

# 与えられたどの凸多角形にも交差しない直線全体を構成するアルゴリズム

戸田貴久\*

2010年2月3日

## 1 はじめに

いくつかの種類異なる物体の集まりが平面上に配置されているとき、それらを種類ごとに区別することを考えてみたい。同じ種類の物体同士が近くにまとまっているとき、人間ならば大雑把に区別することが可能だろう。それでは、コンピュータにさせるにはどうすれば良いだろう。

まず前提として、物体の位置・概形は、画像センサなど各種センサを用いて認識できるものとする。このとき、互いに離れて配置されている2種類の物体の集まりを区別するための、おそらくもっとも単純な方法の1つは、2つを分離する直線を1つ求めることである。

2種類だけでなく、複数種類の場合も考察してみたい。類別が目的なので、直線は同一種類の物体同士を分離しないものが望ましい。そのような直線の中で、複数種類の物体の集まりを異なる仕方で分離する直線は、一般に複数存在する(図1)。したがって、本質的に異なる分類が可能なすべての分離直線を求めれば良いことになる。

以上の考察の下で、我々の問題を数学的に表現してみたい。単純化のため、同一種類の物体の集まりは、有限個の点集合とみなす<sup>1</sup>。このとき、我々の問題は、 $k$ 種類の有限点集合  $S_1, \dots, S_k$  が与えられるとき、それらによって張られるどの凸多角形  $[S_i]$  ( $1 \leq i \leq k$ )

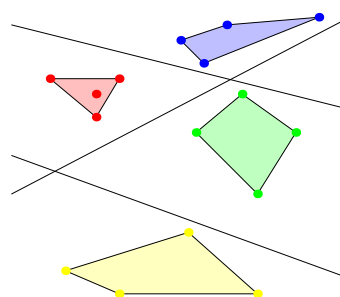


図1: 4個の凸多角形のうちどれとも交差しない3つの直線。これらは、互いに異なる仕方で4個の凸多角形を分離している。

とも交差しない直線に関する問題を考察することである。

本予稿では、特に、どの凸多角形  $[S_i]$  ( $1 \leq i \leq k$ ) とも交差しない直線全体の表現の構成問題を扱う。平面上の点配置に関する組合せ的な性質が、双対変換により、直線のアレンジメントとしてより扱いやすい形に表現されるように、我々は凸多角形の配置に関する組合せ的な性質を、双対変換により別の形で表現する。本予稿では、まずこのように変換して得られる集合のもつ幾何学的な性質を考察する。その後、その性質を用いて我々の問題を求めるアルゴリズムを設計し、漸近的な計算時間を解析する。

我々の扱う問題とは対照的に、与えられた物体をすべて貫通する直線に関する問題(横断線問題)は、これまで研究されてきている。横断線問題は、計算幾何学において双対変換の応用例としてよく知られ

\*京都大学大学院人間・環境学研究科博士後期課程

<sup>1</sup>各物体を多角形とみなし、その頂点集合によりその物体を表す。同一種類の物体の集まりは、各物体を表す頂点集合の和集合とみなされる。

ている [1] [2] [3]。また、組合せ幾何学の観点から、与えられた物体を貫通する直線全体として表される集合のもつ組合せ的な性質（ヘリー性）の研究も行われている [4] [6] [7]。

## 2 準備

計算幾何学から、いくつかの基本事項を導入する [1] [2] [3]。平面  $\mathbb{R}^2$  上において、原点とは異なる点と原点を通らない直線との 1 対 1 の対応を

$$\text{点} : (a, b) \neq (0, 0) \iff \text{直線} : ax + by = 1$$

として導入する。便宜的に、 $\mathcal{D}_o(p)$  により、点  $p$  に対応する直線を表し、 $\mathcal{D}_o(l)$  により、直線  $l$  に対応する点を表すこととする。上で与えられる  $\mathcal{D}_o$  は原点  $o$  に伴われる、点と直線間の双対変換と呼ばれる。

次の命題は、双対変換  $\mathcal{D}_o$  が、点と直線間の位置関係を保存することを意味している。原点を通らない直線  $l$  に伴われる 2 つの開半平面のうち原点を含まない方を  $l^+$  と表し、他方の開半平面を  $l^-$  と表す。

**命題 2.1** 平面上で、 $l$  を原点を通らない直線、 $p$  を原点とは異なる点とする。点  $p$  がそれぞれ  $l^+, l, l^-$  に含まれることは、点  $\mathcal{D}_o(l)$  がそれぞれ  $\mathcal{D}_o(p)^+, \mathcal{D}_o(p), \mathcal{D}_o(p)^-$  に含まれることと同値である。□

**定義 2.1** 写像  $\Phi_o : 2^{\mathbb{R}^2} \rightarrow 2^{\mathbb{R}^2}$  は、任意の部分集合  $S \subseteq \mathbb{R}^2$  に対して、次の集合

$$\Phi_o(S) := \{ \mathcal{D}_o(l) \in \mathbb{R}^2 \mid l \text{ は } S \text{ と交差せず、} \\ \text{原点を通らない直線} \}$$

を対応させる写像とする。写像  $\Phi_o$  の像として表される集合を非交差集合と呼ぶ。特に、 $\Phi_o(S)$  のことを  $S$  に対する非交差集合と呼ぶ。

平面上に  $k$  本の直線からなる集合  $H := \{l_1, \dots, l_k\}$  が与えられるとき、平面はこれらの直線によっていくつかの連結な部分に分割される（図 2）。この分割をアレンジメントと呼び、 $\mathcal{A}(H)$  と表す。アレンジ

メントは、分割によって現れる交点・開線分・開領域（セル）を頂点とし、それらの間の直接的な<sup>2</sup>接続関係を辺とするグラフとして表現される。これを接続グラフと呼ぶ。

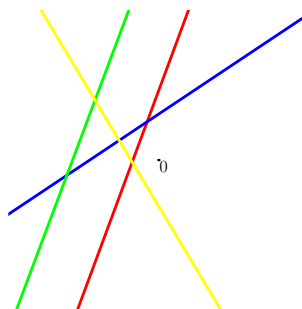


図 2: 4 直線のアレンジメント

**定理 2.1** 平面上で、 $k$  本の直線のアレンジメントの接続グラフは  $O(k^2)$  の時間と記憶量で構成できる。□

$C$  をアレンジメント  $\mathcal{A}(H)$  のセルとする。このとき、それぞれ  $\deg_0(C), \deg_1(C)$  をセル  $C$  に接続する頂点、辺の数を表すとする。

**定理 2.2**  $H$  を  $k$  本の直線からなる集合とする。 $C_1, \dots, C_m$  をアレンジメント  $\mathcal{A}(H)$  のすべてのセルとする。このとき、

$$\sum_{i=1}^m \deg_0(C_i) \cdot \deg_1(C_i) = O(k^2)$$

が成り立つ。□

## 3 凸多角形に関する非交差直線問題

いくつかの凸多角形が与えられるとき、どの凸多角形とも交差しない直線に関するさまざまな計算問

<sup>2</sup>交点と開線分、あるいは、開線分と開領域との間の接続関係を意味している。交点と開領域の間の接続関係は、ここでは認めない。

題が自然に考えられる。本予稿ではこれらをまとめて、"凸多角形に関する非交差直線問題"と呼ぶ。本節では、これらのうち、どの凸多角形とも交差しないすべての直線の表現を構成する問題を扱う。

問題 3.1 平面  $\mathbb{R}^2$  上で、有限点集合  $S_1, \dots, S_k$  が与えられるとき、これらにより張られるどの凸多角形  $[S_i]$  ( $1 \leq i \leq k$ ) にも交差しないすべての直線の表現を構成せよ。

本予稿では、問題 3.1 において与えられる点集合の中に原点を含むものと仮定する。この仮定は、本質的に重要なものではなく、議論を単純化するために設けるだけである。

### 3.1 非交差集合の幾何学的性質

与えられたどの凸多角形とも交差しない直線全体は、双対変換  $\mathcal{D}_o$  により、平面上の点集合として表現される。このように表現された点集合は、定義 2.1 によれば、有限個の凸多角形の和集合に対する非交差集合として言い表される。本小節では、有限個の凸多角形の和集合に対する非交差集合の幾何学的性質を扱う。

このような点集合は有限個の連立不等式を使って表示される。次の命題は、1つの凸多角形に対する非交差集合は、一般に2つの連立不等式によって特徴づけられることを示している(図3)。

命題 3.1 有限点集合  $S = \{(a_i, b_i) \in \mathbb{R}^2 \mid 1 \leq i \leq r\}$  に対して、凸多角形  $[S]$  の非交差集合  $\Phi_o([S])$  は次の2つの連立不等式 (+), (-) の各解集合の和として表される。

$$(+)\begin{cases} a_1x + b_1y > 1 \\ \dots \\ a_rx + b_ry > 1 \end{cases} \quad (-)\begin{cases} a_1x + b_1y < 1 \\ \dots \\ a_rx + b_ry < 1 \end{cases}$$

□

補足 3.1 もし与えられた集合  $S$  に属する点  $(a_i, b_i)$  が原点ならば、2つの条件  $a_ix + b_iy > 1$ ,  $a_ix + b_iy < 1$  は、それぞれ集合として  $\emptyset, \mathbb{R}^2$  と解釈され、この命題は成り立つ。一般に、凸多角形  $[S]$  が原点を含むならば、その非交差集合は、有限個の開半平面の共通部分として表される凸集合である。

以下では、上の連立不等式 (+) の解集合  $\bigcap_{p_i \in S} \mathcal{D}_o(p_i)^+$  を  $\Phi_o([S])^+$  により表し、(-) の解集合  $\bigcap_{p_i \in S} \mathcal{D}_o(p_i)^-$  を  $\Phi_o([S])^-$  により表すことにする。

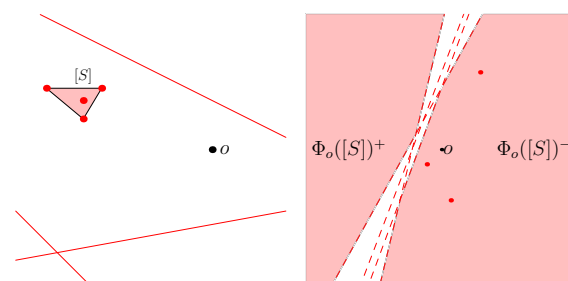


図3: 凸多角形  $[S]$  と原点  $o$  を分離しない3直線(左)  $[S]$  の非交差集合  $\Phi([S])$  と左図の3直線に対応する3点(右)

次の命題は解集合  $\Phi_o([S])^+, \Phi_o([S])^-$  の幾何学的な意味を表している。

命題 3.2  $S$  は平面上の有限点集合とする。

1.  $p \in \Phi_o([S])^+$  ならば、直線  $\mathcal{D}_o(p)$  は原点  $o$  と点集合  $S$  を分離する。  
(一方の開半平面が原点  $o$  を、他方が  $S$  を含む。)
2.  $p \in \Phi_o([S])^-$  ならば、直線  $\mathcal{D}_o(p)$  は原点  $o$  と点集合  $S$  を同じ側に含む。  
(一方の開半平面が両者を同時に含む。)

□

次の命題により、複数の凸多角形の和集合の非交差集合は、個々の凸多角形の非交差集合の共通部分として表されることが分かる。

命題 3.3 任意の部分集合  $T_1, T_2 \subseteq \mathbb{R}^2$  に対して、

$$\Phi_o(T_1 \cup T_2) = \Phi_o(T_1) \cap \Phi_o(T_2)$$

が成り立つ。 □

本小節の残りの部分では、非交差集合の組合せ幾何学的な考察を行う。Edelsbrunner [3] が、横断線問題を求めるアルゴリズムの設計において問いかけた言葉を借りるならば、我々の問いかけは次のようになる。「総数  $n$  個の点で張られる  $k$  個の凸多角形の和集合に対する非交差集合の複雑さはどのくらいか？すなわち、非交差集合を構成するだけの価値があるのだろうか？」

命題 3.4 平面上において、 $k$  個の凸多角形で、少なくとも1つの凸多角形は原点を含むものが与えられるとする。これら凸多角形の和集合に対する非交差集合は、高々  $\frac{1}{2}k^2 - \frac{1}{2}k + 1$  個の連結成分からなり、各連結成分は高々有限個の開半平面の共通部分として表される凸集合である。 □

命題 3.5 平面上において、 $k$  個の凸多角形で、少なくとも1つの凸多角形は原点を含むものが与えられるとする。これら凸多角形の和集合に対する非交差集合において、その境界はそれぞれ高々  $O(nk)$  個の辺、頂点からなる。ただし、与えられた点の総数を  $n$  で表す。 □

### 3.2 アルゴリズム

本小節では、問題 3.1 を求めるアルゴリズムを設計し、その解析を行う。

以下のアルゴリズムは、初期化段階と反復段階からなる。初期化段階では、問題の組合せ構造をあらかじめ計算するために、アレンジメントを構成する。反復段階では逐次添加法により非交差集合を構成する。

アルゴリズム中に用いられる記法を導入する。平面上の点を  $p_j$  で表す。添字  $j$  は、 $p_j$  が  $S_j$  に所属することを表す。初期化段階で構成したアレンジメン

ト  $\mathcal{A}(H)$  において、各セル  $C$  の第  $i$  番目の符号  $v_i(C)$  を

$$v_i(C) := \begin{cases} + & C \subseteq \mathcal{D}_o(\hat{p}_i)^+ \text{ のとき} \\ - & \hat{p}_i = o \text{ または } C \subseteq \mathcal{D}_o(\hat{p}_i)^- \text{ のとき} \end{cases}$$

と定義する。

初期化段階：

1. 各有限点集合  $S_i$  ( $1 \leq i \leq k$ ) から1点づつ代表元  $\hat{p}_i$  を選ぶ。
2. 直線集合  $H := \{\mathcal{D}_o(\hat{p}_i) \mid \hat{p}_i \neq o, 1 \leq i \leq k\}$  に関するアレンジメント  $\mathcal{A}(H)$  を構成する。
3. アレンジメント  $\mathcal{A}(H)$  に関する各セル  $C$  の符号ベクトル  $(v_1(C), \dots, v_k(C))$  を計算する。
4. アレンジメント  $\mathcal{A}(H)$  の各セル  $C$  に含まれる候補解の集合  $\text{Sol}(C)$  に十分大きな初期値 (例えば  $\mathbb{R}^2$ ) を代入する。

反復段階：

5. for 点  $p_j$  in  $\bigcup_{i \leq k} S_i \setminus \{o\}$  do  
for セル  $C$  in  $\mathcal{A}(H)$  do  
case A: 直線  $\mathcal{D}_o(p_j)$  が  $C$  と交差または隣接<sup>3</sup>するとき  
 $\text{Sol}(C) := \text{Sol}(C) \cap \mathcal{D}_o(p_j)^{v_j(C)}$ ;  
case B:  $C \subseteq \mathcal{D}_o(p_j)^-$  かつ  $v_j(C) = +$  または  
 $C \subseteq \mathcal{D}_o(p_j)^+$  かつ  $v_j(C) = -$  のとき  
 $\text{Sol}(C) := \emptyset$ ;  
done  
done

補足 3.2 無限遠直線を表す記号  $l_\infty$  を、以下の条件を満たすように導入すれば、上述のアルゴリズムから原点に関する例外を取り除くことができる。無限遠直線  $l_\infty$  は、 $\mathbb{R}^2$  上のどの点も通らず、それに伴われる開半平面は  $\mathbb{R}^2, \emptyset$  である。そして、 $\mathcal{D}_o$  により原点  $o$  は  $l_\infty$  に写されるものとする。

<sup>3</sup>直線  $l$  がセル  $C$  に "隣接する" とは、 $l$  の一方の開半平面が  $C$  を含み、かつ  $l$  自身が  $C$  の (少なくとも1つの) 境界点を含む、という意味で用いている。

命題 3.6 上述のアルゴリズムは、凸多角形  $[S_1], \dots, [S_k]$  の和集合に対する非交差集合を構成する。

証明 求めるべき非交差集合 (真の解集合)  $\Phi_o(\bigcup_{i \leq k} [S_i])$  は、アレンジメント  $\mathcal{A}(H)$  のすべてのセルによって被覆されているので、各セル  $C$  に対して、

$$\text{Sol}(C) = C \cap (\Phi_o(\bigcup_{i \leq k} [S_i]))$$

を示せば十分である。非交差集合は

$$\begin{aligned} \Phi_o(\bigcup_{i \leq k} [S_i]) &= \bigcap_{i \leq k} \Phi_o([S_i]) \\ &= \bigcap_{i \leq k} (\Phi_o([S_i]^+) \cup \Phi_o([S_i]^-)) \end{aligned}$$

として表され、 $\Phi_o([S_i]^+)$  と  $\Phi_o([S_i]^-)$  は互いに素であるので、点  $x$  が解集合  $\Phi_o(\bigcup_{i \leq k} [S_i])$  に属することは、次のように言い換えられる：

各  $i \leq k$  に対して、 $S_i$  のすべての点  $p_i$  が  $x \in D_o(p_i)^+$  を満たすか、または  $S_i$  のすべての点  $p_i$  が  $x \in D_o(p_i)^-$  を満たすかのいづれかである。

ゆえに、初期化段階で選んだ代表元  $\hat{p}_j$  に対して定まるセル  $C$  の符号が  $v_j(C) = +$  ならば、反復段階で点  $p_j$  を添加したとき、 $C$  に属する解は  $D_o(p_j)^+$  に含まなければならない (図 4)。  $v_j(C) = -$  の場合も同様である。よって、セル  $C$  における候補解の集合  $\text{Sol}(C)$  は、セル  $C$  に含まれる真の解をすべて含むことが導かれる。

逆に、添加する各点  $p_j$  に関してこのように更新処理を続ければ、真の解だけを表現できること (偽の解が  $\text{Sol}(C)$  に混入しないこと) は明らかである。□

定理 3.1 平面  $\mathbb{R}^2$  上で、有限点集合  $S_1, \dots, S_k$  が与えられるとき、これらの点集合により張られる凸多角形  $[S_i]$  ( $1 \leq i \leq k$ ) にも交差しない直線全体の表現は、 $O(nk(\log n + k))$  の時間と  $O(nk)$  の記憶量で構成できる。ただし、与えられた点の総数を  $n := \sum_{i \leq k} |S_i|$  で表す。

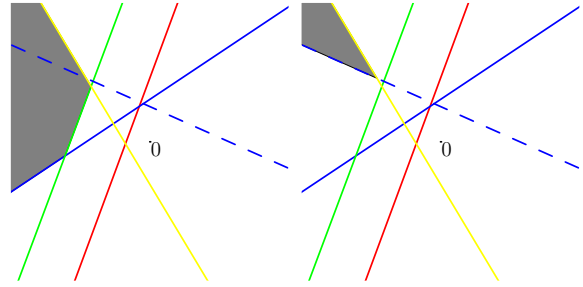


図 4: 灰色の更新前の候補解の集合  $\text{Sol}(C)$  はセル  $C$  自身とする (左)、青色の点線  $D_o(p_j)$  を添加したとき、 $\text{Sol}(C)$  が更新される (右)

証明 定理 2.1 から、初期化段階は  $O(k^2)$  の時間と記憶量で計算できる。以下では反復段階の解析を行う。

点  $p_j$  を添加するときにかかる計算時間を考察する。各セル  $C$  ごとに、直線  $D_o(p_j)$  とセル  $C$  との位置関係を判定して、場合 A、場合 B、あるいは、何もしないで次に進むかの 3 つの場合に処理を分岐させる。そのために、セル  $C$  に接続する頂点と直線  $D_o(p_j)$  の位置関係をみて、次のように判定すれば良い。

1. + 符号を持つ頂点と - 符号を持つ頂点が存在するならば、直線とセルは交差している。
2. 直線上にいくつかの頂点があり、それ以外はすべて + 符号あるいはすべて - 符号ならば、直線とセルは隣接している。
3. すべての頂点が  $v_j(C)$  と反対の符号ならば、場合 B が該当する。
4. すべての頂点が  $v_j(C)$  と同じ符号ならば、どの場合にも該当しない (何もなくて良い)。

すべてのセルに対してこの判定をするために、定理 2.2 より  $O(k^2)$  の時間がかかる。

場合 A が該当するのは  $O(k)$  回である。場合 A が該当したとき、セルの候補解の集合を更新する。このために、Jiagang et al [8] を用いれば  $O(\log n)$  時間で計算できる。一方、場合 B が該当したとき、そのセルには解がないと分かるので、空集合とする。よって、

点  $p_j$  を添加したときにかかる時間は  $O(k^2 + k \log n)$  である。ゆえに、反復段階の計算時間は  $O(nk(\log n + k))$  である。

次に、反復段階で必要な記憶量を解析する。各セル  $C$  ごとに候補解の集合  $S(C)$  を頂点のリストとして記憶しておく必要がある（このときのデータ構造は Jiagang et al [8] に従う）。計算過程で必要となる頂点の総数は  $O(nk)$  である（命題 3.5）。これがアルゴリズム全体の記憶量を定める。□

## 参考文献

- [1] 浅野哲夫著, 『計算幾何学』, 朝倉書店, 2002.
- [2] 今井 浩・今井桂子著, 『計算幾何学』, 共立出版, 1996.
- [3] H.Edelsbrunner 著, 今井 浩・今井桂子訳, 『組合せ幾何学のアルゴリズム』, 共立出版, 1995.
- [4] J.Matousek 著, 岡本吉央訳, 『離散幾何学講義』, Springer, 2005.
- [5] M.de Berg and M.van Kreveld and M.Overmars and O.Schwarzkopf 著, 浅野哲夫訳, 『コンピュータ・ジオメトリ』, 近代科学社, 2000.
- [6] Raphael Wenger, Handbook of Discrete and Computational Geometry, Helly-Type Theorem and Geometric Transversals, CRC Press, 1997, 375–393.
- [7] Raphael Wenger, *Progress in geometric transversal theory*, Contemporary Mathematics, 1999.
- [8] WU Jigang and JI Yongchang and CHEN Guoliang, *An Optimal Online Algorithm for Halfplane Intersection*, Journal of Computer Science and Technology, vol.15, 2000, 295–299.