

## 複合長方形領域の最小分割†

大 附 辰 夫\*\* 佐 藤 政 生\*\*  
橋 昌 良\*\* 鳥 居 司 郎\*\*\*

複合長方形領域を重複なく最小個の長方形に分割する問題を扱う。ここでは、複合長方形領域が中空部分(窓)を含んだり、複数個の連結成分から成っているような一般的な場合を考察する。分割手順は二つのアルゴリズムから成っている。1番目のアルゴリズムは、縮退していない複合長方形領域を最小個の長方形に分割するものである。同じXまたはY座標上の二つの凹点間が領域内であるとき、複合長方形領域は縮退しているといい、そうでない場合には縮退していないという。2番目のアルゴリズムは、与えられた(縮退している)複合長方形領域を最適にいくつかの縮退していない複合長方形領域に分解するものである。複合長方形領域の頂点の数を $n$ とすると、1番目のアルゴリズムの計算複雑度は $O(n \log n)$ となり、2番目のアルゴリズムは $O(n^{5/2})$ となることを報告する。ここで扱う問題は、LSIのネットワーク処理、画像処理、図形データベースなどに於ける基本的問題の一つである。

### 1. ま え が き

デジタル座標(整数)をもつ平面上で、水平または垂直な線分で囲まれた図形——以後、**複合長方形領域**という——を重複なく最小個の長方形に分割する問題は、LSIのネットワーク処理、画像処理、図形データベースなどにおける基本的問題の一つである。たとえば、LSIネットワーク処理におけるマスク原図の自動作画においては、パターンジェネレータのオーバーチャが長方形であるので、自動的にマスク上に与えられた図形を露光するためには、この図形をいくつかの長方形に分割する必要がある。また、LSIのレイアウト問題においてチップの配線領域をチャンネルモデル<sup>1)</sup>によって表現して、配線のチャンネル割付けをする場合にも、このような分割が必要になる。近年のLSIの集積度の著しい増大に対処するためには、十分規模の大きい問題に対して能率のよいアルゴリズムが求められている。

従来、複合長方形領域の分割問題に対して、最小分割数を与える公式やヒューリスティックな分割アルゴリズムが提案されている<sup>2)-4)</sup>が、いずれも扱う図形の一般性、計算の能率、あるいは、計算の手間の評価などの点で不完全であった。たとえば、文献2)、3)では計算の手間が明確に評価されていない。筆者が見積も

るとこれらのアルゴリズムの計算複雑度は $O(n^2)$ である。また後述の縮退がある場合には最小分割を保証しない。文献4)では縮退がある場合の理論的定式化がされているが、領域が窓を含む場合についての考察がされていない。また計算の手間についてはまったく論じられていない。本論文では、一般的な複合長方形領域(後述の縮退のある場合を含む)の最小分割数を与える公式を導き、最小分割を求めるための従来より効率のよいアルゴリズムを提案する。

まず、この問題を幾何学的な用語を用いて、与えられた複合長方形領域を分割するのに必要な長方形の最小個数を表す公式を与える。次に、そのような分割を見つけるためのアルゴリズムについて述べる。一般の複合長方形領域を分割する問題は、二つの部分問題に分解される。その一つは、与えられた複合長方形領域を、最適に、いくつかの縮退していない部分領域に分解するものである。この部分問題は、与えられた複合長方形領域から導かれた2部グラフの最大マッチング問題に帰着される。もう一つの部分問題は、各部分領域(縮退していない複合長方形領域)を最小個の長方形に分割するものである。この部分問題に対して提案するアルゴリズムの効率性は、本質的に、複合長方形領域の辺を取り扱うデータ構造であるバランス木に依存する。最後に、与えられた複合長方形の頂点の数を $n$ としたとき、前者の問題に対するアルゴリズムの複雑度は $O(n^{5/2})$ となり、後者は $O(n \log n)$ となることを示す。

### 2. 最小分割定理

以下では次のような記号を用いる。複合長方形領域

† Minimum Partitioning of Rectilinear Regions by TATSUO OHTSUKI, MASAO SATO, MASAYOSHI TACHIBANA (Department of Electronics and Communication Engineering, School of Science and Engineering, Waseda University) and SHIROU TORII (Nippon Telegraph and Telephone Public Corporation).

\*\* 早稲田大学理工学部電子通信学科  
\*\*\* 日本電信電話公社

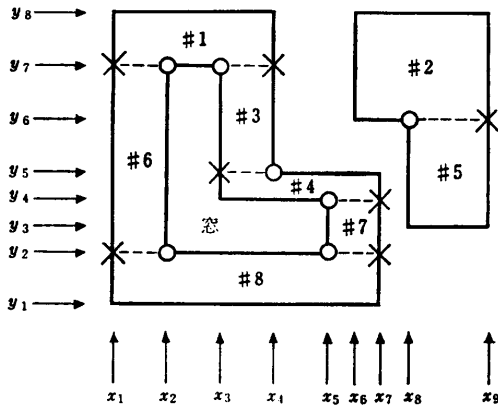


図 1 縮退していない複合長方形領域の最小分割

Fig. 1 Minimum order partitioning of a non-degenerate rectilinear region.  $N=18(\mu=11, \nu=7)$ ,  $w=1$ ,  $k=2$  and  $P(18, 1, 2)=8$

は、図 1 のように窓を含んだり、複数個の連結成分から成っている場合もあるとする。

R: 複合長方形領域

$n$ : R の頂点 (辺) の数

$\mu$ : R の凸点 (内角が  $90^\circ$ ) の数

$\nu$ : R の凹点 (内角が  $270^\circ$ ) の数

$w$ : R が含む窓の数

$k$ : R の連結成分の数

ここで  $n$  は偶数で、かつ

$$\mu + \nu = n \quad (1)$$

という関係が成立することは明らかである。

まず単純な場合として  $w=0$ , かつ  $k=1$  の場合、

$$\mu - \nu = 4 \quad (2)$$

であることは下記のような帰納法によって証明できる。

i)  $n=4$  の場合、 $\mu=4$ ,  $\nu=0$  であるので式(2)が成立することはいうまでもない。

ii)  $n>4$  の場合、 $\nu \geq 1$ , すなわち少なくとも 1 個の凹点を含む。その一つの頂点 A より縦または横方向に線を引けば、R の一つの辺と交差する。この交点を B とすれば R は切断線 AB によって二つの複合長方形領域  $R_1$ ,  $R_2$  に分割される (図 2 参照)。ただし、B が R の頂点と一致する場合には、R の代わりにその頂点をわずかに移動させた複合長方形領域を考えても、凸点および凹点の数は変化しないので、一般性を失うことはない。  $R_1$  の凸点、凹点の数をそれぞれ  $\mu_1$ ,  $\nu_1$ ,  $R_2$  の凸点、凹点の数をそれぞれ  $\mu_2$ ,  $\nu_2$  とすれば、R の凹点 A は、 $R_1$ ,  $R_2$  の片方の凸点となり、(R の頂点でない) 点 B は、 $R_1$ ,  $R_2$  両方の凸点となっているので、

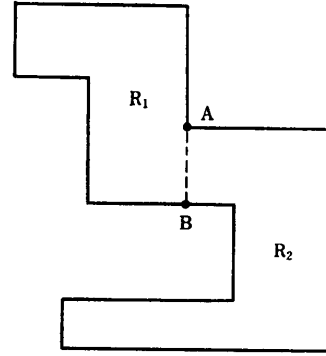


図 2 複合長方形領域の 2 分割

Fig. 2 Bipartitioning of a rectilinear region.

$\mu_1=5$ ,  $\nu_1=1$ ;  $\mu_2=6$ ,  $\nu_2=2$ ;  $\mu=8$ ,  $\nu=4$

R において、

$$\begin{cases} \mu = \mu_1 + \mu_2 - 3 & (3) \\ \nu = \nu_1 + \nu_2 + 1 & (4) \end{cases}$$

という関係が成立する。一方  $R_1$ ,  $R_2$  のおのおのについて式(2)が成立すると仮定すれば、

$$\mu_1 - \nu_1 = \mu_2 - \nu_2 = 4 \quad (5)$$

であるので、これと式(3), (4)より式(2)が得られる。

以上 i), ii) より、 $w=0$  かつ  $k=1$  の場合に式(2)が成立することが示された。このことは、おのおのの連結成分における外側の境界上の凸点 ( $\mu$  個), 凹点 ( $\nu$  個) に関して式(2)が成立することを含んでいる。また、おのおのの窓を構成する境界上の凸点 ( $\mu'$  個), 凹点 ( $\nu'$  個) に関して、凸点、凹点の立場を逆転して考えれば、

$$\mu' - \nu' = -4 \quad (6)$$

という関係も成立することになる。R の境界を構成するすべての閉曲線に関して、式(2)あるいは式(6)の関係を適用し、これを加え合わせれば次の補題が導かれる。

[補題 1]

$$\mu - \nu = 4k - 4w \quad (7)$$

また、この補題と式(1)から次の補題も得られる。

[補題 2]

$$\mu = n/2 - 2w + 2k \quad (8)$$

$$\nu = n/2 + 2w - 2k \quad (9)$$

同じ X または Y 座標上の二つの凹点  $V_1$ ,  $V_2$  間を結ぶ線分が複合長方形領域の領域内である (境界線を含まない) とき、縮退 (degeneracy) しているといい、 $V_1$  と  $V_2$  を結ぶ線分のことを縮退線 (degenerate chord) と呼ぶ。

複合長方形領域Rの分割は、Rを重複しない長方形の和として表すことである。結果として得られた長方形の数を分割数と呼び、 $P(n, w, k)$ で表す。これを用いて次に補題を示す。

【補題3】縮退していない複合長方形領域の任意の分割に対して、

$$P(n, w, k) \geq n/2 + w - k \tag{10}$$

が成り立つ。

(証明) 複合長方形領域Rのある分割において、分割数が $P(n, w, k)$ であったとする。この分割において、領域Rの内部を通る線分を分割線と呼ぶ。ただし、分割線としては、「同一直線上にあるいかなる二つの分割線も端点を共有しない」という意味で極大なものだけを対象とする。たとえば、図3の分割においては線分 $(A_1B_1)$ ,  $(A_2B_2)$ , ...,  $(A_7B_7)$ が対象とする分割線である。分割線の中でおのおのの凹点(図3の○印) $A_1, A_2, \dots, A_7$ を端点とするもの(最低1個はある)を一つずつ選び、それぞれのもう一つの端点(図3の×印)を $B_1, B_2, \dots, B_7$ とすれば、 $\{A_i; i=1 \sim 7\}$ と $\{B_i; i=1 \sim 7\}$ の間には1:1対応がある。ただし、 $B_i; i=1 \sim 7$ はRの境界上または内部の分割線の上にある。ここでおのおのの凹点 $A_i$ は少なくとも1個の長方形の頂点であり、おのおのの分割点 $B_i$ はちょうど2個の長方形の頂点である。また、おのおのの凸点はちょうど1個の長方形の頂点である。一方、Rを構成する長方形の頂点の数の総和は明らかに $4P(n, w, k)$ 個である。以上の事実より下記の方程式を得る。

$$4P(n, w, k) \geq \mu + 3\nu \tag{11}$$

これに式(8), (9)を代入すれば、式(10)が得られる。(証明終)

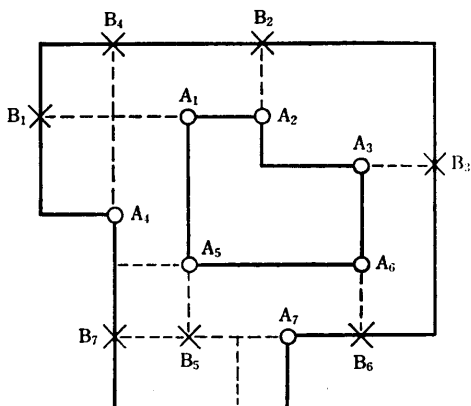


図3 複合長方形領域の最小でない分割  
Fig. 3 A non-optimum partitioning of a rectilinear region.

Rが縮退していないとき、最小分割は簡単に得られる。図1のように、各凹点に対して水平に分割線(点線で示されている)を引けば最小分割が得られる。何となれば、この分割においては、頂点以外の分割線の端点は凹点に対応するもの(×印の点)ばかりであり、各凹点から出ている分割線はちょうど1個である。したがって、式(11)——すなわち式(10)——の等号が成立する。これを補題として記す。

【補題4】縮退していない複合長方形領域に対して、式(10)の等号を満たすような分割が存在する。

補題3と補題4より、次の定理を得る。

【定理1】縮退のない複合長方形領域の最小分割数は、

$$P(n, w, k) = n/2 + w - k \tag{12}$$

となる。

次に、Rが縮退しているような一般的な場合について考察する。もし、Rの二つの縮退線が共通の点をもたないならば、それらは独立しているといい、そうでない場合には、従属しているという。縮退線の定義より、X軸に平行な縮退線どうし、また、X軸に垂直な縮退線どうしは独立である。たとえば、図4において、AとE、BとCなどは独立であり、AとF、BとHなどは従属である。

複合長方形領域Rの縮退線をCとする。CでRを切断したとすると、Rは図5のようにもう一つの複合長方形領域R\*に変換される。ここで、(a)kが1増加するか、(b)wが1減少するか2通りが考えられる。

一方、式(12)より、縮退していない複合長方形領域に対して、

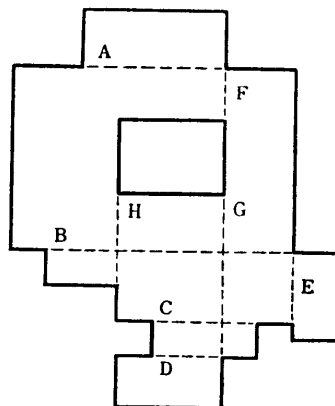


図4 縮退線の集合  
Fig. 4 Set of degenerate chords of a rectilinear region.

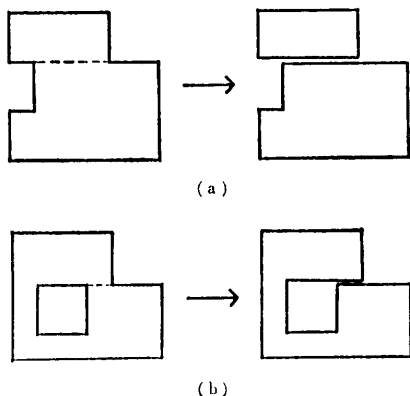


図5 縮退線による分割

Fig. 5 Exploiting a degenerate chord.

$$P(n, w-1, k) = P(n, w, k+1) \\ = P(n, w, k) - 1 \quad (13)$$

という関係が成立することは明らかである。したがって、変換された複合長方形領域  $R^*$  がもはや縮退していなければ、 $R$  の縮退線を利用することによって長方形の数を式 (12) で与えられる値より 1 個少なくでき、しかもこれ以上少なくできないことになる。 $R^*$  が別の縮退線を含む場合には、縮退線がなくなるまで図 5 のような変換を繰り返すことができる。この操作によって利用された縮退線の数だけ  $(w-k)$  の値、すなわち長方形の数が減少する。

縮退している複合長方形領域の分割数は利用された縮退線の数によって異なる。ある縮退線  $C$  が他の縮退線  $C'$  と交差していて、そして  $R$  が  $C$  を利用することによって  $R^*$  に変換されたならば、 $C'$  はもはや  $R^*$  の縮退線ではなくなる。このことは、最小分割するためにはできるだけ多くの独立な縮退線を利用しなければならないことを示している。 $d$  を独立な縮退線の最大数とすると、最小分割における長方形の数は次のように表せる。

【定理 2】 複合長方形領域の最小分割数は、

$$P(n, w, k) = n/2 + w - k - d \quad (14)$$

となる。

### 3. 縮退していない場合の分割アルゴリズム

入力データとして、複合長方形領域の頂点の  $x, y$  座標が右回りの順で与えられるとする。ここで、右回りとは、複合長方形領域  $R$  の境界に沿って進むとき、 $R$  の内部が右側に見える向きをいう。この章では、縮退していない複合長方形領域を最小の長方形に分割す

る効率のよいアルゴリズムを提案する。このアルゴリズムは以下の二つの段階から成る。

**Phase 1.** 与えられた  $R$  の頂点を走査しながら水平な辺の集合を  $y$  座標値の大きい順に整理する。このとき、各水平な辺に関して走査されたときの向きを記憶しておく。この整理された集合のリストを  $Y$ -List と呼ぶことにする。

**Phase 2.** 垂直な辺を表現するために  $X$ -List を用いる。 $X$ -List は最初、空である。Phase 1 で整理された順に  $Y$ -List から一つずつアイテムを取り出し、各水平な辺  $H$  に対し以下の操作を行う。 $H$  の二つの端点  $A, B$  と等しい  $x$  座標値をもつ垂直な辺が  $X$ -List の中にあるか否か (4 通り)、および  $H$  の向き (2 通り) によって図 6 に示すような 8 通りの場合が考えられる。ここで、 $H$  は太線で示されていて、矢印はその辺の向きを表している。 $H$  が  $Y$ -List から取り出されたとき、 $X$ -List はその上部の  $y$  座標が  $y(H)$  より大きく、その下部の  $y$  座標が  $y(H)$  以下であるような垂直な辺を含んでいる。ここで  $y(H)$  は  $H$  の  $y$  座標である。各  $H$  に対する操作は、もし必要ならば (つまり図 6 (i) を除く場合には)、一つか二つの長方形を分離

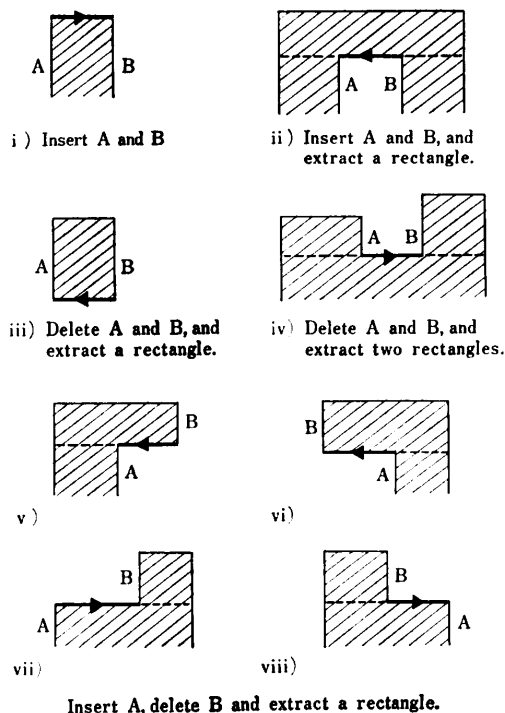


図6 X-List 内の垂直な辺の集合の更新と長方形の分離

Fig. 6 Update of the set of vertical edges in X-List and extraction of rectangles.

し、 $X$ -List の要素を図6のように更新することである。

Phase 2 において  $Y$ -List のすべての辺が走査されたとき、このアルゴリズムは終了する。たとえば、図1において複合長方形領域の中の長方形は付けられた番号の順に分離される。このアルゴリズムでは複合長方形領域は水平な切断線のみによって分割される。水平方向あるいは垂直方向のみによる分割が最小分割するために必要というわけではないが、後述のように計算の効率性という観点から必須のものである。

#### 4. バランス木の利用

前章で述べたアルゴリズムの Phase 2 における  $X$ -List を表現するためにバランス木を用いると処理の効率化が図れる。バランス木の代表的なものとして、AVL 木<sup>6)</sup>と2-3木<sup>6)</sup>がよく知られているが、ここではAVL木を採用した。

AVL 木は、おのおのの節点に対して、左部分木と右部分木の高さの差がただか1であるという平衡構造をもつ2分探索木である。AVL 木(節点数を  $m$  とする)の平衡構造および節点をその  $key$  の値に基づいて中間順\* (inorder) に配置することによって、下記の基本操作に要する処理時間を  $O(\log m)$  で抑えることができる。

**MEMBER** : あるアイテムが AVL 木の節点としてすでに含まれているか否かを判定する。本アルゴリズムでは、あるアイテム(値  $a$  をもつとする)に対して、AVL 木の節点のなかで  $a$  より小さくて最大のもの、あるいは、 $a$  より大きくて最小のものを見つけるという操作も行う。

**INSERT** : 一つのアイテムを AVL 木の節点として挿入し、必要があれば AVL 木という性質を満たすように再配置する。

**DELETE** : AVL 木の節点として含まれている一つのアイテムを除去し、必要があれば AVL 木という性質を満たすように再配置する。

例として、図1の図形を分割することを考える。 $y$  座標が  $y_8$  での処理が終わると AVL 木は図7(a)のように四つの垂直な辺が挿入される。ここで、かっこ内の値は各辺の最大の  $y$  座標値である。次に  $Y$ -List か

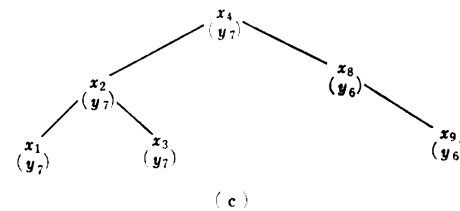
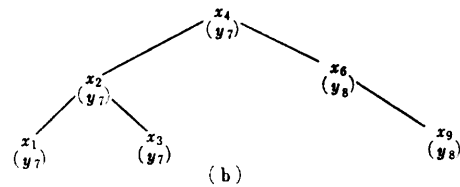
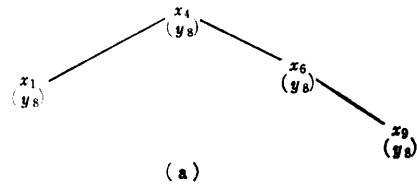


図7 AVL木の更新

Fig. 7 Updating AVL tree.

ら取り出された水平な辺  $H$  を考える。AVL 木のなかには、 $H$  の二つの端点の  $x$  座標  $x_2, x_3$  のどちらとも等しい  $x$  座標をもつ垂直な辺が存在せず、 $H$  の向きが右から左に向いているので、 $H$  が図6(ii)の型に相当することがわかる。そこで、 $x_2, x_3$  を AVL 木に挿入し、図1の#1の長方形を分離するとともに、AVL 木のなかの垂直な辺  $x_1, x_4$  の最大の  $y$  座標値を  $y_7$  に更新する。こうすることによって AVL 木は図7(b)のように更新される。さらに次に  $Y$ -List から取り出された水平な辺は図6(vi)の型に相当するので、AVL 木は図7(c)のように更新され、図1の#2の長方形が分離される。

このような操作を繰り返してゆくと図1のような分割が得られる。

#### 5. 一般の場合の分割アルゴリズム

この章で考察する問題は、縮退している複合長方形領域をいくつかの縮退していない部分領域に分解し、そのおのおのの最小分割を求めれば全体の最小分割となるような縮退線の選択法についてである。前述のように、このためには  $R$  の独立な縮退線の最大集合を求めることが必要である。この問題は以下のようにグラ

\* 2分探索木の各節点において、  
 [左部分木に属する節点の  $key$  の値]  
 $\leq$  [根の  $key$  の値]  
 $\leq$  [右部分木に属する節点の  $key$  の値]<sup>10)</sup>  
 という性質を満たす場合、節点が中間順に配置されているという。

フ理論の用語で表すことができる。Rの各縮退線を節点で表し、もし2本の縮退線が交差するならば対応する2節点間を枝で結ぶ。このようにして2部グラフ  $G(H \cup V, E)$  を得る。ここで、Hは水平、Vは垂直な縮退線に対応する節点の集合である。

あるグラフのすべての枝を被覆する節点の集合を節点被覆 (vertex cover) という。節点被覆のなかで要素数が最小なものの節点数を  $\alpha$  とする。もし節点あるいは枝の集合のどの二つの要素も隣接しなければ、それらの集合は独立 (independent) であるという。独立な節点 (枝) 集合のなかで要素数が最大なものの節点 (枝) 数を  $\beta(\gamma)$  とする。  $\gamma$  個の独立な枝の集合を最大マッチング (maximum matching) という。これらの用語に関して、以下の定理が知られている<sup>7)</sup>。

[定理3] (Gallai)

$$\alpha + \beta = n \quad (15)$$

ここで、  $n$  は節点数である。

[定理4] (König) 2部グラフならば、

$$\alpha = \gamma \quad (16)$$

が成り立つ。

以上のことより、グラフ  $G(H \cup V, E)$  の最大独立節点集合が、(縮退している) 複合長方形領域の求められるべき分解に対応していることは明らかである。グラフ  $G$  の最大マッチングが求まれば、最大独立節点集合も容易に求められる<sup>8)</sup>。また、2部グラフの最大マッチングを求める効率のよいアルゴリズムは文献<sup>9)</sup> に与えられている。

たとえば、図8は図4の複合長方形領域から得られた2部グラフである。最大マッチングは太線で示され、これから得られる最大独立節点集合  $\{A, C, D, E, H\}$  は丸印が付いている。これに対応する最小分割は図9(a)であり、また、  $\{A, E, G, H\}$  を利用した最小でない分割が(b)である。

2部グラフを利用するという着想は文献<sup>4)</sup> に示され

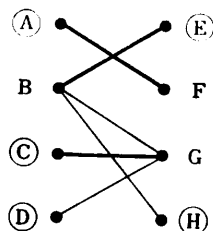


図8 図4における縮退線の交差を表す2部グラフ  
Fig. 8 Bipartite graph representing intersection of degenerate chords in Fig. 4.

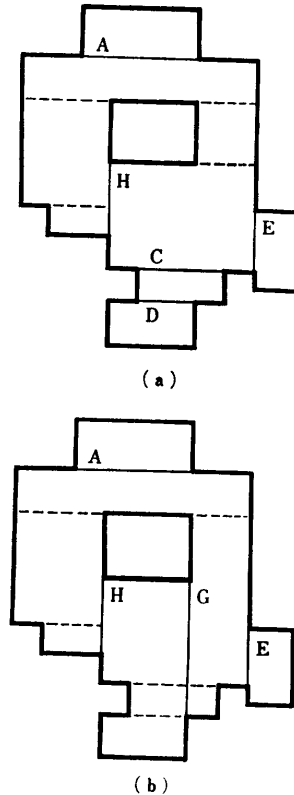


図9 最小分割(a)と最小でない分割(b)

Fig. 9 Optimum partitioning (a) and non-optimum partitioning (b) of a rectilinear region.

ているが、複合長方形領域に窓がない場合だけを扱っている。また、計算時間の解析はなされていない。

## 6. 計算複雑度の解析

最初に、縮退していない複合長方形領域を分割するアルゴリズムの解析を行う。最悪の場合には  $O(n^2)$  の手間を必要とした従来のアルゴリズム<sup>2),3)</sup> に比べて、前述のアルゴリズムは  $O(n \log n)$  の手間で済む。Phase 1 はよく知られている整列方法—たとえば、併合法 (merge sort) —を用いれば  $O(n \log n)$  の手間で実行できる。Phase 2 では、4章で述べたように AVL 木を X-List のために用いる。AVL 木の節点の数はただか  $n/2$  (垂直な辺の数) であり、挿入・削除はそれぞれ  $n/2$  (平行な辺の数) 回であるので、総合の計算時間は  $O(n \log n)$  で抑えられる。

次に、縮退している複合長方形領域の取扱いに関する計算時間の解析を行う。すべての縮退線はただか  $O(n^2)$  の手間で発見できる。あとは、いかに効率よく2部グラフの最大マッチングを求められるかである。著者が知る限りでは、最も効率のよいアルゴリズムは

「最大フローの手法」によるもの<sup>9)</sup>で、 $O(n^{5/2})$ の手間で最大マッチングが求められる。したがって、2番目のアルゴリズムは $O(n^{5/2})$ の手間で実行できる。

これらのアルゴリズムの計算複雑度は最悪の場合について解析されている。実際には、各段階において、AVL木を含む要素数はきわめて少ないのがふつうであるので、1番目のアルゴリズムはほとんど線形時間で実行できると考えてよい。さらに、2番目のアルゴリズムはふつう、前述の評価よりもはるかに速く実行できる。その理由は、縮退線の数はたいてい $n$ よりかなり少なく、また、2部グラフが非連結になりやすいからである。

## 7. あとがき

本論文では、複合長方形領域の最小分割数を与える公式を導き、従来よりも効率のよい分割アルゴリズムを提案した。最後に、さらに一般的な問題を取り扱うために、残された問題を挙げる。

- (1) 多角形が斜めの辺を含む場合
- (2) 分割の際に長方形の重複を許した場合

## 参 考 文 献

- 1) Wiesel, M. and Mlynski, D. A. : An Efficient Channel Model for Building Block LSI, *IEEE Proc. ISCAS*, pp. 118-121 (1981).
- 2) 小山田哲治 : LSI アートワークパタンの最小分割, *情報処理*, Vol. 16, No. 7, pp. 576-580 (1975).
- 3) 川西 宏, 西出 雅, 増田猛雄 : LSI マスクパターンの図形処理の一算法, *電子通信*, Vol. J 63-A, No. 9, pp. 611-618 (1980).
- 4) Ferrari, L., Sankar, P.V. and Sklansky, J. : Minimal Rectangular Partitions of Digitized Blobs, *IEEE Pattern Recognition Conf.*, pp. 1040-1043 (1980).
- 5) Wirth, N. : *Algorithm + Data Structure = Program*, pp. 215-226, Prentice-Hall, Englewood Cliffs (1976).
- 6) Aho, A.V., Hopcroft, J. and Ullman, J. : *The Design and Analysis of Computer Algorithms*, pp. 146-152, Addison-Wesley, Reading (1974).
- 7) Harary, F. : *Graph Theory*, pp. 94-96, Addison-Wesley, Reading (1969).
- 8) Even, S. : *Graph Algorithms*, pp. 135-138, Computer Science Press, Inc., Potomac (1979).
- 9) Hopcroft, J. and Karp, R. : An  $n^{5/2}$  Algorithm for Maximum Matchings in Bipartite Graphs, *SIAM Comput.*, Vol. 2, No. 4, pp. 225-231 (1973).
- 10) Aho, A.V., Hopcroft, J. and Ullman, J. : *The Design and Analysis of Computer Algorithms*, pp. 52-55, Addison-Wesley, Reading (1974).

(昭和58年1月5日受付)  
(昭和58年3月11日採録)