T2R2 東京科学大学 リサーチリポジトリ
Science Tokyo Research Repository

論文 / 著書情報
Article / Book Information

論題(和文)	障害物を含む領域における連結度を考慮した配線長見積もりを用いた最長配線手法
Title(English)	A Maximum Wire Length Router in Routing Area with Obstacles using Upper Bound Estimation Considering Connectivity
著者(和文)	小平行秀, 末廣傑, 高橋篤司
Authors(English)	Yukihide Kohira, Suguru Suehiro, Atsushi Takahashi
出典(和文)	第21回 回路とシステム軽井沢ワークショップ 論文集, Vol. , No. , pp. 569-574
Citation(English)	Proc. the 21st Workshop on Circuits and Systems in Karuizawa, Vol. , No. , pp. 569-574
発行日 / Pub. date	2008, 4
URL	http://search.ieice.org/
権利情報 / Copyright	本著作物の著作権は電子情報通信学会に帰属します。 Copyright (c) 2008 Institute of Electronics, Information and Communication Engineers.

Powered by T2R2 (Science Tokyo Research Repository)

第21回　回路とシステム
軽井沢ワークショップ
The 21st Workshop on Circuits and Systems
in Karuizawa, April 21-22, 2008

障害物を含む領域における連結度を考慮した配線長見積りを用いた最長配線手法
A Maximum Wire Length Router in Routing Area with Obstacles using Upper Bound Estimation Considering Connectivity

小平 行秀 †, 末廣 傑 †, 高橋 篤司 †
† 東京工業大学 大学院理工学研究科 集積システム専攻

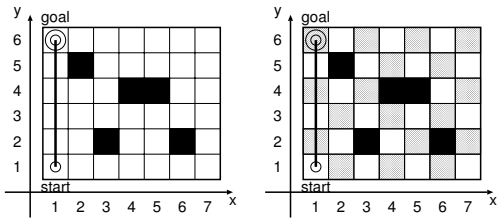
Yukihide KOHIRA†, Suguru Suehiro†, Atsushi TAKAHASHI†
† Dept. of Communications and Integrated Systems, Tokyo Institute of Technology

1　はじめに

集積回路において，複数の信号の配線長に大きな差異があると，配線遅延量の差により遅延量にずれが生じてしまう．このような遅延量のずれは致命的な誤作動を招いてしまう可能性がある．近年のクロック周波数の飛躍的な増加やプロセスの更なる微細化によって配線遅延の影響はますます大きくりつつあり，配線遅延量を高い精度で実現することが必要とされている．配線長を増やすことで配線遅延量は増大させることができるので，例えば全てのネットの遅延量を揃えるためには，遅延量の最大値に揃えるように配線遅延量を増大させればよい．遅延量を揃えるために，配線の迂回によって配線長を増やさなければならないが，配線領域には様々な要因によって配線不可能な部分があり複雑な迂回をしなければならないことが多い．各ネットに迂回に必要な配線領域を割り当てる際，与えられた領域で実現できる最大配線長を正確に見積もることができれば，限りある配線領域を適切に配分することが可能となる．

本稿では，プリント基板 (PCB) 上での一層配線を想定し，障害物を含む領域に始点と終点が与えられたときの最長配線問題について考える．障害物を含む配線領域において始点と終点が与えられたときの一層直交配線による最短経路は，迷路法などの手法で多項式時間で求めることができることが知られている [2]．一方，障害物を含む領域において始点と終点が与えられたとき，2 点間を結び全ての領域を埋める配線が存在するかを判定する問題が NP 完全であると知られている [1]．障害物を含む領域において最長経路を発見する問題が解けると，配線領域を埋める配線が存在するかを判定する問題が解けることになるので，最長経路を発見する問題は NP 困難である．

文献 [4] では，障害物を除いた配線領域と始点と終点の座標の偶奇から配線長の上界を見積もる手法と，配線領域を障害物に沿うカットラインにより障害物のない矩形領域に分割し，矩形領域の通過順を決め，矩形領域内を配線する手法が提案された．しかし，この上界の見積りでは，障害物や配線可能領域の構造を全く考慮せずに上界を見積もるため，最大配線長と見積り値に大きな差がある場合がある．また，配線手法においては，障害物の形状によってはカットラインを決められず，許容解があるのにも関わらず見つけられないことがある．



(a) 配線長 6.　　(b) 2 部グリッド.

図 1: 問題例.

そこで本稿では，障害物や配線可能領域の構造を考慮するために配線領域の連結度を用いることで配線長の上界を見積もる手法を提案する．また，その上界を見積もり手法を用い，始点から終点まで一点ずつ逐次的に配線を拡張する US 配線 (Upper bound base Seed router) を提案する．また，様々な発見的手法も導入し，提案手法の性能を計算機実験により評価する．

2　準備

2.1　問題の定義

本稿では，障害物を含む領域において一層直交配線を生成する問題について考える．

矩形領域を配線間隔で格子状に区切った小領域を**グリッド**と呼ぶ．左右方向を x 軸，上下方向を y 軸，左下のグリッドを座標 $(1,1)$，単位長さを配線間隔に対応させることで，グリッドを座標で表現する．矩形領域の左右方向のグリッド数を M，矩形領域の上下方向のグリッド数を N とする．グリッドを点，2 つのグリッドの隣接関係を枝で表現すると，矩形領域はグリッドグラフで表現できる．このグリッドグラフを $G(M,N)$，もしくは，単に G と表現する．

矩形領域内のいくつかのグリッドは障害物として与えられ，障害物は配線が通過できない．矩形領域から障害物を除去した領域を**配線領域**と呼ぶ．配線領域内

に**始点**と**終点**が異なるグリッドに 1 つずつ与えられる．配線は始点から上下左右の隣接グリッドに移動しながら各グリッドを高々 1 回だけ通過して終点へと到達する．始点と終点を含む配線経路が通過したグリッドの総数をその配線の**配線長**とする．

以上をまとめると，問題は下記のように定義される．

障害物を含む領域における最長配線問題

入力： 配線領域 (矩形領域から障害物を除いた領域)，始点，終点

出力： 配線長が最大の配線

図 1(a) に問題例を示す．矩形領域 $G(7,6)$ 内に，黒マスで表されている障害物 $(2,5),(3,2),(4,4),(5,4),(6,2)$，1 重丸で表されている始点 $(1,1)$，2 重丸で表されている終点 $(1,6)$ が与えられている．実線は配線を表し，この配線の配線長は 6 である．

2.2 2 部グリッドと配線長の上界

矩形領域はグリッドグラフなので 2 部グラフであり，障害物を除いた配線領域も 2 部グラフである．そこで，配線領域の各グリッドを (x 座標 $+y$ 座標) が偶数のグリッドを白色，(x 座標$+y$ 座標) が奇数のグリッドを灰色に 2 分割する．この白色と灰色に 2 分割されているグリッドを**2 部グリッド**と呼び，白色のグリッドを**白グリッド**，灰色のグリッドを**灰グリッド**と呼ぶ．図 1(b) に図 1(a) を 2 部グリッドで表現した図を示す．

2 部グリッドで表現した配線領域に始点，終点が与えられたとき，配線長の上界を見積もる手法が文献 [4] で提案されている．

定理 1 ([4]) 配線領域 G_r 内に始点 s，終点 t が与えられたとき，配線長の上界 $L^u(G_r,s,t)$ は下記の式で表現できる．

$$L^u(G_r,s,t)=\begin{cases}2\min\{|V_w|,|V_g|\} \\ \qquad (s \text{ と } t \text{ が異なる色}) \\ 2\min\{|V_w|-1,|V_g|\}+1 \\ \qquad (s \text{ と } t \text{ がともに白色}) \\ 2\min\{|V_w|,|V_g|-1\}+1 \\ \qquad (s \text{ と } t \text{ がともに灰色})\end{cases}$$

ここで，V_w,V_g はそれぞれ，配線領域 G_r 内にある白グリッドの集合，配線領域 G_r 内にある灰グリッドの集合を表す．

この上界は，白グリッドか灰グリッドを全て通過する場合の配線長を表している．しかし，障害物の配置により，白，灰どちらも一部のグリッドを通過することができず，最大配線長と見積り値に大きな差がある場合がある．図 2，図 3 に既存手法 [4] により得られる配線長の上界の見積り値と最大配線長に差がある例を示す．

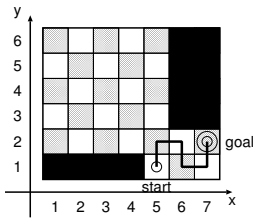


図 2: 既存 [4] の上界の見積り値 30，最大配線長 6.

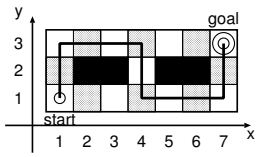


図 3: 既存 [4] 上界の見積り値 17，最大配線長 13.

3 配線長の上界の見積り

3.1 提案手法

本稿では，障害物や配線領域の構造を考慮するために，配線領域を 2 連結成分に分解し，始点と終点を繋ぐ配線が通過する 2 連結成分を求め，各 2 連結成分に対して配線長の上界を見積もることにより，配線領域全体の配線長の上界を求める手法を提案する．

グラフは，どの点を取り除いても連結ならば，**2 連結**であるという．また，2 連結という性質を満たす極大な部分グラフを**2 連結成分**という．除去すると連結グラフが連結でなくなる点を**切断点**という．一般的に，グラフは 1 つ以上の 2 連結成分に分割でき，切断点は 2 つ以上の 2 連結成分に属し，それ以外の点は 1 つの 2 連結成分のみに属する．グラフを 2 連結成分に分割する時間計算量はグラフの点数と枝数の和のオーダである [5]．

定理 2 グラフ内の 2 点 a,b を与えたとき，a,b を繋ぐパスの切断点系列 $(c_1,c_2,\ldots,c_n)$ とパスが通過する 2 連結成分の系列 $(C_1,C_2,\ldots,C_{n+1})$ が一意に定まる．

証明 グラフ G の 2 連結成分分割から，グラフ G_b を以下のように定義する．G の各点と G の各 2 連結成分をそれぞれ G_b の点で表現し，2 連結成分に G の点が含まれるときに対応する G_b 内の 2 点を枝で接続する．明らかに，G_b の点集合は G の点に対応する点集合 V_G と 2 連結成分に対応する点集合 C_G に 2 分割できるので，G_b は 2 部グラフである．グラフ G を 2 連結成分に分割したので，G_b に閉路は存在しない．つまり，G_b は木である．したがって，G の点に対応する G_b の 2 点 $a,b\in V_G$ を繋ぐパス P は一意に定まり，G_b は 2 部グラフなのでパス P は，$P=(a,C_1,c_1,C_2,\ldots,c_n,C_{n+1},b)(c_i\in V_G(1\le i\le n),C_k\in C_G(1\le k\le n+1))$ と求まる．G の点に対応する G_b の点は，G の点が切断点のときに次数が 2 以上であり，それ以外は次数 1 である．したがって，$c_i\in V_G(1\le i\le n)$ は切断点である．以上より，P から切断点系列 $(c_1,c_2,\ldots,c_n)$ と 2 連結成分の系列 $(C_1,C_2,\ldots,C_{n+1})$ が一意に定まり，命題が成り立つ． ∎

上記の定理を利用して，2 点を繋ぐパスが通過する各 2 連結成分に対して，既存手法 [4] により配線長の上界を求め，その総和を基に配線長の上界を見積もる手法を提案する (図 4)．

配線長の上界の見積り手法 $L^u_{bi}(G_r, s, t)$

入力： 領域領域 G_r，始点 s，終点 t

出力： 配線長の上界

Step 1： $s = t$ ならば 1，s, t を繋ぐパスがなければ 0 を出力し，終了．

Step 2： G_r を 2 連結成分に分割する．

Step 3： $s(= c_0)$ と $t(= c_{n+1})$ を繋ぐパスを求める．

Step 4： s と t を繋ぐパス上の切断点系列 $(c_1, c_2, \ldots, c_n)$ と 2 連結成分の系列 $(C_1, C_2, \ldots, C_{n+1})$ を求める．

Step 5： $\sum_{i=1}^{n+1} L^u(C_1, c_{i-1}, c_i) - n$ を出力して，終了．

図 4: 配線長の上界を見積もる提案手法.

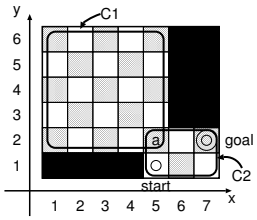


図 5: 図 2 の 2 連結成分分解．$L^u_{bi}(G_r, s, t) = 6$.

始点 s と終点 t の配線長の上界は，始点 $s(= c_0)$ と終点 $t(= c_{n+1})$ を繋ぐパスが通過する切断点の系列 $(c_1, c_2, \ldots, c_n)$ と 2 連結成分の系列 $(C_1, C_2, \ldots, C_{n+1})$ を求め，各 2 連結成分の各 2 連結成分の配線長の上界の総和を基に求める．2 連結成分 $C_i (1 \leq i \leq n+1)$ の配線長の上界は配線領域を C_i，始点を c_{i-1}，終点を c_i とした配線長の上界を既存手法 [4] により求める．これを総和すると各切断点は配線長として 2 回数えられるので，始点と終点の配線長の上界を，(各 2 連結成分の配線長の上界の総和)−(切断点の数) とする．

例として，図 2 の配線領域の 2 連結成分分解を図 5 に示す．C_1, C_2 の 2 つ 2 連結成分が得られ，始点 $s = (5, 1)$ と終点 $t = (7, 2)$ を繋ぐパスは C_2 のみを通過する．C_2 の配線長の上界 $L^u(C_2, s, t) = 6$ より，提案手法の配線長の上界は 6 となり，最大配線長と一致する．このように，提案手法は始点と終点を繋ぐパスが通過しない 2 連結成分を配線領域から除外して配線長の上界を見積もるので，既存手法 [4] よりも厳しい上界を与える．

提案手法の時間計算量について考える．前述のように 2 連結成分の分割はグラフの点数と枝数の和のオーダーだが，グリッドグラフの枝数は点数のオーダーとなる．2 点を繋ぐパスが通過する切断点と 2 連結成分を求めるのは，2 点を繋ぐパスを 1 つ見つければ可能であるので，点数のオーダーである．既存手法は，白グリッドと灰グリットの数を数えるだけなので点数のオーダーである．したがって，提案手法の時間計算量は点数のオーダーである．

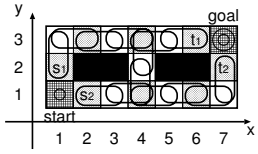


図 6: 図 3 に対する配線長の上界 $L^u_{bi,st}(G_r, s, t) = 13$.

3.2 さらに厳しい上界

提案手法の配線長の上界の見積り値は，最大配線長と必ずしも一致しない．例えば，図 3 の配線領域の 2 連結成分に分割しても 2 連結成分は 1 つになり，提案手法の配線長の上界の見積り値は，既存手法と同じ 17 である．

最長配線を実現する配線が存在するとき，最長配線上で始点に隣接する点は配線領域内でも始点に隣接する．同様に，最長配線上の終点に隣接する点は配線領域内でも終点に隣接する．そこで，始点の隣接点と終点の隣接点の全ての組合せに対して，配線領域から始点と終点を取り除いた配線領域において，始点の隣接点を始点に，終点の隣接点を終点に設定し，提案手法により得られる上界に始点と終点の分の 2 を足した値の最大値も上界であり，この値は前節の上界と同じ値か，それよりも小さい．つまり，より厳しい上界を求めることができる．

定理 3 配線領域 G_r に始点 s，終点 t が与えられたとき，

$$L^u_{bi,st}(G_r, s, t) = 2 + \max_{s_n \in S_n, t_n \in T_n} L^u_{bi}(G_r/\{s, t\}, s_n, t_n)$$

は，配線領域 G_r 内で始点 s と終点 t を結ぶ配線長の上界である．ここで，S_n は s の隣接点の集合，T_n は t の隣接点の集合を T_n，$G_r/\{s, t\}$ は配線領域 G_r から s と t を取り除いた配線領域を表す．ただし，s_n, t_n を繋ぐパスがなければ $L^u_{bi}(G_r/\{s, t\}, s_n, t_n) = -2$，$s_n = t_n$ ならば $L^u_{bi}(G_r/\{s, t\}, s_n, t_n) = -1$ とする．

例として，図 3 に示される配線問題に対して，本節で提案した配線長の上界を求める．始点 $(1, 1)$ と終点 $(7, 3)$ を除去した配線領域に対して 2 連結成分分割を図 6 に示す．始点，終点を $(s_1, t_1), (s_1, t_2), (s_2, t_1), (s_2, t_2)$ に設定した 4 通りに対して前節で提案した手法を用いると，始点，終点を (s_1, t_2) に設定したときに最大値 11 である．この値に s と t の分の 2 を足した 13 が配線長の上界であり，最大配線長と一致する．

本節で提案した配線長の上界の見積り手法の時間計算量を考える．始点と終点の隣接点はそれぞれ高々4 点であるので，始点と終点の隣接点の対は高々16 個であり，前節で提案した配線長の上界の見積り手法を高々16 回適用する．前節で提案した配線長の上界の見積り手法の時間計算量は点数のオーダーなので，本節で提案した配線長の上界の見積り手法の時間計算量も点数のオーダーである．

4　提案配線手法

本節では，3.2 節で提案した配線長の上界の見積り手法を用いた配線手法である**US 配線**(Upper bound base Seed router) を提案する．

始点と終点を繋ぐパスが存在するとする．着目点を始点とし，着目点をパス上の隣接点に移動し，移動前の点をパスから除去するという操作を繰り返すことで，始点から終点までパスを辿ることができる．US 配線手法において，この操作と同様の操作を用い，始点と終点を接続する配線を生成する．

US 配線手法では，まず，**配線の先端**を始点，**既配線領域**を空集合とし，配線の先端が終点となるまで，配線領域から既配線領域を除いた**未配線領域**に含まれる隣接点に配線の先端を移動し，移動前の点を既配線領域に加える操作を逐次的に繰り返す．

US 配線手法では，隣接点を選ぶ指標として，3.2 節で提案した配線長の上界の見積りを用いる．未配線領域内の配線の先端の隣接点全てに対して，配線領域を未配線領域から配線の先端を除いた領域，配線の先端の隣接点を始点，問題で与えられた終点を終点とした配線長の上界を，3.2 節で提案した配線長の上界の見積り手法により求め，配線長の上界が最大の隣接点に配線の先端を移動する．

3.2 節で提案した上界の見積り値は，始点と終点を結ぶパスが存在するとき正の値となり，始点と終点を結ぶパスが存在しないとき 0 となる．配線の先端を移動する隣接点を選択する際に必ず隣接点の配線長の上界を求めるので，既配線領域で配線領域を複数の連結成分に分割しても，終点が含まれる連結成分の隣接点の配線の上界が正の値，終点が含まれない連結成分の隣接点の配線の上界が 0 となり，終点が含まれない連結成分の隣接点に配線の先端が移動することはない．したかって，必ず問題で与えられた始点と終点を繋ぐパスが得られる．

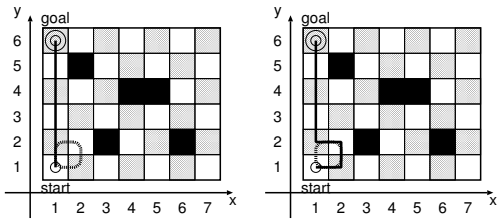
配線長の上界が最大の隣接点が複数ある場合は，配線長の下界となる終点から始点までの距離を第 2 の選択基準とし，配線長の上界が最大の隣接点の中で終点からの距離が最長の隣接点の任意の 1 点に配線の先端を移動する．

US 配線手法の時間計算量について考える．配線の先端が移動する隣接点を選択する際に，3.2 節で提案した配線長の上界の見積り手法を隣接点数回適用する．隣接点数は配線の先端が始点の場合は高々4，配線の先端が始点以外の場合は高々3 なので，点数のオーダーで配線の先端を隣接点に移動できる．配線の先端の移動回数は，配線が配線領域を埋めつくした場合が最大であるので，高々点数回である．したがって，この US 配線手法の時間計算量は点数の 2 乗オーダーである．

5　その他の発見的な配線手法

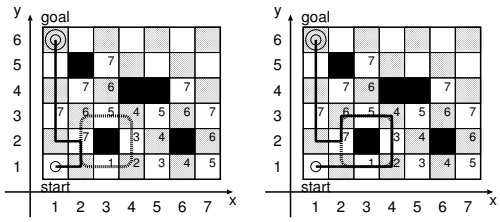
本稿では，US 配線手法の性能を比較するため，US 配線手法以外の発見的手法を導入する．配線を生成するため，始点と終点を接続する**種配線**と，始点と終点の接続性を保証したまま，配線長を増加させる**配線拡張**の 2 つのフェーズを用いる．US 配線手法は，始点と終点を接続する配線手法であるので，種配線手法の 1 つである．

本章では，種配線における配線の先端が移動する隣接点の選び方として，US 配線の隣接点を選ぶ 2 番目の選択基準であった，距離が最大の点を選ぶ**最遠配線**を導入する．



(a) 拡張前.　　(b) 拡張後.

図 7: 4 点サイクル拡張.



(a) 拡張前　　(b) 拡張後

図 8: 増大サイクル拡張.

種配線では必ずしも最長配線を得られない．種配線手法で得られた配線の一部を切り捨て，未配線領域内で得られるパスに繋ぎ変えることで，始点と終点の接続性を保証したまま，より配線長が大きい配線を得られることがある．そこで配線拡張手法として，4 点からなるサイクルに着目する**4 点サイクル手法**と未配線領域での距離と配線長に着目する**増大サイクル手法**を導入する．

5.1　最遠配線

終点から始点までの距離は配線長の下界であり，これは迷路法などの手法で求めることができる [2]．そこで最遠配線では，US 配線の隣接点を選ぶ 2 番目の選択基準であった，未配線領域における終点から配線の先端の隣接点までの距離を用いる．配線の先端を移動する際に，未配線領域における終点から配線の先端の隣接点までの距離を求め，距離が最も遠い任意の隣接点に配線領域を移動する．

最遠配線の時間計算量について考える．終点から配線の先端の隣接点までの距離は点数のオーダーで計算可能である．配線の先端の移動回数は US 配線と同様，高々点数回であるので，最遠配線の時間計算量は点数の 2 乗オーダーである．

5.2　配線拡張手法

本節では，配線拡張手法として，4 点サイクル手法と，増大サイクル手法を導入する．導入する 4 点サイクル手法と，増大サイクル手法は，既配線と未配線の領域がちょうど 1 区間ずつあるようなサイクルに対して，既配線と未配線を入れ換える修正を行う．

4 点サイクル手法は，配線領域内の 4 つのグリッドによるサイクルの 2 点が既配線領域に，他の 2 点が未配線領域に属し，既配線領域に属する 2 点を接続する枝が配線に含まれる場合，その枝を配線から除去し，4 点サイクル上のパス (既配線，未配線，未配線，既配線) を配線に繋げる (図 7)．4 点サイクル手法は，配線長をちょうど 2 ずつ増やす配線拡張である．種配線の配線長から，4 点サイクル手法を繰り返すことで配線長が改善しなくなるときの配線長までで，偶奇が一致する任意の配線長を実現できるので，指定した配線長を実現する配線問題に対して有効である．

増大サイクル手法は，既配線上のある 1 点 a から既配線上の他の 1 点までの未配線領域のみを経由したときの距離が配線上の距離より大きい点の中で，最も配線上の距離の距離が小さい点 b に対して，a から b までの配線を除去し，a から b までの未配線領域上の最短パスを配線に繋げる (図 8)．4 点サイクルは，未配線領域のみを経由したときの距離が 3，配線上の距離 1 なので，増大サイクル手法も 4 点サイクル手法を行う．つまり，増大サイクル手法は 4 点サイクル手法をより一般化した配線拡張である．

4 点サイクル手法，増大サイクル手法ともに，始点から終点へ順に配線拡張を行い，配線上のどこにも配線拡張ができなくなるまで繰り返す．

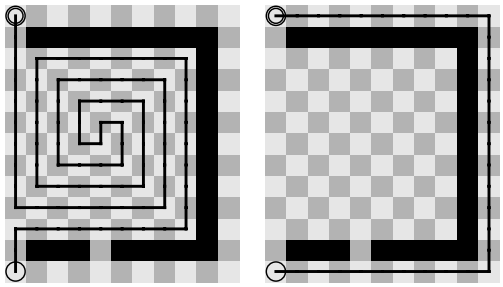
平面配線領域上の配線に対して，面に着目して配線を微修正する手法として Flip[3] が提案されている．Flip では，面を形成するサイクル上に既配線と未配線の領域が 2 区間以上あるような面に対しても配線の修正行う．4 点サイクル手法による修正は，4 点からなる面に対する Flip を 1 回行う修正に対応する．一方，増大サイクル手法による修正は，Flip による 1 回の修正では実現しないことがあり，増大サイクル手法は Flip に比べ，配線長を伸ばす操作に特化した手法である．

6 計算機実験

提案手法の有効性を確かめるため，文献 [4] で挙げられている 8 つの例題 (data1-data8) と新たに作成した 2 つの例題 (large1,large2) に対して，上界と配線を求める計算機実験を行った．配線長の上界を見積もる手法としては，既存手法 [4]，3.1 節で提案した配線長の上界の見積り手法 (提案手法 1) と 3.2 節で提案したより厳しい上界を与える手法 (提案手法 2) を比較する．配線手法としては，種配線として，迷路法による最短配線，最遠配線，US 配線の 3 手法，配線拡張手法として，何もしない，4 点サイクル手法，増大サイクル手法の 3 手法を用い，それぞれの種配線に対してそれぞれの配線拡張手法を適用することで，計 9 手法を適用する．3.40GHz/1GB Intel Pentium-4 CPU，1GB RAM の PC に gcc3.5.5 の C++で実装し，配線長の上界の見積り値，配線手法の配線長，配線手法の計算時間を比較する．

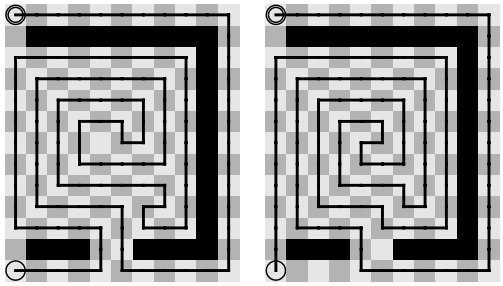
実験結果を表 1 に示す．なお，表 1 中の既存の配線手法の配線長は文献 [4] より転載した．各例題に対する各手法の計算時間は，多くの場合で 0.01 秒未満となった．そこで，表 1 には，計算時間が 0.01 秒以上の場合にのみ示す．

配線長の上界の見積りは，data1-8 の 8 つの例題では，始点と終点を含む配線領域の 2 連結成分は 1 つとなるため，既存の配線長の見積り [4] と提案手法 1 は同じ値となる．一方，large1,large2 では，始点と終点を含む配線領域は複数の 2 連結成分に分割されるため，提案手法 1 は既存の配線長の見積り [4] より厳しい上界を与える．提案手法 1 では 2 連結成分は 1 つとなった data6, data7, data8 でも，提案手法 2 は複数の 2 連結成分に分割するため，他の見積りより厳しい上界を得る．



(a) 最短+4 点 (配線長 85)．　(b) 最遠+4 点 (配線長 33)．

図 9: data7 の配線結果．

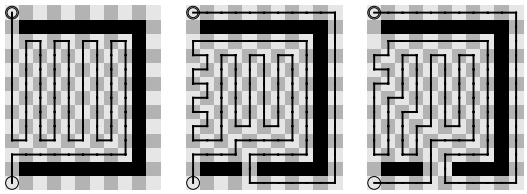


(a) 最遠+4 点 (配線長 115)．　(b) 最遠+増大 (配線長 113)．

図 10: data8 の配線結果．

既存の配線手法 [4] を data4,data5 に適用すると，障害物の形状によりカットラインを決められず，始点と終点を繋ぐパスが存在するのにも関わらず見つけられない．一方，今回適用した 9 つの配線手法は，全ての例題に対して始点と終点を接続する配線が得られる．

種配線として，最短配線，最遠配線を用い，配線拡張を 4 点サイクル手法，増大サイクル手法を用いる配線手法では，平均すると，種配線は最短配線よりも最遠配



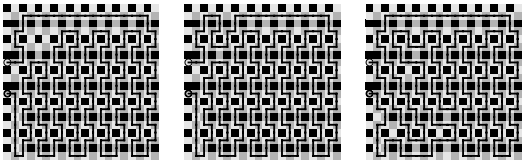
(a) data6(配線長 85)．　(b) data7(配線長 113)．　(c) data8(配線長 115)．

図 11: US 配線による配線結果．

表 1: 実験結果．計算時間は 0.01[s] 以上の場合にのみ示す．

	領域	上界			最短配線				最遠配線			US 配線 (提案)		
		既存 [4]	提案 1	提案 2	既存 [4]	—	4 点	増大	—	4 点	増大	—	4 点	増大
data1	10x9	82	82	82	80	10	78	80	78	80	80	80	80	80
data2	11x13	119	119	119	115	23	119	119	105	115	115	119	119	119
data3	11x13	109	109	109	87	13	77	107	105	107	107	107	107	107
data4	11x13	105	105	105	N.G.	13	37	103	99	103	103	103	103	103
data5	11x13	97	97	97	N.G.	13	37	95	91	95	95	95	95	95
data6	11x13	115	115	85	33	13	85	85	33	33	33	85	85	85
data7	11x13	115	115	113	33	13	85	89	33	33	113	113	113	113
data8	11x13	117	117	115	33	13	89	89	33	115	113	115	115	115
large1	20x20	291	267	267	N.A.	9	9	231	251	251	261	251	251	251
												0.05[s]	0.05[s]	0.06[s]
large2	70x100	6654	6650	6650	N.A.	154	6584	6574	6480	6538	6538	6622	6624	6624
								1.34[s]	2.78[s]	2.78[s]	3.43[s]	42.85[s]	42.85[s]	43.01[s]

N.G.　配線が得られない
N.A.　配線手法を適用していない
—　種配線後に配線拡張手法を適用しない
4 点　種配線後に 4 点サイクル手法を適用する
増大　種配線後に増大サイクル手法を適用する



(a) 最遠 (配線長 251).　(b) 最遠+増大 (配線長 261).　(c) US(配線長 251).

図 12: large1 の配線結果.

線，配線拡張は 4 点サイクル手法よりも増大サイクル手法の方が良いが，data7 では最遠配線よりも最短配線の方が良く (図 9)，data8 の最遠配線適用後の配線拡張は，増大サイクル手法より 4 点サイクル手法の方が良くなる (図 10) ので，与えられる入力により得意，不得意がある．

他方，提案した US 配線では，配線拡張をしなくても，多くの場合で他の手法の中での最良解と同等，もしくはさらに良い解を得る (図 11)．ただし，large1(図 12) のように配線領域が規則的な例題の場合，US 配線は規則性を見つけることができず，配線長が大きい解を見つけられない．また，large2(図 13) のように配線領域が大きい時には，他の配線手法より，計算時間が長い．

7　まとめと今後の課題

本稿では，配線領域の 2 連結性に着目し，配線長の上界を見積もる手法を提案した．また，その上界を見積もり手法を用い，始点から終点まで一点ずつ逐次的に配線を拡張する手法である US 配線を提案した．さらに，種配線，配線拡張の 2 段階で配線する発見的手法を導入し，計算機実験により評価した．提案した配線長の上界を見積もる手法は，高速であり，既存手法 [4] よりも厳しい上界が求まる．また，US 配線手法は，多くの例題で他の配線手法より配線長が大きい配線が得られた．

US 配線は他の手法と時間計算量は変わらないが，実際には計算時間が長い．今後の課題は，より高速により長い配線を得る手法を提案することである．また，本稿では 1 本の配線の最長配線問題を扱ったが，本稿で提案した手法を用い，複数の配線の指定長配線問題を解く手法を提案することである．

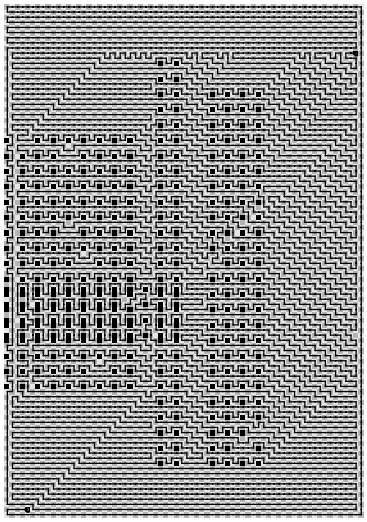


図 13: large2 の US 配線の結果 (配線長 6622).

謝辞　本研究を進めるにあたり貴重なご意見を頂いた富士通研究所澁谷利行氏に深謝致します．

参考文献

[1] A. Itai, C. H. Papadimitriou, and J. L. Szwarcfiter. Hamiltonian Paths in Grid Graphs. *SIAM, Comp.*, Vol. 11, No. 4, pp. 676–686, 1982.

[2] N. Sherwani. *VLSI Physical Design Automation: Theory and Practice.* IEEE PRESS, 1995.

[3] 久保ゆき子, 高島康裕, 中武繁寿, 梶谷洋司. Flip による自己変換するスタイナ木とその VLSI 最適配線への応用. 情報処理学会論文誌, Vol. 41, No. 4, pp. 881–888, 2000.

[4] 末廣傑, 小平行秀, 高橋篤司. 障害物を含む領域における最大配線長見積りに関する考察. 信学技報, CAS2007–97, pp. 19–23, 2008.

[5] 石畑清. アルゴリズムとデータ構造. 岩波書店, 1989.