

平面グラフで長さの総和最小な非交差道を求めるアルゴリズム

高橋 淳也 鈴木 均 西関 隆夫

東北大学 工学部

仙台市青葉区荒巻字青葉 東北大学工学部情報工学科

あらかし G は無向平面グラフであり, G の辺の長さは非負実数であるとする. 本文では G の 2 つの面の周上に k 組の端子対が指定されたときに, それぞれの端子対を結び, 長さの総和が最小な“非交差道”を求めるアルゴリズムを与える. なお, 非交差道とは点や辺を共有するかもしれないが, 互いに平面上で交差はしていない道のことである. そのアルゴリズムの計算時間は $O(T(G) \log k)$ である. ここで $T(G)$ は平面グラフ G で 1 点から全点間の最短路を求めるのに要する時間である.

An Algorithm for Finding Non-Crossing Paths with Minimum Total Length in Planar Graphs

Jun-ya TAKAHASHI Hitoshi SUZUKI Takao NISHIZEKI

Tohoku University

Dept. of Information Engineering, Tohoku University, Aramaki Aza Aoba, Aoba-ku, Sendai-shi, 980 Japan

Abstract Let G be an undirected plane graph with non-negative edge length, and suppose that k terminal pairs are located on two specified face boundaries. This paper presents an algorithm for finding k “non-crossing paths” in G , each connecting a terminal pair, whose total length is minimum. Here “non-crossing paths” may share common vertices or edges but do not cross each other in the plane. The algorithm runs in time $O(T(G) \log k)$, where $T(G)$ is the time required for finding shortest paths from a single vertex to all other vertices in a plane graph G .

1. まえがき

平面に埋め込まれているグラフ、即ち平面グラフで、指定された k 個の端子対間の互いに点素で長さの総和が最小な k 本の道を求める問題は VLSI の一層配線問題等に応用可能である。しかし、この問題は NP-完全であるので^{[1][2]}、効率のよいアルゴリズムはありそうにない。ここで、グラフの辺は VLSI の配線領域に対応している。この一層配線問題において、1つの配線領域に複数の配線を通すことができるものと仮定すれば、長さの総和が最小である“非交差道”を求める問題に帰着させることができる。ここで、“非交差道”とは点や辺を共有するかもしれないが互いに平面上で交差はしていない道のことである。非交差道を求める問題に関しては、端子が置かれる面の個数に制約がある場合に対しては効率のよいアルゴリズムがあることが期待される。このような背景から、本文では、平面グラフ上で、 k 個の端子対がグラフの指定された 2 つの面の周上にもみ存在する場合に、長さの総和最小な k 本の非交差道を求める多項式時間アルゴリズムを与える。このアルゴリズムは、例えば、VLSI 配線の最終段階に現れる、チップの外周におかれたパッドと内部ブロック周囲にあるピン間を結ぶ一層配線問題に適用できる(図1参照)。2章において問題と用語の定義を行い、3、4章で問題に対する多項式時間アルゴリズムを与え、その計算時間が $O(T(G) \log k)$ であることを示す。ここで、 $T(G)$ は平面グラフ G で 1 点から全点の最短路を求めるのに要する時間である。最後に、5章で今後の課題を示す。なお、平面グラフの 2 つの面の周上に端子が指定されているとき、(長さの総和最小とは限らない) 点素な道を求める $O(n \log n)$ 時間のアルゴリズムが知られている^{[3][4]}。ここで、 n は G の点数である。

2. 準備

本章では、用語と問題の定義を示す。点集合 V 、辺集合 E を持つグラフ G を $G = (V, E)$ と書く。本文では G は 2-連結無向平面グラフであり、 G の辺の長さは非負であるとする。 G は平面 R^2 上に埋め込まれているとする。 G の R^2 上の像を $Image(G) \subset R^2$ と書く。 $R^2 - Image(G)$ の連結成分を G の面と呼ぶ。道で連結したい 2 点 s_i と t_i からなる対 (s_i, t_i) を端子対と呼ぶ。本文では、端子対の個数を k と表す。また、端子はすべて G の指定された 2 つの面 f_1, f_2 の周上にあるものとする。

G の道 P の 2 点 v, w に対して、 v から w へ行く P の部分道を $P[v; w]$ と書く。 G の 2 点 v_1 と v_r の間の道を $P = v_1, v_2, \dots, v_r$ としたとき、グラフ G を道 P に沿って切れ目を入れるという処理を次のように定義する。

まず P の各点 $v_j, j = 1, 2, \dots, r$ を二つの点 u_j と w_j で置き換える。 P 上の各辺 (v_j, v_{j+1}) を 2 本の平行な辺 (u_j, u_{j+1}) と (w_j, w_{j+1}) で置き換える。道 P を v_1 から v_r へ進むときに点 v_j の左側に接続している辺を u_j に、右側に接続している辺を w_j に接続する。こうして得られたグラフを P に関する G の切れ目グラフと呼ぶ。

本文では次の非交差道問題を解くアルゴリズムを与える。

非交差道問題

平面グラフ G 、および G の 2 つの面 f_1, f_2 の周上にある点からなる k 個の端子対 $(s_i, t_i), i = 1, 2, \dots, k$ 、が与えられたとき、各端子対間を結ぶ、長さの総和が最小な k 本の非交差道を求める(図2参照)。

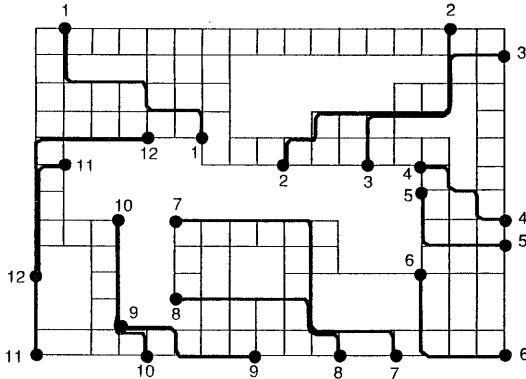


図1. 非交差道問題の VLSI 配線問題への適用例

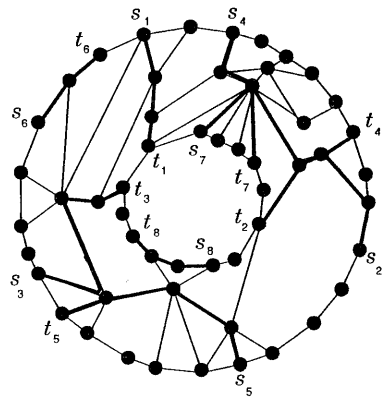


図2. 平面グラフの非交差道の例

まず第3章で 1 つの面の周上にだけ端子がある場合に対するアルゴリズムを与え、次に第4章で 2 つの面の周上に端子がある場合に対するアルゴリズムを与える。なお、 k 本の道を列挙する代わりに、 k 本の道の和になっている G の林 F を出力するものとする。無論、林 F の $(s_i, t_i), 1 \leq i \leq k$ 、間の唯一の道は非交差道問題の解である。 k 本の道の長さの総和は、林 F 上の最親共通先祖問題^[5]を解くことにより、 $O(n)$ 時間で求まる^[6]。ここで、 n は G の点数である。

3. 1 つの面の周上に端子がある場合.

本章では、1 つの面 f の周上には端子が無い場合に対して非交差道問題を解くアルゴリズムを与える。一般性を失うことなく f は外面であるとしてよい。端子対の集合を S と書く。端子対の配置に関して次のような場合分けを考える。

場合1: 必要ならば端子 s_i と t_i を入れ換えたり、端子対番号を付け換えたりしたときに、端子 $s_1, t_1, s_2, t_2, \dots, s_k, t_k$ がこの順序で面 f の周上で時計回りに現れる場合。

場合2: その他の場合。

まず場合 1 に対するアルゴリズム PATH1(G, S) を与えよう。PATH1 はまずグラフ G を k 個の部分グラフ G_1, G_2, \dots, G_k に分割し、各 G_i には端子 s_i と t_i が含まれるようにする。次に各 G_i で s_i と t_i 間の最短路を求め、求まった k 本の道の和から得られる林を出力する。

procedure PATH1(G, S);

1. begin
2. s_1 と s_i , $2 \leq i \leq k$, を結ぶ最短路を Q_i と表す;
3. $T := (\{s_1\}, \emptyset)$; { T は切れ目を入れる道の和からなる木}
4. for $i = 1$ to $k-1$ do
5. begin
6. 道 Q_{i+1} を s_{i+1} から s_1 までたどったときに現れる木 T との最初の交点を u とし, T を u から s_i へ行く道 $T[u; s_i]$, 外面 f の周上を時計回りに s_i から s_{i+1} へ行く道, および道 $Q_{i+1}[s_{i+1}; u]$ の 3 本の道から成る閉路の内部にある G の部分グラフを G_i とする;
7. $T := T + Q_{i+1}[s_{i+1}; u]$
8. end;
9. 道 Q_k と外面 f の周上を時計回りに s_k から s_1 へ行く道から成る閉路の内部にある G の部分グラフを G_k とする;
10. $F := (\emptyset, \emptyset)$; { F は求める林}
11. for $i = 1$ to k do
12. begin
13. グラフ G_i において s_i と t_i の間の最短路 P_i を求める;
14. if P_i は F と交差する
15. then
16. begin
17. 道 P_i を s_i から t_i まで進んだときに最初に現れる F の点を v とし, 最後に現れる F の点を w とする;
18. $F := F + P_i[s_i; v] + P_i[w; t_i]$
19. end
20. else $F := F + P_i$
21. end;
22. F を解とする
23. end;

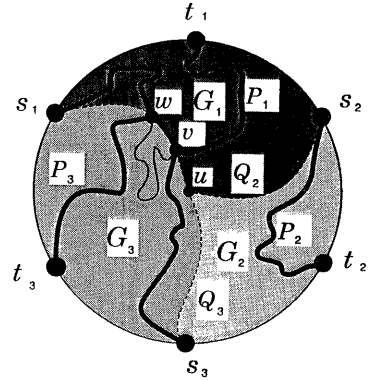


図 3. PATH1 の説明図

図 3 に PATH1 の説明図を示す。同図で、道 Q_i は点線で、道 P_i は実線で、林 F は太線で描かれていて、部分グラフ G_1, G_2, G_3 は色分けされている。上に示した手続き PATH1 が正しく長さ最小の非交差道を求めることは次の補題から明かであろう。

補題 1. PATH1 で求めた G の部分グラフを G_i とすると、グラフ G における端子 s_i と t_i の間の最短路の 1 つはグラフ G_i に存在する。

次に PATH1 の計算時間の解析を行う。行 2 と行 13 以外は明らかに $O(n)$ 時間で実行できる。また、行 2 の s_1 と全点間の最短路を求める部分にかかる時間は $O(T(G))$ である。したがって、この手続きの計算時間が $O(T(G))$ 時間であることを示すには、行 13 の処理がループの繰り返しを考慮しても $O(T(G))$ 時間で実行できることを示せばよい。行 13 では、 k 本の最短路を求めるが、各最短路は木 T によって境界付けられている領域で求められる。したがって、 T の辺は G_1, G_2, \dots, G_k の高々 2 個にしか現れず、それ以外の G の辺は高々 1 個にしか現れない。よって行 13 は $O(T(G))$ 時間で実行できる。以上より PATH1 は $O(T(G))$ 時間で実行できる。

次に PATH1 を利用して場合 2 に対するアルゴリズムを与えよう。 f の周上を時計回りにたどったときに、 v_1, v_2, \dots, v_b の順で f の点が現れるものとする。また、一般性を失うことなく $s_1 = v_1$ ($= v_1$) から面 f の周上を反時計回りに t_1 まで行く道に、 s_1, t_1 以外の端子が現れないとしてよい。さらに、各端子対 (s_i, t_i) に関して、 f の周を v_1 から時計回りに進んだときに最初に s_i が現れ、次に t_i が現れるものとしてよい。さらに、 f の周上を v_1 から v_b まで時計回りにまわったとき、 s_1, s_2, \dots, s_k がこの順に現れるものとしてよい (図 4(a) 参照)。 f の周上の各点 v に対して、 $index(v)$ を v の添字を示すものとする。即ち、 $v = v_i$ ならば $index(v) = i$ である。この記法を用いて端子対の親子関係を定義する。 $index(s_i) < index(s_j) < index(t_j) < index(t_i)$ が成立しているとき、 (s_i, t_i) は (s_j, t_j) の先祖であるといい、 (s_j, t_j) は (s_i, t_i) の子孫であるという。 $index(s_i) < index(s_j) < index(t_i) < index(t_j)$ のときは、明らかに非交差道は存在しない。 (s_i, t_i) の先祖の内、最大の端子対番号を持つ先祖を (s_i, t_i) の親と呼び、 (s_i, t_i) を親の子と呼ぶ。 S の各端子対 $p_i = (s_i, t_i)$ を 1 つの節点 p_i として持ち、端子対間の親子関係から誘導される木を T_g と書く。親を持たない端子対 (s_1, t_1) に対応する節点 p_1 を木 T_g の根という。端子対 (s_i, t_i) に対応する T_g の節点 p_i の深さに 1 を加えた数を端子対 (s_i, t_i) の世代という (図 4(b) 参照)。

場合 2 に対するアルゴリズム PATH2 のアイデアは 2 つある。まず最初のアイデアは、同じ世代の端子対の集合は場合 1 の性質を満たすので、同じ世代の端子対間の非交差道問題の解は PATH1 によって求められるということである。また、ある世代に含まれる端子対に関して PATH1 を適用して得られる道に沿って G に切れ目を入れて G を分割したとき、各連結成分に存在する端子対間の G における最短路の 1 つはその連結成分中に存在する。したがって世代 1 から順に世代毎に PATH1 を実行して非交差道を探ればよいが、単純にインプレメントすると $O(kT(G))$ 時間か

かってしまう。そこで、まず真中の世代の端子対を結ぶ非交差道を求め、それらの道に沿って G に切れ目を入れて得られるグラフの連結成分毎に上半分と下半分の世代の非交差道を求めるという分割統治法を用いる。これが2番目のアイデアである。図5に例を示す。この図では、真中の世代である世代3の端子対を結ぶ林 F を求め(図5(a)参照)、 F に沿って G に切れ目を入れて G を分割した様子を示している(図5(b)参照)。同図では、上半分の世代の端子対が存在する領域を濃い色で、下半分の世代の端子対が存在する領域を薄い色で描いている。このようにして全体の計算時間が $O(T(G) \log k)$ であるようなアルゴリズムが構成できる。そのアルゴリズムを示す前に、いくつか定義を与える。

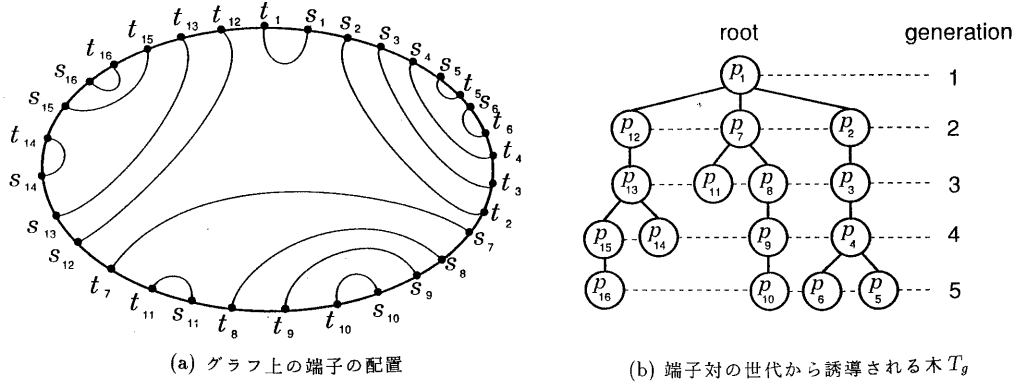


図4. 端子対の世代付け

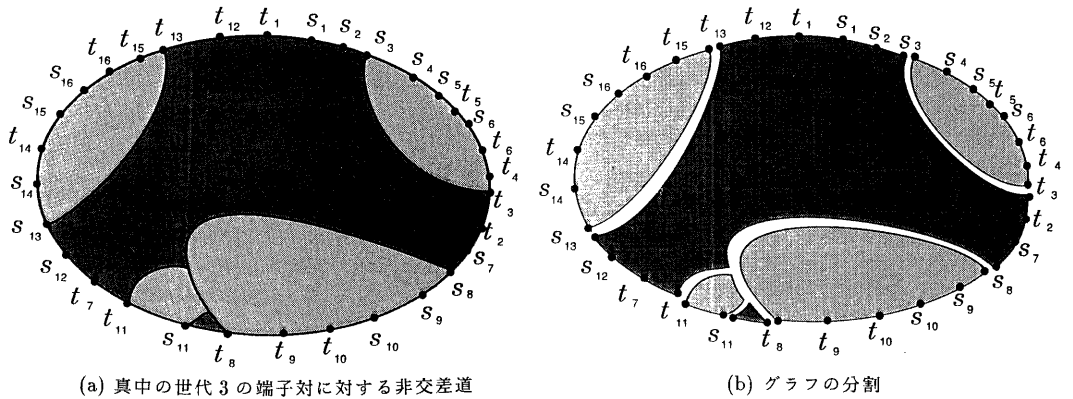


図5. REDUCEの説明図

端子対 (s_i, t_i) を結ぶ道 P_i の内部とは、道 P_i を s_i から t_i へ行き、さらに t_i から外周上を反時計回りに s_i へ行く閉路の内部領域であると定義する。また端子対を結ぶ道の集合 \mathcal{P} の内部とは、 \mathcal{P} に含まれる道の内部の和と定義する。また、端子対 (s_i, t_i) を結ぶ道 P_i の外部とは、道 P_i を s_i から t_i へ行き、さらに t_i から外周上を時計回りに s_i へ行く閉路の内部領域であると定義する。また端子対を結ぶ道の集合 \mathcal{P} の外部とは、 \mathcal{P} に含まれる道の外部の和と定義する。

procedure PATH2(G, S);

1. begin
2. 端子対の世代を計算し、世代の最大値を g とする;
3. $F := (\emptyset, \emptyset)$;
4. REDUCE($G, [1, g], F$);
5. 行4の出力 F を解とする
6. end;

procedure REDUCE($G, [l, h], F$);

1. begin
2. if $l = h$ { 1つの世代しかない }
3. then
4. begin
5. 世代 l の端子対からなる集合を S^l とする;
6. 手続き PATH1(G, S^l) を実行して、求まった道の集合を \mathcal{P} とする;
7. for 各道 $P_i \in \mathcal{P}$ do

```

8.      if  $P_i$  が林  $F$  と共有点を持たない then  $F := F + P_i$ 
9.      else
10.     begin
11.      $P_i$  を始点  $s_i$  から終点  $t_i$  までたどった時に最初に現れる林  $F$  の点を  $v$ , 最後に現れる  $F$  の点を
12.      $w$  とする;
13.      $F := F + P_i[s_i; v] + P_i[w; t_i]$ 
14.     end
15.     else
16.     begin
17.      $m := \lfloor (l+h)/2 \rfloor$ ;
18.     世代  $m$  の端子対の集合を  $S^m$  とする;
19.     手続き PATH1( $G, S^m$ ) を実行して, 求まった道の集合を  $\mathcal{P}$  とする;
20.     for 各道  $P_m \in \mathcal{P}$  do
21.     if  $P_m$  が林  $F$  と共有点を持たない then  $F := F + P_m$ 
22.     else
23.     begin
24.      $P_m$  を始点  $s_m$  から終点  $t_m$  までたどった時に最初に現れる林  $F$  の点を  $v$ , 最後に現れる  $F$  の点を
25.      $w$  とする;
26.      $F := F + P_m[s_m; v] + P_m[w; t_m]$ 
27.     end;
28.      $G$  の  $in(\mathcal{P})$  内の部分グラフを  $G_{in}$  とし,  $out(\mathcal{P})$  内の部分グラフを  $G_{out}$  とする;
29.     REDUCE( $G_{in}, [m+1, h], F$ );
30.     REDUCE( $G_{out}, [l, m-1], F$ );
31.     end;
end;

```

手続き REDUCE 以外の部分にかかる PATH2 の計算時間は明らかに $O(n)$ 時間である。次に REDUCE の計算時間を調べよう。REDUCE は、端子対の世代に関する分割統治法を用いている。REDUCE が呼び出されたとき、真中の世代の端子対を結ぶ非交差道が手続き PATH1 を用いて求められる。PATH1 にかかる時間は $O(T(G))$ である。次に、求まった真中の世代の道に沿ってグラフに切れ目を入れ、問題を上半分の世代と下半分の世代に分割し、分割された2つの問題を再帰的に REDUCE によって解く。このとき、再帰呼び出しの深さは高々 $\log k$ である。同じ深さの再帰呼び出しで各部分グラフに対して実行される PATH1 にかかる時間の合計が $O(T(G))$ であることは、 G の各辺がそれらの部分グラフの2つにしか現れないことからわかる。したがって REDUCE の全体の計算時間は $O(T(G) \log k)$ 時間である。以上より場合2の非交差道問題の解は $O(T(G) \log k)$ 時間で求まることがわかる。

4. 2つの面の周上に端子対がある場合

本章では、2つの面 f_1 と f_2 の周上に端子対がある場合のアルゴリズムを与える。

面 f_1 の周上にある点集合を $V(f_1)$ とし、面 f_2 の周上にある点集合を $V(f_2)$ とする。 f_1 の周上と f_2 の周上の両方に端子対が存在する端子対 (s_i, t_i) について、一般性を失うことなく $s_i \in V(f_1)$, $t_i \in V(f_2)$ とする。 $s_i \in V(f_1)$, $t_i \in V(f_2)$ である端子対 (s_i, t_i) の集合を S_{12} と書き、 $s_i, t_i \in V(f_1)$ である端子対 (s_i, t_i) の集合を S_1 と書き、 $s_i, t_i \in V(f_2)$ である端子対 (s_i, t_i) の集合を S_2 と書く。また、必要ならば端子対の番号付けを変えて、 $1 \leq i \leq l$ ならば $(s_i, t_i) \in S_{12}$ であり、 $l+1 \leq i \leq k$ ならば $(s_i, t_i) \in S_1$ または $(s_i, t_i) \in S_2$ であるとする。

以下では、記述を明確にするため、原点 O を中心とする半径1の円 C_1 と半径 $1/2$ の円 C_2 に囲まれた平面領域 Σ 内に G が埋め込まれているとする。また、 $(s_i, t_i) \in S_{12}$ ならば、 $Image(s_i) = (\cos(\frac{2i\pi}{l}), \sin(\frac{2i\pi}{l}))$, $Image(t_i) = (\frac{1}{2} \cos(\frac{2i\pi}{l}), \frac{1}{2} \sin(\frac{2i\pi}{l}))$ とし、 $Image(G) \cap (C_1 \cup C_2) = \{Image(s_i), Image(t_i) \mid 1 \leq i \leq l\}$ として一般性を失わない。

s_i を始点とし t_i を終点とする Σ 内の道を P とし、 P 上の2点 v, w は $v \in P[s_i; w]$ であるとする。点 X が道 $P[v; w]$ を v から w までたどったときに動径 OX が回転する回転角を θ とする。ただし反時計回りを正とする。 $|\theta| > 2\pi$ のこともある。このとき道 $P[v; w]$ の回転角 $\theta(P[v; w])$ を $\theta(P[v; w]) = \frac{1}{2\pi}\theta$ と定義する。非交差道問題の解を $P_1, P_2, \dots, P_l, \dots, P_k$ とすると、明らかに $\theta(P_1) = \theta(P_2) = \dots = \theta(P_l)$ が成り立っている。

次の補題が成立する。

補題2. $1 \leq i, j \leq l$ に対して、 s_i と t_j の間の任意の最短路を P_{si} とし、 s_j と t_j の間の任意の(最短路とは限らない)道を P_j とする。 $|\theta(P_{si}) - \theta(P_j)| \geq 2l$ であるとき、 $|\theta(P_{si}) - \theta(P'_j)| = l$ であり、かつ P_j より長くない道 P'_j が存在する。しかも $\theta(P_{si}) > \theta(P_j)$ ならば $\theta(P_{si}) > \theta(P'_j)$ であり、 $\theta(P_{si}) < \theta(P_j)$ ならば $\theta(P_{si}) < \theta(P'_j)$ である。

補題2において、もし P_j が最短路であるならば、 P'_j も最短路のはずであるから、 P_{si} に対して回転角の差が $2l$ 以上の最短路が存在する端子対には回転角の差の絶対値が丁度 l であるような最短路が必ず存在することがわかる。

補題3. 互いに回転角が等しい l 本の道を P_1, P_2, \dots, P_l としたとき、 P_1, P_2, \dots, P_l に交差があっても、道の長さの総和が同じで、互いに交差しない l 本の道が存在する。

補題4. 端子対 (s_1, t_1) 間の任意の最短路を P_{s1} とし、非交差道問題の解である k 本の道を P_1, P_2, \dots, P_k とする。 $|\theta(P_{s1}) - \theta(P_1)| \geq 2l$ ならば、 $|\theta(P_{s1}) - \theta(P'_1)| = l$ なる非交差道問題の解 P'_1, P'_2, \dots, P'_k が存在する。

G 上の s_1 と t_1 の間の最短路を $P_{s_1}^0$ とし, $P_{s_1}^0$ に関する G の切れ目グラフを G'_0 と書く, G'_0 上には s_1 を置き換えた 2 点 v, v' と, t_1 を置き換えた 2 点 w, w' が存在する. v, v', w および w' は G'_0 で, 同一面 f の周上に存在する. ここで, f を v から時計回りに進んだとき, v, w, w', v' がこの順で現れるものとする. グラフ G'_0 における v と w' の間の最短路および v' と w の間の最短路に対応する G の 2 本の道を, それぞれ $P_{s_1}^+$ および $P_{s_1}^-$ と書く. $\theta(P_{s_1}^+) - \theta(P_{s_1}^0) = l$. $\theta(P_{s_1}^-) - \theta(P_{s_1}^0) = -l$ が成立する. 図 6 に G^0 の $P_{s_1}^+$ と $P_{s_1}^-$ の例が示されている. ここで $P_{s_1}^+$ は実線で, $P_{s_1}^-$ は破線で描かれている. 補題 2, 3, 4 から次の補題がいえる.

補題 5. s_1 と t_1 を結ぶ道が $P_{s_1}^0, P_{s_1}^+, P_{s_1}^-$ のいずれかであるような非交差道問題の最適解が存在する.

以上の性質により, 非交差道問題のアルゴリズムが次のように構成できる.

ALGORITHM PATH

入力: G と端子対の集合 $S_1 \cup S_2 \cup S_{12}$

出力: 互いに交差しない長さの総和最小な k 本の道を含む林

1. begin
2. G において s_1 と t_1 の間の最短路 $P_{s_1}^0$ を G で求める;
3. $P_{s_1}^0$ に関する G の切れ目グラフ G'_0 を構成し, 道 $P_{s_1}^+, P_{s_1}^-$ を求める;
4. $P_{s_1}^+$ に関する G の切れ目グラフ G'_1 と, $P_{s_1}^-$ に関する G の切れ目グラフ G'_2 を構成する;
5. for $i = 0$ to 2 do
6. begin
7. 手続き PATH2 を G'_i 上で実行する;
8. 手続き PATH2 で求めた G'_i の林に対応する林を F_i とする
9. end;
10. 林 F_0, F_1, F_2 のうち, その上の k 本の非交差道の長さの総和が最小な林を解とする
11. end;

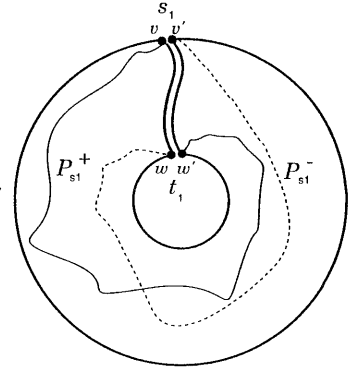


図 6. 切れ目グラフ G'_0 の例

アルゴリズム PATH の実行時間は以下の通りである. 2 行目は最短路を求める処理であるから, $O(T(G))$ 時間で実行できる. 5 行目から 10 行目のループは 3 回繰り返される. また, 章 3 で示したように, 6 行目の手続き PATH2 は $O(T(G) \log k)$ 時間で終了する. 他の行に関しては $O(n)$ 時間で実行できるから, 全体の実行時間は $O(T(G) \log k)$ 時間である.

以上より次の定理が成立する.

定理 1. k 個の端子対が 2 つの面の周上にある平面グラフ G で, 長さの総和最小な非交差道は $O(T(G) \log k)$ 時間で求まる.

なお, Dijkstra 法を用いれば $T(G) = n \log n$ である^{[7],[8]}. また, Frederickson の方法を用いれば $O(n \log n)$ 時間かかる前処理を施した時に, $T(G) = n$ である^[9]. したがって $O(n(\log n + \log k))$ 時間で長さの総和最小な非交差道が求まる.

5. むすび

本文では, k 個の端子対が平面グラフの指定された 2 つの面の周上にのみ存在する場合に, 長さの総和最小な k 本の非交差道を求める多項式時間アルゴリズムを与え, その計算時間が $O(T(G) \log k)$ であることを示した. 今後の課題として, アルゴリズムをより高速化すること, 3 つ以上の面の周上に端子があるより一般的な場合に対するアルゴリズムを求めること等があげられる. なお, 平面グラフではなく, 2 次元平面上に端子が与えられた場合に長さの総和最小の非交差道を求めるアルゴリズムについては次の機会に報告する.

参考文献

- [1] J. F. Lynch, "The equivalence of theorem proving and the interconnection problem," ACM SIGDA, the Netherlands (1982).
- [2] M. R. Kramer and J. van Leewen, "Wire-rooting is NP-complete," Report Mo.RUU-CS-82-4, Department of Computer Science, University of Utrecht, Utrecht, the Netherlands (1982).
- [3] 鈴木均, 赤間長浩, 西関隆夫, "平面グラフで林を求めるアルゴリズム - 指定された二つの面の両方にまたがる端子がある場合 -," 信学論 A, Vol.J71-A, No.10, pp.1897-1905 (1988).
- [4] 鈴木均, 赤間長浩, 西関隆夫, "平面グラフで林を求めるアルゴリズム - 各ネットの端子が指定された二つの面の片方にある場合 -," 信学論 A, Vol.J71-A, No.12, pp.2163-2171 (1988).
- [5] H. N. Gabow and R. E. Tarjan, "A linear-time algorithm for a special case of disjoint set union," Journal of Computer and System Sciences, 30, pp.209-221(1985).
- [6] 鈴木均, 赤間長浩, 西関隆夫, "平面グラフで内素な道を求めるアルゴリズム," 信学論 A, Vol.J71-A, No.10, pp.1906-1916 (1988).
- [7] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, "The Design and Analysis of Computer Algorithms," Addison-Wesley, Reading, MA, 1974.
- [8] R. E. Tarjan, "Data Structures and Network Algorithms," SIAM, Philadelphia, PA., 1983.
- [9] G. N. Frederickson, "Fast algorithms for shortest path in planar graphs, with applications," SIAM J. Comput., 16, pp.1004-1022 (1987).