4E－2

平面上で二組の端子対の間の長さの和が最小で辺素な道を求めるアルゴリズム[1]

藤尾好文*　鈴木均**　西関 隆夫*
*東北大学大学院情報科学研究科, **茨城大学工学部

1. はじめに

グラフ上で辺素な道を求める問題は, これまでに様々な研究がなされている [3][4][5]. しかし, 障害物のあるユークリッド平面上で辺素な道を求める問題は, ほとんど取り扱われていない. 本文では, 平面上での辺素な道の定義を与え, さらに, 平面上で二本の辺素な道で長さの和が最小なものを求める効率の良いアルゴリズムを与える. ただし, 障害物の形状は長方形であるとし, また取り扱う道は, 座標軸に水平, 垂直な線分だけからなっているとする. 本文のアルゴリズムの計算時間は $O(n\log n)$ である. ここで n は障害物の個数である. 平面上で, 2点間の最短路を求めるだけで $\Omega(n\log n)$ 時間必要なので, このアルゴリズムは最適であることがわかる.

2. 準備

本節ではいくつかの定義を与え, 扱う問題を定式化する.

道で結びたい2点の組を**端子対**と呼ぶ. 本文では, 2組の端子対 (a,a'), (b,b') が与えられるとする. 端子対の各点を**端子**と呼ぶ. また, 道は座標軸に水平, 垂直な線分からなるものとする. 道 P_1 と道 P_2 が辺素であるとは, $P_1\cap P_2=\phi$ であるか, または, $P_1\cap P_2$ の各連結成分が点である時である. $P_1\cap P_2$ の連結成分中に線分があれば, P_1 と P_2 は辺素ではない. 2本の辺素な道で長さの和が最小なものを**最短辺素道**と呼ぶ. 障害物の形状は長方形であるとし, 2つの長方形は接しても良いが重ならないものとする. なお, 障害物が接している部分, すなわち**接合部**にも道を通せるものとする. しかし, 2本の道 P_1,P_2 が同じ接合部を通るとすると, P_1 と P_2 は辺素な道ではないことに注意しよう. 2点 p_1 と p_2 の間の最短路を $[p_1,p_2]$ と書く.

次に問題を定義する. 平面上に n 個の長方形障害物, および2組の端子対 $(a,a'),(b,b')$ が入力として与えられる. この時 $a-a'$ を結ぶ道 P_a と $b-b'$ を結ぶ道 P_b のうちで P_a と P_b が辺素であり長さの和が最小なもの, すなわち最短辺素道を求める.

3. アルゴリズム

本文では, 最短辺素道を求めるために, 障害物と端子によって, 平面を $O(n)$ 個の長方形領域に分割する [?]. さらにその分割からグラフ $G=(V,E)$ を作る. ここで, 点集合 V は分割された長方形の頂点の集合であり, 辺集合 E は長方形の辺の集合である. さらにグラフ G の辺のうち, 接合部以外の辺を2本の多重辺で置き換えたグラフを G' とする. G' の点数と辺数はいずれも $O(n)$ であり, G' は $O(n)$ の記憶量を用いて $O(n\log n)$ 時間で求まる [5]. 図1の (a) は, 問題例であり, (b) は (a) から得られるグラフ G' である (図1).

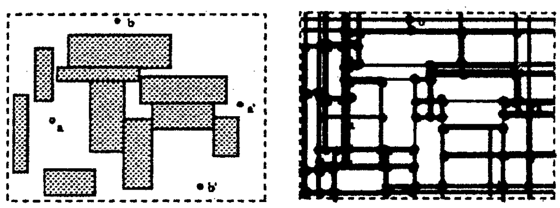
(a)　(b)

図 1: (a) 問題例 (b)(a) から得られるグラフ G(太線は多重辺を表す)

G' の多重辺を2本の道が通るならば, それらの道は元の平面領域では同一の線分を通る. したがって, G' のグラフ上で求められた辺素な道が直ちに平面上の辺素道となるとは限らない. しかし, G' 上の多重辺は平面上で接合部ではないので, どちらかの道を微小に平行移動することで, 辺素な道にすることができる. この微少な平行移動を無視すれば, G' の辺素な道は平面領域の辺素な道に対応するので, 本文ではグラフ G' 上で最短辺素道を求める問題を考える.

もし, $[a,a']\cap[b,b']=\phi$ ならば $[a,a']$ と $[b,b']$ は最短辺素道である. したがって以下では $[a,a']\cap[b,b']\neq\phi$ とする. また $[a,a']\cap[b,b']$ は一つの連結成分, すなわち, 1本の道だけであるとしても一般性を失わない. その道を $[p,q]$ とする. 最短辺素道に関して, 次の補題を証明することができる.

補題 1 $[a,a']\cap[b,b']$ の連結成分を $[p,q]$ とする. $p\in P_a\cup P_b$ かつ $q\in P_a\cup P_b$ であるような最短辺素道 P_a,P_b が存在する.□

辺数が $O(m)$ のグラフ上で2点 p,q 間の最短辺素道を求める $O(m\log m)$ 時間のアルゴリズムが知られている [3]. p,q 間の最短辺素道を P,Q とする. 図2(a) に $[a,a']$, $[b,b']$ の例を示し, (b) にその場合の P,Q を示す. P と二つの端子 a,a' を結んだ道 P'_a と, Q と二つの端子 b,b' を結んだ道 P'_b は辺素な道であるが, 最短辺素道であるとは限らない. しかし, 以下で与えるいくつかの道の組合せのいずれかが最短辺素道であることを証明できる.

[1] An algorithm for finding a pair of shortest edge-disjoint paths in the plane.
Yoshifumi Fujio*, Hitosi Suzuki**, Takao Nishizeki*
Graduate School of Information Sciences, Tohoku University
**Faculty of Engineering, Ibaraki University

(1) G'から道 Pの各辺をとり除いたグラフの上の p と qから 4 つの端子までの 8 本の最短路.

(2) G'から道 Q の各辺をとり除いたグラフの上の p と qから 4 つの端子までの 8 本の最短路.

(3) (1) と (2) の道を全てとり除いたグラフの上の p と qから 4 つの端子までの 8 本の最短路.

以上をまとめると以下のアルゴリズムが得られる.

手順 1 a と a'の間の最短路 $[a,a']$ と b と b'の間の最短路 $[b,b']$ を求め, $[a,a'] \cap [b,b']$ が有限個の点の集合ならば, $[a,a']$ と $[b,b']$ を出力し, 停止する. そうでなければ $[p,q] = [a,a'] \cap [b,b']$ とする.

手順 2 2 点 p,q間の最短辺素道を求め, それらを P,Q とする.

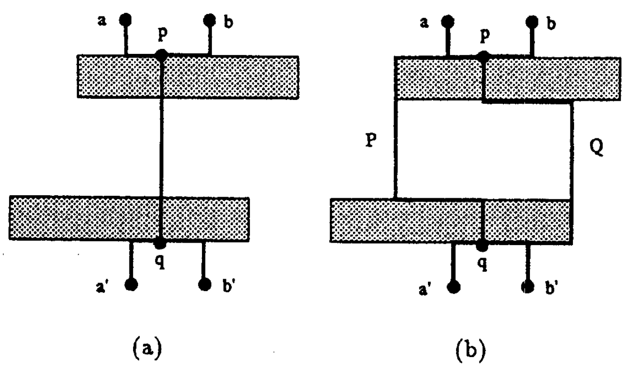

図 2: (a) $[p,q] = [a,a'] \cap [b,b']$ (b) Pと Q

手順 3 $[a,a']$ の辺を除去したグラフ上の b,b'間の最短路を B_3とし, $[b,b']$ の辺を除去したグラフ上の a,a'間の最短路を A_3とする.

手順 4 Pの辺を除去したグラフ上の a',p 間の最短路を P'_4, b',p 間の最短路を Q'_4, a,q間の最短路を P_4, b,q間の最短路を Q_4とする.

手順 5 Q の辺を除去したグラフ上の a',p 間の最短路を P'_5, b',p 間の最短路を Q'_5, a,q間の最短路を P_5, b,q間の最短路を Q_5とする.

手順 6 $A_6 = P'_4 + [a,p]$ とし, G'から A_6の辺を除去したグラフ上の b,b'間の最短路 B_6を求める. 同様に, $B'_6 = Q'_4 + [b,p]$ とし, $G' - V(B'_6)$ 上の a,a'間の最短路 A'_6 を求め, $A''_6 = P_4 + [a',p]$ とし, $G' - V(A''_6)$ 上の b,b'間の最短路 B''_6を求め, $B'''_6 = Q_4 + [b',p]$ とし, $G' - V(B'''_6)$ 上の a,a'間の最短路 A'''_6を求める.

手順 7 手順 6 の添字 4 を 5 に, 添字 6 を 7 にかえて実行する.

手順 8 各 $i = 3,6,7$ と$* = ,',\,'',\,'''$, について A^*_iと B^*_i の長さの和を求め, それが最小なものを出力する.

このアルゴリズムでは平面上で定数本の最短路を求めるだけで, 最短辺素道を求めているので, 計算時間は $O(n \log n)$ である.

4. むすび

本文では, 2 組の端子対 $(a,a'),(b,b')$ が与えられた時に, 2 本の最短辺素道を効率良く求めるアルゴリズムを与えた. 端子対が 3 組以上ある場合に最短辺素道を求める効率の良いアルゴリズムを開発することや, 道の太さを考慮した最短辺素道を効率良く求めるアルゴリズムを開発することが今後の課題としてあげられる.

参考文献

[1] E. W. Dijkstra, A note on two problems in connexion with graphs, Numer. Math., 1 (1959), 269-271.

[2] D. B. Johnson, Efficient algorithms for shortest paths in sparse networks, Assoc. Comput. Mach., 24(1977), 1-13. bibitemRDW89 P. J. de Rezend, D. T. Lee, and Y. F. Wu, Rectilinear shortest paths with rectangular barriers, Discrete and Computational Geometry, 4, (1989), 41-53.

[3] J. W. Suurballe, Disjoint paths in a network, Networks, 4(1974), 125-145.

[4] J. W. Suurballe, The single-source, all-terminals problem for disjoint paths, Unpublished technical memorandum, Bell Laboratories(1982).

[5] J. W. Suurballe, R. E. Tarjan, A Quick Meethod for Finding Shortest Pairs of Disjoint Paths, Networks, 14(1984), 325-336.

[6] R. E. Tarjan, Data structures and network algorithms, Soc. Ind. Appl. Math. (1983).