|} - \frac{W}{|W|} \right|^2 \\ &= 1 - \frac{|UW^T|^2 + 2\det(UW^T)}{|U|^2|W|^2}. \end{aligned} \quad (1)$$

ここで、 $|X|$ は行列 X のフロベニウスノルム、 s はスカラー、 $R(\mu)$ は μ ラジアン回転行列を表す。この距離は定義から回転・伸縮・平行移動に関して不変である。

3.2 最適化問題の定式化

2.2 節の最適化問題を定式化する。ただし、ここでは式の細かい展開や定理の証明は省略する。詳しくは文献 [9] を参照のこと。

まず、目的関数について考える。重心が原点にある入力図形を W 、求める図形（タイル）を U とする。(1) 式より、この2つの図形の距離 $d(U, W)$ を最小にすることは、

$$\frac{|UW^T|^2 + 2\det(UW^T)}{|U|^2|W|^2} \quad (2)$$

を最大にすることとなる。ここで、 \mathbf{u}_x を図形 U の x 座標を並べた n 次元縦ベクトルとする。同様に、 \mathbf{u}_y を図形 U の y 座標、 \mathbf{w}_x を図形 W の x 座標、 \mathbf{w}_y を図形 W の y 座標を並べたベクトルとして、

$$U = \begin{pmatrix} \mathbf{u}_x^T \\ \mathbf{u}_y^T \end{pmatrix}, W = \begin{pmatrix} \mathbf{w}_x^T \\ \mathbf{w}_y^T \end{pmatrix}, \mathbf{u} = \begin{pmatrix} \mathbf{u}_x \\ \mathbf{u}_y \end{pmatrix}, \mathbf{w} = \begin{pmatrix} \mathbf{w}_x \\ \mathbf{w}_y \end{pmatrix}$$

とする。このとき、(2) 式の分母と分子はそれぞれ、

$$|U|^2|W|^2 = \mathbf{u}^T \mathbf{u} \mathbf{w}^T \mathbf{w}, \quad (3)$$

$$\begin{aligned} |UW^T|^2 + 2\det(UW^T) &= \\ \mathbf{u}^T \begin{pmatrix} \mathbf{w}_x \mathbf{w}_x^T + \mathbf{w}_y \mathbf{w}_y^T & \mathbf{w}_x \mathbf{w}_y^T - \mathbf{w}_y \mathbf{w}_x^T \\ \mathbf{w}_y \mathbf{w}_x^T - \mathbf{w}_x \mathbf{w}_y^T & \mathbf{w}_x \mathbf{w}_x^T + \mathbf{w}_y \mathbf{w}_y^T \end{pmatrix} \mathbf{u} \end{aligned} \quad (4)$$

と書くことができる。ここで対称行列 V を

$$V = \frac{1}{\mathbf{w}^T \mathbf{w}} \begin{pmatrix} \mathbf{w}_x \mathbf{w}_x^T + \mathbf{w}_y \mathbf{w}_y^T & \mathbf{w}_x \mathbf{w}_y^T - \mathbf{w}_y \mathbf{w}_x^T \\ \mathbf{w}_y \mathbf{w}_x^T - \mathbf{w}_x \mathbf{w}_y^T & \mathbf{w}_x \mathbf{w}_x^T + \mathbf{w}_y \mathbf{w}_y^T \end{pmatrix} \quad (5)$$

とすれば、(2) 式は、次の形式で書くことができる：

$$\frac{\mathbf{u}^T V \mathbf{u}}{\mathbf{u}^T \mathbf{u}}. \quad (6)$$

次に制約条件について考える。これはタイリングタイプによって異なるが、ここでは例として IH07 について述べる。IH07 の incidence symbol は、図 4 で与えられる。ここで、図 5 のように点を取ることにする。このとき点の数を n とし、 $N = n/6$ とする。incidence symbol から分か

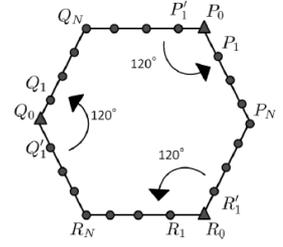
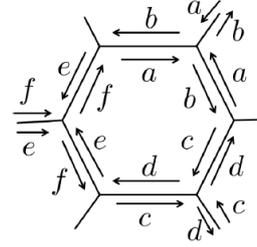


図 4 IH07 の incidence symbol 図 5 境界上の点の関係

るように、すべての辺は J 型であり、同じ辺上の点同士の間には制約がない。また図 4, 図 5 から、 $\angle P_0, \angle Q_0, \angle R_0$ を挟む 2 辺は同じラベルを持つ。よって、タイリング辺は P_0, Q_0, R_0 を中心とする 120 度回転で重ならなければならない。したがって S を 120 度回転を施す行列とすると次式が成り立つ：

$$\begin{cases} S(P'_i - P_0) = P_i - P_0 & (i = 1, \dots, N) \\ S(Q'_i - Q_0) = Q_i - Q_0 & (i = 1, \dots, N) \\ S(R'_i - R_0) = R_i - R_0 & (i = 1, \dots, N). \end{cases} \quad (7)$$

IH07 の場合は回転のみを考慮したが、同様に考えることにより、ほかのタイプで出てくる平行移動・並進鏡映等で重なる条件も定式化できる。また、出力図形の重心が原点に重なるという条件は、 $\mathbf{1}$ を要素がすべて 1 であるような n 次元縦ベクトルとして、次式で記述できる：

$$\begin{cases} \mathbf{u}_x^T \mathbf{1} = 0 \\ \mathbf{u}_y^T \mathbf{1} = 0. \end{cases} \quad (8)$$

(7), (8) 式には定数項が入っておらず、すべて変数の線形結合だけで表されている。したがって、行列 A を用いて、

$$A\mathbf{u} = \mathbf{0} \quad (9)$$

とまとめることができる。IH07 の場合、行列 A のサイズは $(n+2) \times 2n$ である。同様にすべての isohedral タイリングのタイプについて制約条件を (9) 式の形で書くことができる（行列 A のサイズはタイプによって異なる）。

3.3 固有値問題への帰着

以上より、目的関数は (6) 式、制約条件は (9) 式に定式化された。したがってエッシャー風タイリング問題は

$$\begin{aligned} \text{最大化} & \quad \frac{\mathbf{u}^T V \mathbf{u}}{\mathbf{u}^T \mathbf{u}} \\ \text{制約条件} & \quad A\mathbf{u} = \mathbf{0} \end{aligned} \quad (10)$$

という最適化問題と書くことができる。ここで注意が必要なのは、行列 A が isohedral タイリングのタイプによって異なること、入力図形 W の第一頂点の決め方によって、行列 V が変わることである。つまり、(10) は isohedral タイリングのタイプを 1 つに決め、入力図形 W の第一頂点を固定したうえでの問題である。実際は、全ての isohedral タイリングのタイプについて、第一頂点の点を変えながら最適化問題 (10) を解き、全ての場合を計算したのちに、最も目的関数値が大きいものを最適解とすることとなる。

ここで、制約条件を変形する。Ker A の正規直交基底を \mathbf{b}_i ($i = 1, \dots, m$) とすると、制約式 (9) は、

$$\mathbf{u} = \xi_1 \mathbf{b}_1 + \xi_2 \mathbf{b}_2 + \dots + \xi_m \mathbf{b}_m \quad (11)$$

と書ける。行列 B を Ker A の正規直交基底を並べたもの、 ξ を $\xi_1, \xi_2, \dots, \xi_m$ を並べた縦ベクトルとすると、(11) 式は、

$$\mathbf{u} = B\xi \quad (12)$$

と書ける。以上を用いると、最適化問題 (10) は、

$$\text{最大化} \quad \frac{\xi^\top B^\top V B \xi}{\xi^\top \xi} \quad (13)$$

という、無制約最適化問題となる。これは ξ のレイリー商最大化問題であり、対称行列 $B^\top V B$ の最大固有値を求める問題である。この行列の最大固有値に対応する固有ベクトル ξ から、出力図形を表すベクトル $\mathbf{u} = B\xi$ が求まる。

3.4 固有値問題の解析解

本節では、対称行列 $B^\top V B$ の最大固有値と固有ベクトルの求め方について述べる。最大固有値を求める代表的な数値解法としてはべき乗法があるが、ここではべき乗法を使う必要はなく、以下のように解析解が求まる。

まず、行列 V, B の性質について述べる。行列 V は (5) 式で与えられた行列であり、次のように変形できる：

$$V = \frac{1}{\mathbf{w}^\top \mathbf{w}} (\mathbf{w} \mathbf{w}^\top + E \mathbf{w} \mathbf{w}^\top E^\top). \quad (14)$$

ここで、 $E = \begin{pmatrix} O & I \\ -I & O \end{pmatrix}$ であり、 O は零行列、 I は単位行列である。(14) 式から、行列 V は $\text{span}\{\mathbf{w}, E\mathbf{w}\}$ への射影行列であることが分かる。 $\text{span}\{\mathbf{w}, E\mathbf{w}\}$ は \mathbf{w} と $E\mathbf{w}$ によって張られる空間で、その中の任意のベクトルは

$$a\mathbf{w} + bE\mathbf{w} = \begin{pmatrix} aI & bI \\ -bI & aI \end{pmatrix} \mathbf{w} = G\mathbf{w} \quad (a, b \in \mathbb{R})$$

と表される。このベクトルは入力図形 W を回転・伸縮したものに对应する。すなわち、行列 V は、「入力図形 W と同じ形の図形全体からなる空間」への射影行列である。また、行列 B については、 BB^\top という行列を考えると、

$$BB^\top = \sum \mathbf{b}_i \mathbf{b}_i^\top \quad (\mathbf{b}_i \text{ は } B \text{ の列ベクトル})$$

となる。これは $\{\mathbf{b}_i\}$ が Ker A の正規直交基底であることに注意すれば、 $\text{span}\{\mathbf{b}_1, \mathbf{b}_2, \dots\} = \text{Ker}A$ への射影行列である。さらに、Ker A が $A\mathbf{u} = \mathbf{0}$ の解空間、すなわちタイリングできるための条件を満たす空間であることより、行列 BB^\top は、「タイリングできる図形の全体からなる空間」への射影行列となる。これらの性質を用いて、 $B^\top V B$ の最大固有値、固有ベクトルを求めることができる。

定理 3.1 (文献 [9] p.12 定理 3.2) $B^\top V B$ の最大固有値は $\frac{|B^\top \mathbf{w}|^2}{|\mathbf{w}|^2}$ であり、それに対応する固有ベクトルは、 $B^\top \mathbf{w}$ と $B^\top E\mathbf{w}$ の二つである。

以上より、最適化問題 (13) の解を与えるベクトルは $B^\top \mathbf{w}$ と $B^\top E\mathbf{w}$ となることが分かった。また、 $\mathbf{u} = B\xi$ であるので、最終的に求めたい解 (タイル) は、 $BB^\top \mathbf{w}$ と $BB^\top E\mathbf{w}$ となる。ここで、 $BB^\top \mathbf{w}$ と $BB^\top E\mathbf{w}$ は同じ形状となるため、ベクトル $BB^\top \mathbf{w}$ のみを考えればよい。

3.5 計算量

小泉と杉原の解法の計算量について考察する。主に計算時間がかかるのは、行列 B の計算、最大固有値の計算の中の $B^\top \mathbf{w}$ の計算、解ベクトル $BB^\top \mathbf{w}$ の計算である。

行列 B は Ker A の正規直交基底を並べたものであった。行列 A から行列 B を求める計算はグラムシュミットの正規直交化で実現でき、その計算量は $O(n^3)$ である。この計算の計算回数は、行列 A が isohedral tiling のタイプと入力図形の点の数 n のみによって決まる点に注意すると、1 つの入力図形に対して、29 回の計算で十分である。

$B^\top \mathbf{w}$ の計算と $BB^\top \mathbf{w}$ の計算は行列とベクトルの積であり、その計算量はそれぞれ $O(n^2)$ である。 $B^\top \mathbf{w}$ の計算は、タイリングタイプと第一頂点の位置の組合せだけ計算する必要があるので $29 \times n$ 回必要である。一方、 $BB^\top \mathbf{w}$ の計算は、暫定解よりも最大固有値の値が大きい場合のみ計算すればよく、その計算回数は入力図形と計算順序によるが、 $B^\top \mathbf{w}$ の計算よりも少ない回数で済む。

3.6 結果と問題点

ここまでで説明してきた小泉と杉原の解法を用いて、タイリングを生成した結果を図 6、図 7 に示す。この結果は文献 [11] から引用した。(a) は入力した点画で、(b) は出

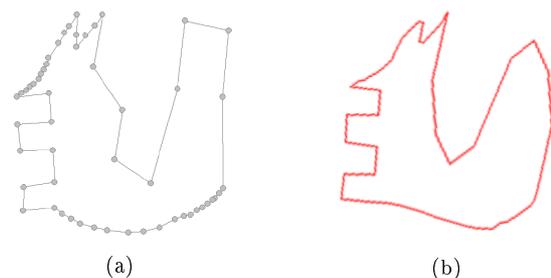


図 6 きつねのタイリング (成功例)

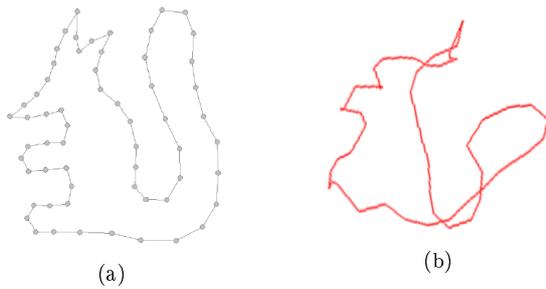


図7 きつねのタイリング (失敗例)

力された最適解である。図6を見ると、入力図形に近いタイルが出力されていることが分かる。一方、図7では、入力図形と出力図形があまり近いとは言えず、また、タイルが自己交差を持っていることが分かる。つまり、小泉らの解法には、入力図形の点の配置によって、似た図形でも出力されるタイルの質が大きく変わるといった課題や、自己交差を持つ図形が出力されるといった問題点がある。

4. 改善手法

4.1 自己交差に対するペナルティの導入

まず、図7(b)のように、自己交差を持つ図形が出力される問題について考える。この原因は(9)式で表される制約条件にある。先ほどと同様 IH07 を例にとって説明する。IH07 は 120 度回転によって各辺が重なるタイプであり、制約条件は(7)式で表された。しかし、例えば図8のような自己交差を持つ図形もこの制約条件を満たす。そのため、自己交差を持つ図形も解として出力されてしまうことがある。同じことが、平行移動や並進鏡映を制約条件に持つ場合でも言える。この問題を解決するためには、制約

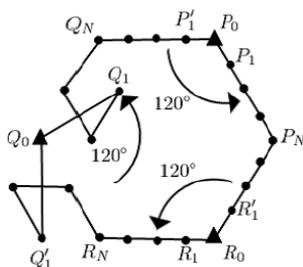


図8 自己交差を持つ図形 (IH07)

条件に、図形が自己交差を持たないという条件を付け加えなければならないが、小泉らの解法では制約条件が(9)式で表されることが重要であったため、上述の制約条件を組み込むのは難しい。そこで、自己交差があった場合に、目的関数にペナルティとして課し、その図形が選ばれないようにする。図形 $BB^T \mathbf{w}$ が自己交差を持つ場合 $\alpha = 0.1$ 、自己交差を持たない場合 $\alpha = 0$ とし、目的関数を

$$\frac{|B^T \mathbf{w}|^2}{|\mathbf{w}|^2} - \alpha \quad (15)$$

とする。本研究では、自己交差の検出には平面走査法 [1]

を用いる ($O(n \log n)$ の計算量で交差の検出を行う)。なお、自己交差を持つタイルを完全に排除しても良いが、後述する局所探索を用いる場合は、ペナルティを課して解候補補に加える方が多い。以下では、(15) 式を新たな目的関数とする。

4.2 点の位置を変える方法

次に、似た入力図形でも点の配置によって解の質が変わるという課題について考える。この原因も制約条件にある。小泉らの提案した制約条件は、図5や(7)式を見ると分かるように、各タイリング辺上の点の数が等しくなければならない。つまり、1つのタイリング頂点の位置が決まると、残りのタイリング頂点の位置も自動的に決まることになり、タイリング頂点の位置に関する自由度が低いと言える。そのため、似た図形でも点の配置が異なるとタイリング頂点の位置が変わり、その結果として、出力される図形が大きく異なると考えられる。

この問題を改善するため、本節では点の配置を変える手法を提案する。(入力図形を近似する多角形の生成法を提案すると言い換えることもできる。)なお、次節では、同じ目的のため制約条件を緩和する手法を提案する。線画として与えられた入力図形に対して、それを近似する多角形は無数に存在する。本研究では、良いタイルが得られる多角形(点の配置)を、以下の局所探索法を用いて探索する。

局所探索法を用いるためには、近傍を定義する必要がある。本稿では、現在の点画に対して、点の配置を少し変えた点画の集合を近傍とする。点配置を変える際、図形の形(見た目)が変わらないように配置を変えることが求められる。そのため、入力図形の線画を適当な点画として近似し、この点画に対して次の2つの前処理を行う。

- 前処理1 点画の各点を固定点と可動点に分類分けを行う。
- 前処理2 前処理1を行った後に、各固定点の間にスプライン補間を行う。そして各可動点を補間関数の上に x 座標が等分となるように配置する。

前処理の詳しい説明は本稿では省略する(文献 [4], [12] を参照)が、図9の図形に対して前処理を行うと、図10となる(黒点が固定点、白点が可動点を表す)。

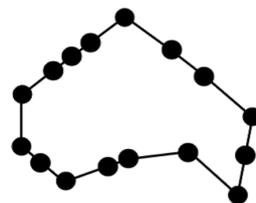


図9 前処理前の図形

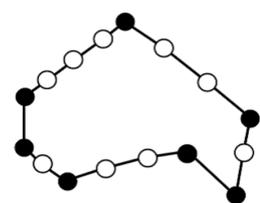


図10 前処理後の図形

次に、前処理後の初期図形の点の配置を変えることを考える。前処理によって、初期図形の点は固定点と可動点に

分類分けされ、各固定点の間はスプライン補間が行われている。ここで、前処理2の方法で可動点を配置することとすれば、各固定点間の可動点の数 m_q を決めることで、可動点は配置される。すなわち、 $m_q (q = 1, \dots, l)$ の組合せ

$$\{(m_1, m_2, \dots, m_l) \mid \sum_{q=1}^l m_q = n - l, m_q \in \mathbb{N}\} \quad (16)$$

が決まれば、ベクトル \mathbf{w} は一意に定まる (\mathbb{N} は非負整数の集合)。ここで、ベクトル \mathbf{w} の近傍 $N(\mathbf{w})$ を、

$$N(\mathbf{w}) = [m_q \neq 0 \text{ となる一つの要素に対し}, \\ \{(m_q - 1) \wedge (m_{q-1} + 1)\} \vee \{(m_q - 1) \wedge (m_{q+1} + 1)\}] \\ \text{としたときに得られるベクトル}$$

と定義する (m_0 は m_l を、 m_{l+1} は m_1 を表すとする)。

以上より近傍が定義できたので、この近傍を用いて局所探索を行い、目的関数が最大となる図形を求める。まず、入力図形を適当な点画とし、その図形に対して前処理を行い、前処理後の図形に対してタイルを求める。これを初期解とする。次に、近傍内の図形に対して目的関数を計算し、目的関数値が大きくなるものが得られたとき、そのタイルを暫定解とする。この操作を繰り返し、近傍内に改善解が得られる図形が存在しなくなったとき、その解を出力する。

この手法を用いて計算実験を行った結果を図11に示す。(a) は入力図形の点画、(b) は (a) から求まるタイリング図形、(c) は局所探索後の点画、(d) は (c) から求まるタイリング図形である。これを見ると、局所探索を適用する前の点画よりも、局所探索を適用した後の点画の方が、入力図形に近い解が得られていることが確認できる。

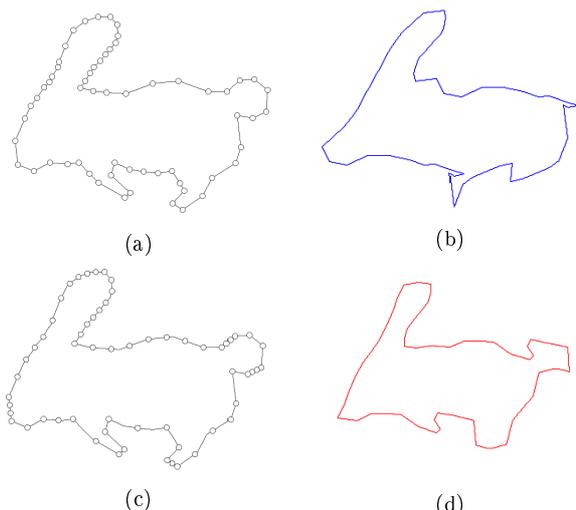


図11 うさぎのタイリング

4.3 制約条件の緩和

前節とは異なる改善手法を提案する。これは、制約条件を定式化する際に、タイリング辺上の点の数に自由度を持たせることで制約条件を緩和するといった方法である。タイリング辺上の点の数に自由度を持たせれば、タイリング頂点の位置の自由度が高くなるので、出力される解の改善が期待される。

ここで IH07 を例にとって考える。図5のタイリング辺上の点の数に自由度を持たせると、図12のようになる。

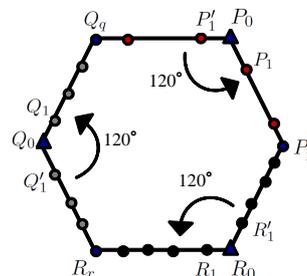


図12 タイリング辺上の点の数に自由度を持たせた図形

このとき、(7) 式は次のように変化する：

$$\begin{cases} S(P'_i - P_0) = P_i - P_0 & (i = 1, \dots, p) \\ S(Q'_i - Q_0) = Q_i - Q_0 & (i = 1, \dots, q) \\ S(R'_i - R_0) = R_i - R_0 & (i = 1, \dots, r). \end{cases} \quad (17)$$

この式は (7) 式と同様に、(8) 式と合わせることで $\{A_k \mathbf{u} = \mathbf{0} \mid 1 \leq k \leq (p, q, r) \text{ の組合せ}\}$ の形で書くことができる。 p, q が決まると r は一意に決まることに注意すれば、 p, q, r の組合せは $O(n^2)$ 通り存在する。こうしてタイリング辺上の点の数に自由度を持たせることで、解の候補は、

$$\{B_{ik} B_{ik}^\top \mathbf{w}_j \mid 1 \leq i \leq 29, 0 \leq j \leq n - 1, k = 1, 2, \dots\} \quad (18)$$

となる。 i はタイリングタイプ、 j は第一頂点の位置、 k は点の数の自由度に対応する。この中で目的関数値が最大となる p, q, r の組合せを求めればよいが、タイリングタイプによっては k の範囲が $O(n^3)$ や $O(n^4)$ となるものもある。3.5 節で述べたように、行列 B を求める計算は $O(n^3)$ 時間を要するため、(18) 式のすべてのタイルを求めると計算時間が膨大となる。そこで、ある程度質の良いタイルを近似的に求めることとし、その手法としては前節と同様、局所探索法を用いる。近傍は、いずれかのタイリング辺上の点の数を1増やし、他の辺上の点を減らす操作で定める。つまり、IH07 の場合は、タイリング辺上の点の数を (p, q, r) とすると、近傍の全組合せは以下のようになる：

$$(p + 1, q - 1, r), (p + 1, q, r - 1), (p - 1, q + 1, r), \\ (p, q + 1, r - 1), (p - 1, q, r + 1), (p, q - 1, r + 1).$$

この近傍を用いて局所探索を行う。タイリング辺上の点の数をすべて等しくしたときの制約条件行列を $A^{(1)}$ とし、

$A^{(1)}$ から得られた解を初期解とする．次に，上で述べた近傍内の制約条件行列 $A \in N(A^{(1)})$ に対して目的関数値を求める．目的関数値が大きくなるようなタイルが得られたとき，それを暫定解として行列の更新を行う．この操作を繰り返し，近傍内の全ての制約条件行列 A に対して暫定解の更新が行われなくなったとき，その解を出力する．なお，行列 A から行列 B を求める際に，行列の類似性を用いた計算の高速化を組み込んでいる．

この手法を用いて数値実験を行った結果を図 13 に示す．(a) は入力図形，(b) は小泉らの解法で求まるタイル，(c) は提案手法で求めたタイル，(d) はそのタイリングである．

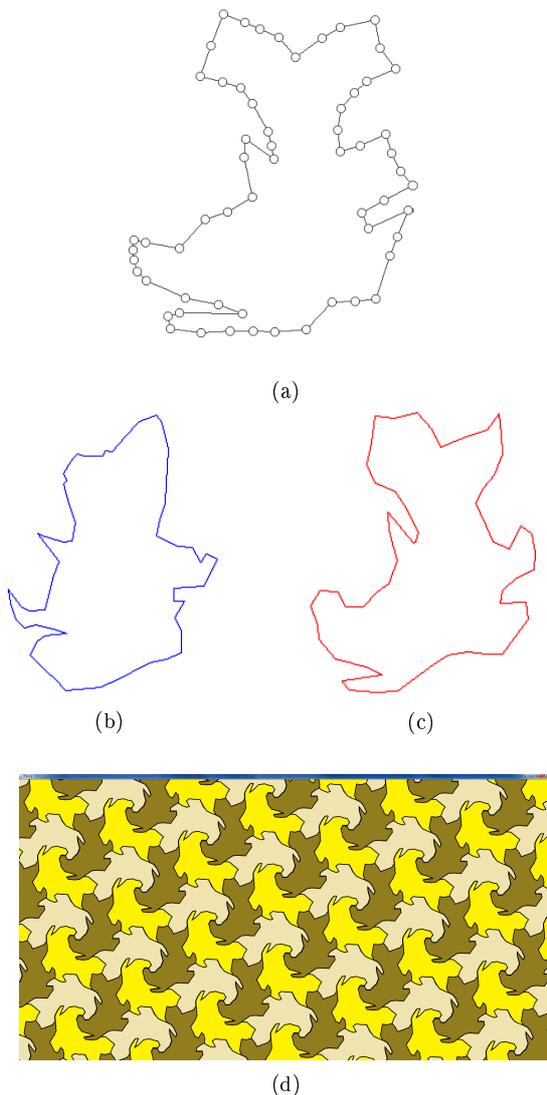


図 13 シャチほこのタイリング

図 13 より，制約条件の緩和による解精度の向上が確認できる．しかし，この手法にも問題点が存在する．例として挙げた IH07 の近傍サイズは 6 と小さい．タイリングタイプによってはさらに近傍サイズが小さいものも存在し，局所探索がうまく機能しないことがある．また，タイリン

グ边上の点の数に自由度を持たせたが，対応する辺同士の点の数は同じでなければならないため，タイリング頂点の位置はまだ制限されていると言える．

5. まとめ

本稿では，入力図形が与えられ，その図形にできるだけ近い図形によるタイリングを求める問題に対して，小泉らの手法 [10] と局所探索のアイデアを組み合わせることで，より質の高いタイリングを求める手法を提案した．計算実験により提案手法を評価したところ，既存解法と比較して，より精度の高い解を得られることが確認できた．計算時間は点の数や図形の形状に依存するが，計算の効率化のアイデアを組み込むことで，数秒から数分程度と，実用的な時間でタイリングを得られることを確認した．

今後の課題のひとつとして，2 種類の図形に対するエッシャー風タイリング [7] を求める，高性能，高速アルゴリズムの開発があげられる．

参考文献

- [1] M. ドバーク, M. ファン, クリベルド, M. オーバマーズ, O. シュワルツコップ 著 浅野 哲夫 訳: コンピュータ・ジオメトリ - 計算幾何学: アルゴリズムと応用 -, 近代科学社, 2000.
- [2] M. C. Escher: M. C. エッシャー グラフィック, タッシェン・ジャパン, 2008.
- [3] B. Grünbaum and G. C. Shephard: Tiling and Patterns, W. H. Freeman, 1987.
- [4] S. Imahori, S. Sakai: A Local-Search Based Algorithm for the Escherization Problem, The IEEE International Conference on Industrial Engineering and Engineering Management, 2012, 151-155.
- [5] C. S. Kaplan: Introductory Tiling Theory for Computer Graphics, Morgan & Claypool Publishers, 2009.
- [6] C. S. Kaplan and D. H. Salesin: Escherization, Proceedings of SIGGRAPH, 2000, 499-510.
- [7] C. S. Kaplan and D. H. Salesin: Dihedral Escherization, Proceedings of Graphics Interface, 2004, 255-262.
- [8] D. G. Kendall: Shape Manifolds, Procrustean Metrics, and Complex Projective Spaces, Bulletin of the London, Mathematical Society, 16(2), 1984, 81-121.
- [9] 小泉 拓: エッシャー風タイリングの自動生成, 東京大学大学院情報理工学系研究科数理情報学専攻 修士論文, 2010.
- [10] H. Koizumi and K. Sugihara: Maximum Eigenvalue Problem for Escherization, Graphs and Combinatorics, 27, 2011, 431-439.
- [11] 酒井 翔平: 局所探索を用いたタイリング自動生成法の改善, 名古屋大学工学部物理工学科 卒業論文, 2011.
- [12] 酒井 翔平: エッシャー風タイリング問題の数理モデルについて - 制約条件の緩和及びその最適化手法 -, 名古屋大学大学院工学研究科計算理工学専攻 修士論文, 2013.