

# 配線間隔が可変なクロストーク回避配線手法

## Flexible spacing router for non-cross-talk layout design

竹谷 啓 豊永 昌彦  
Akira Taketani Masahiko Toyonaga  
高知大学理学部数理情報科学科  
Faculty of Science, Kochi University

### あらし

本論文は, DSM 問題の1つであるクロストーク問題の配線手法からの解決方法の研究について述べるものである. ここで提案する配線手法は, クロストーク現象のみられる2つの信号配線について, 配線間隔を変更することでその影響を緩和させる. いままでも, 1 グリッド間隔のみを広げる配線方法は提案されてきたが, 本手法では, その間隔を 1 グリッド以上離すことができる. 本手法の特徴は, 距離毎の経路探索点を  $n$  個までリストに保持し, 対象となる配線からの距離を指定した値だけ離して配線経路を検索することにより実現した.

多数のリストを使い分ける上で障害となる処理速度についても, 配線領域へのラベル付け等により, 従来迷路配線法とほぼ同等程度になるよう改善されている. これら本手法の機能および優位性を, 様々な2端子問題へ適用した実験により確認したので, ここに報告する.

### 1. はじめに

現代社会で, 携帯電話やパソコンに代表される情報機器は日常生活に欠かせない. これらに加えてゲーム機やテレビ, 白物家電などあらゆる電子機器にも通信・情報処理機能が使われており, 機器の性能向上が求められている.

これらの電子機器の情報処理は, 主に半導体集積回路(一般に LSI: Large Scale Integration や IC チップと呼ばれる)が行っている.

これら集積回路では使われる素子(MOS トランジスタ: スイッチ機能を持つ半導体素子)の大きさが半分

なれば, 1 つのチップ(1 個の LSI)に集積される素子数は約 4 倍になる. つまり同じ機能のチップが微細で 4 倍作ることができ, あるいは, 製造コストが 1/4 に下げられる. さらに, 素子が微細化されれば, より低い電圧でスイッチング動作ができ, 信号入出力時間も短くなり, かつ配線長も短くなるため消費電力が減り, 性能も高まるなどの利点がある.

このような背景から集積回路は長年にわたり微細化と高集積化が進められてきた. この傾向は, 1965 年に Intel の創業者の一人であるゴードン・ムーア(Gordon Moore)博士によって提唱されたムーアの法則(Moore's Law)により 18 ヶ月に2倍のペースで微細化が進むと予測されたが, 今尚そのペースが続いている. 今後も集積回路の半導体製造プロセス技術は集積密度向上をめざし, 少なくともトランジスタが原子とほぼ同じ大きさになる 10~20 年後まで続くと言われている. (このことは国際半導体技術ロードマップ: International Semiconductor Roadmap[3]やムーア博士自身が 2007 年 9 月 18 日に開催された Intel Developer Forum における特別講演で述べている.)

一方で, 製造プロセス技術の微細化に伴い以前は無視できたナノレベルの物理現象に係る問題(DSM: Deep Sub-Micron 問題)が顕在化している.

例えば, 配線抵抗が増えて電流が流れる際に電圧が低下する現象(電圧降下)や電子移動により金属原子が動いて断線する現象(エレクトロマイグレーション)が顕著に現れるようになった. また, 高速な信号動作に伴って配線間に寄生する電荷容量(寄生容量)の影

響が顕在化してきている。これは後に述べるクロストークの一因となってしまう。

従って高性能な集積回路を効率よく製造するためには DSM 問題への対策が必要である。

本論文は、集積回路設計におけるクロストーク回避を目的とした新たな配線法を提案するものである。

クロストークとは異なる信号が電磁気的影響を及ぼしあって混信する現象である。微細化した LSI 回路の配線間には、意図しない電荷容量が寄生するが、これが原因となって信号の混信が生じる。従って配線間の寄生容量を削減すれば、クロストークを回避することができる。従来の回避策は、

1. 近接する同期信号配線の配線間隔を広げる。
2. 信号配線間に適当なシールドを施す。
3. 配線の隣接する距離をできるだけ短くする。

などが挙げられ、多くは、人手配線に対応している。

これを解決する自動配線法として、特定の信号配線について隣接する距離を離す手法[1]が提案された。従来の配線法では、配線経路探索に際して、次の検索候補点を格納するために1つのリストを用いていた。これに対し、同手法は検索候補点を距離に応じて2つのリストに格納する点で異なっている。対象とする特定の配線からの距離に応じて2つのリストを使い分けることで、配線間を1グリッド引き離して配線することができる。

しかしクロストーク回避には、1グリッドの離れた配線間隔だけでは十分といえない。動作周波数の高い回路においては、1グリッド離れた配線間の寄生容量であっても、クロストーク現象が生じるからである。(詳しくは2.1クロストーク現象の項参照)

これを解決するためには、任意の距離 $n$ だけ配線間の距離を開けた配線法が望まれる。

そこで著者は本論文において、[1]の配線法を拡張

して配線間隔を $n$ グリッド離すことができる配線法を検討した。すなわち、距離毎の経路探索候補点を特定の配線からの距離 $n$ としたとき、 $n+1$ 個のリストに格納し、特定の配線から任意の距離の隔てた配線経路を見つけ出すというものである。

しかし、単なる拡張方法では距離 $n$ の配線探索の全過程で $n^2$ 程度の手間が増加する。そのため、実際的な時間内では、 $n=1$ から2程度以上はなれた配線を行うことが困難である。(3.新クロストーク回避配線法の項参照)

そこで、著者は、配線領域に特定配線からの距離をラベル付けすることにより、 $n$ 程度の手間(従来の迷路配線法程度)でリスト格納および配線経路探索を行う方法を加えて新たなクロストーク回避配線手法(Flexible Spacing Router:FS配線)を確立した。

本クロストーク回避配線手法をプログラムに実装して、様々な配線間距離 $n$ で配線実験をおこなった。結果、いずれの場合にも配線間距離を $n$ 離し、かつ高速に処理ができることが確認された。

以下、本論文の構成は2章で研究の背景としてクロストークと配線問題について述べる。クロストークの発生原因と回避法及び、配線問題を定式化し、3章で述べる提案手法へと繋がる迷路配線法・1グリッド離すクロストーク回避配線法を解説する。3章は本論分の中心となる提案手法である新クロストーク回避配線法について、本手法の特徴及び利点を、アルゴリズムを解説することで述べる。4章は実験結果を掲載する。3章で述べる新クロストーク回避配線法の有効性を、簡単な配線問題に適用することで確認する。そして、最後に5章で本論分のまとめを行う。

## 2. 背景

### 2.1 クロストーク現象

回路にコンデンサが挿入されたとき、回路は電氣的に遮断される。一方に固定した電圧を掛けても電流は流れない。

しかし、電圧を±で変動させた場合には、コンデンサ電荷容量  $C$  \*<sup>1</sup>式1をとおして、本来繋がっていないはずの配線間にコンデンサ同様電流が流れ、図 1 に示すように互いに影響を与えてしまう。

これが信号の干渉すなわちクロストークの発生原因である。容量  $C$  が大きいほど、また信号の変化が激しいほど(高周波になるほど)この影響は大きく現れる。(詳細は付録 A.2 参照)

\*<sup>1</sup>式1:  $C \propto \epsilon \frac{S}{d}$ ,  $S$ : 面積,  $d$ : 距離,  $\epsilon$ : 誘電率,

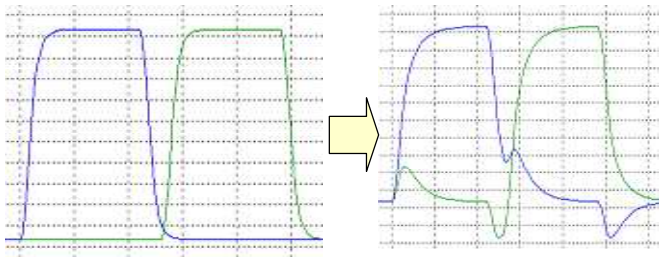


図 1 クロストークによる信号波形の変形

### 2.1.1 クロストーク回避策

クロストーク現象を回避するには、以下のいずれかを行い、配線間の寄生容量を削減しなければならない。

1. 近接する同期信号配線の配線間隔を広げる \*<sup>図2</sup>。
2. 信号配線をできるだけ短くして容量を下げる。
3. 電源線などを挿入し、シールド効果を利用する \*<sup>図3</sup>。

本論文は、「1. 近接する同期信号配線の配線間隔を広げる」による寄生容量の削減を、回避方針としている。

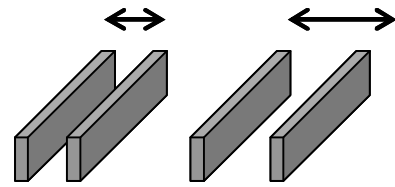


図 2 配線間隔を広げる

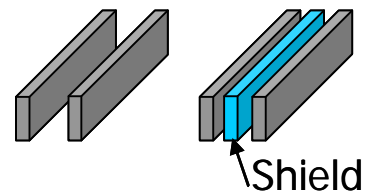


図 3 シールド効果を目指して電源配線の挿入

## 2.2 配線設計

膨大な要素を搭載する集積回路の設計は、複雑であるため階層的、かつトップダウン方式が一般にとられている。

集積回路設計のトップダウン方式とは、集積回路仕様から、製造用のマスクパターン(製造する集積回路を設計したもの)まで至る設計作業をシステム仕様設計・機能設計・論理(RTL: Register Transfer Level)設計・レイアウト設計と大まかに4つの段階に分けて分業化し、逐次設計を進めていく方式である。

各段階に分けることにより個々の作業における効率を高め、総合的に集積回路設計全体を効率よくおこなう手法である。この設計段階の順序を設計フローと呼ぶ。

ここでは4つの設計段階の内レイアウト設計についてのみ説明する(その他についての詳細は[1]を参照)。レイアウト設計段階は、さらに詳細に「配置」、「配線」という2つの小段階に分かれて行われる。

### レイアウト「配置設計」

トランジスタなどで構成した素子(セルと呼ぶ)のマスクパターンをチップ上の2次元平面上に配置する作業。集積回路製造のためのトランジスタパターンの位置を

決める。この位置に基づいてセルのマスクパターンが転写される。

## レイアウト「配線設計」

配置設計によって決まった素子の位置から各素子の端子位置が決まる。これらの接続対象となる素子端子間を電氣的に接続する配線パターンを作る。集積回路では、配線対象が膨大となるため、自動配線技術が用いられている。この配線問題は、以下のように定式化できる。

### [配線問題]

2次元上に配置された複数の端子について、

- 1) 全ての同一電位となる端子間の経路パターンを作り出す。
- 2) 経路パターンは、与えられた配線層を使う
- 3) 各配線層の設計ルールを満たす。
- 4) 配線パターンは、同一電位となる配線パターン以外と相互に接続・接触してはいけない。

この配線問題を解く方法には、迷路配線法、チャンネル配線法、線分探索法、2段階配線法などがある。

本論文の配線手法は、迷路配線を基本としている。次節で迷路配線法について詳細を述べる。

## 2.3 迷路配線法

迷路配線法では配線問題の2), 4)を満たすため配線領域に格子を設定し、格子内のみで経路を探索する方法をとる。また、異なる電位の配線は、すべて障害物として短絡しないような経路探索がおこなわれる。以下迷路配線法における配線モデルについて説明する。

### 2.3.1 配線モデル

迷路配線法では図1に示すようなグリッド状の配線

モデルが用いられる。配線は、すべてこのグリッド上に構成することで、配線問題 3)を自動的に満たすことができる。ここでは、このグリッド状の配線モデルをマップと呼ぶことにする。

まずマップの格子上には端子が配置され、自動配線は、端子間を結ぶ経路をマップ上で検索する。

膨大な端子間を配線する為には、マップを複数重ねた多層配線モデルが用いられる。多くの多層のマップを扱う自動配線では、各層毎に配線経路を縦方向か横方向かを決めて配線を行うHV (Horizontal-Vertical)規則が用いられる。例を挙げると、偶数階層を横方向、奇数階層を縦方向の配線経路で構成するという具合である。具体的にHV規則に基づく配線結果を図2に示す。図では、緑を端子および横配線の層、青を縦方向の配線層とした二層配線である。

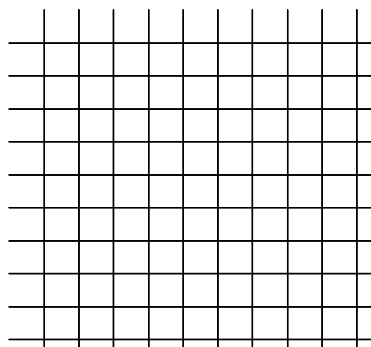


図1 マップイメージ

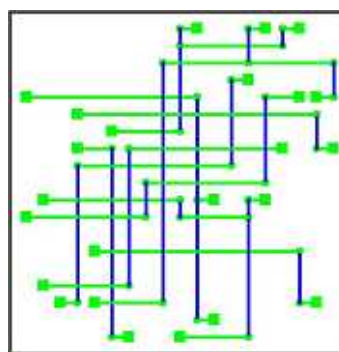


図2 迷路配線法及びHV配線法を併用した配線結果

### 2.3.2 迷路配線アルゴリズム

図 3 に迷路配線法のアルゴリズムを示す。いま、配線対象となる 2 端子を、始点および終点とする。

ステップ 1 は始点を検索リストに登録する処理である。ここで検索リストとは、配線経路を探索する中心座標となる点の情報を格納する為に用いる。

次にステップ 2 において、検索リストに登録された点を順次取り出し、その点の周囲のグリッドについて終点の有無及び配線可能(=空白点)であるかを調べる。終点がある場合はステップ 2.1 へ進み、トレスバック(終点から始点までの最短経路を辿って配線経路を確定)して経路探索を終了しステップ 5 の終了処理へ進む。配線可能である場合はステップ 2.2 へ進む。配線可能な点を候補リストへ格納し、マップ上に中心への後戻り記号(トレスバックコード:TB コード)を印す。ここで候補リストとは、経路探索の結果見つけた配線経路候補の点(以下、候補点とする)の情報を格納する為に用いる。

ステップ 3 では未配線かの判定が行われる。候補リストが空であれば、次の経路探索が行えなくなり未配線が確定する。

ステップ 4 では次の経路探索準備として候補リストの点を検索リストへ移す。その後ステップ 2 へ戻り、終点を発見するか候補リストが空になるまでステップ 2~ステップ 4 を繰り返す。

ステップ 5 は配線経路を出力して終了する終了処理を行う。

以上が迷路配線法のアルゴリズムである。

また、図 4.1~図 4.3 に簡単な配線問題における迷路配線法の振る舞いを示す。図において、検索リストを List、候補リストを List1 とし、始点を S、終点を T、n1~n15 経路探索の中心に隣接する探索対象となる点、

矢印を TB コードとする。

図 4.1 は検索リストに登録された始点 S を中心に経路探索を行った段階である。経路探索の中心 S の隣接点 n1~n4 は配線可能(=空白点)である。従って n1~n4 は候補リストへ登録される。

続いて図 4.2 において、n1~n4 には TB コードが印され、次の経路探索に用いる為に、検索リストへ移される。

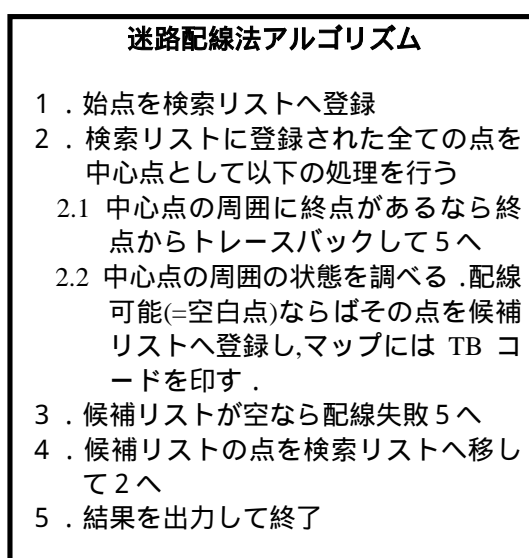


図 3 迷路配線アルゴリズム

図 4.3 は図 4.1, 4.2 の経路探索処理を終点 T が見つかるまで繰り返した段階を示している。検索リストに登録された n5(マップ中の太丸)を中心に経路探索を実行すると、隣接点 n15 を探索した段階で終点 T が見つかる。従って経路探索処理を終了し、トレスバック(終点から始点まで TB コードを辿る)後、配線経路を確定する。その様子を図 4.4 に示す。

(詳細は付録 A.1 を参照。)

以上の説明からわかるように迷路配線法は「検索リスト」と「候補リスト」を用いて始点から終点まで経路があれば必ず配線できるといわれる。

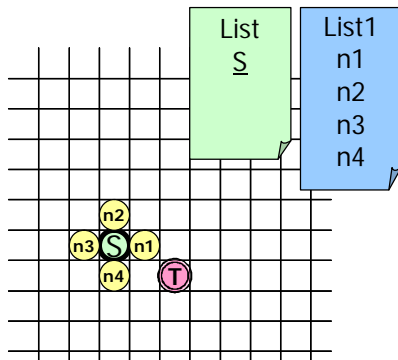


図 4.1 経路探索と候補リストへの登録

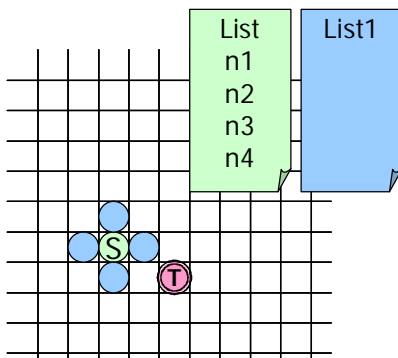


図 4.2 TB コードの印字とリスト移動 1

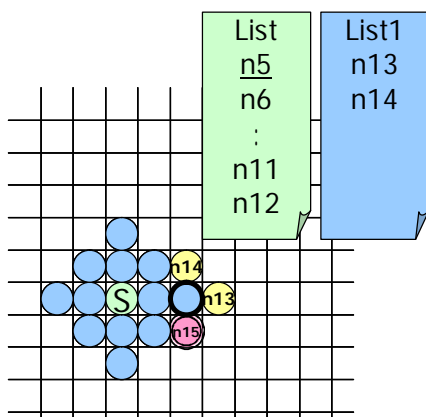


図 4.3 終点の発見

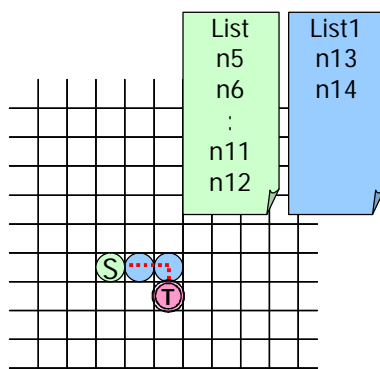


図 4.4 トレースバック

### 2.3.3 一グリッド離すクロストーク回避配線

図 2 で説明したような配線間隔を広げることで配線間のクロストークを回避することができる。

これを実装するためには、例えば図 5 のように予め禁止ブロックのような配線禁止領域を設置して迷路配線法することで可能となる。

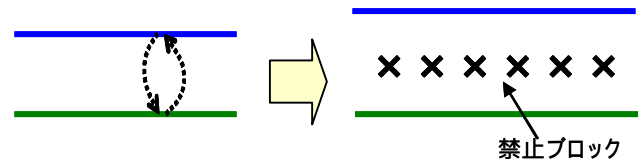


図 5 禁止ブロックの挿入

しかしこの方法では、使用できない配線領域を作ってしまうため、配線領域を十分利用できなくなる可能性がある。こうなると配線問題「1)すべての配線をする」が満たせなくなる。

そこで、本研究室は指定ネットとの隣接を回避する迷路配線法[1]で、禁止ブロックを設置せずに回避配線から一グリッド離れて配線する配線法を提案した。

迷路配線法との違いは、下記の 3 点である。

1. ネット番号(素子や配線が持つ接続関係を表すデータ)によって、回避配線を識別可能
2. 回避配線の識別処理を経路探索時に組み込んでいる
3. 配線経路の候補点を登録する「候補リスト」を使用優先度の高い候補リスト 1 と優先度の低い候補リスト 2 に二重化した

この利点は、候補点(経路の候補となる点)を 2 つの候補リストへ振り分けることで差別化する点である。

アルゴリズムは以下の通りである。迷路探索法と同様の経路探索に、見つけた候補グリッドの隣接 4 方向を探索する識別処理を加えて行う。回避配線が見つければその候補点を候補リスト 2 へ、見つからなければ候補リスト 1 へ振り分ける。その後候補リスト 1 の候補

点のみを検索リストへ移し経路探索に利用する。ここで、未配線が起これば初めて候補リスト 2 の点を用い経路探索を行う。つまり始めは禁止ブロックを用いた時同様、回避配線の周囲 1 グリッド分少ない配線領域で配線を試みるが、配線不可であれば例外的に回避配線の周囲 1 グリッドを利用して配線を行う。これによって迷路配線と同じ配線効率を維持しつつ、極力回避配線から 1 グリッド離れることが可能となった。

以下にアルゴリズムの要約を説明する。

始点及び終点が回避配線に隣接する配線問題において、一グリッド離すクロストーク回避配線法の振る舞いを以下に示す。図において、検索リストを List, 候補リストを List1,2 とし、始点を S, 終点を T, n1 ~ n14 を経路探索の中心に隣接する探索対象となる点、矢印を TB コード、暗線部を回避配線、白抜きのを回避配線の検索点とする。

図 6.1 は始点 S を中心に経路探索を行った結果を示している。候補点 n2 は回避配線に隣接しておらず候補リスト[1]へ、n1, n3 は隣接する為候補リスト[2]へ格納される。

続いて迷路配線法同様、n1 ~ n3 には TB コードが印され、次の経路探索には使用優先度の高い候補リスト [1]に登録された n2 のみを使用される。このとき使用されなかった n1, n3 は候補リスト[2]に残る。n2 を中心に行った経路探索が終了した段階を図 6.2 に示す。探索によって発見された n4 ~ n6 は、検索リストに隣接せず候補リスト[1]に登録される。

図 6.3 に終点 T を発見するまでの、経路探索の広がりを示し。図 6.4 に配線結果を示す。S-T間の実線部が確定した配線経路を表す。

### 一グリッド離すクロストーク回避配線法

1. 始点を検索リストへ登録
2. 検索リストの点全てに対して以下の処理を行う
  - 2.1 目標に隣接するならトレースバックして 8 へ
  - 2.2 隣接点が配線可能(=空白点)なら 3 へ
3. 隣接点が回避配線と隣接しているかを調べる
  - 3.1 隣接していなければ候補リスト[1]へ登録し、TB コードを印す
  - 3.2 隣接していれば候補リスト[2]へ登録し、TB コードを印す
4. 候補リスト[1]が空なら 6 へ
5. 候補リスト[1]に登録されたグリッドを検索リストへ移し、2 へ
6. 候補リスト[2]が空なら 8 へ(未配線)
7. 候補リスト[2]に登録されたグリッドを検索リストへ移し、2 へ
8. 結果を出力して終了

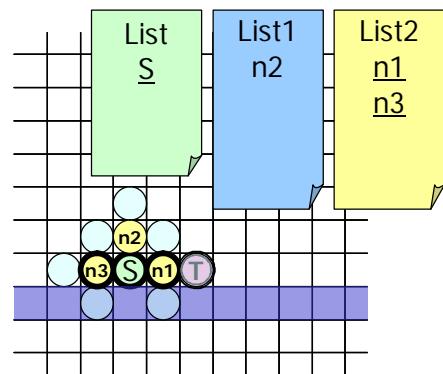


図 6.1 候補点の振り分け

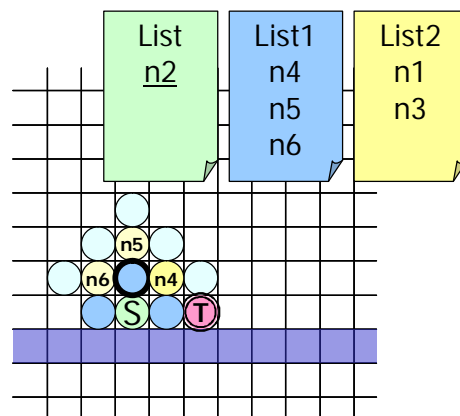


図 6.2 検索リストへの登録

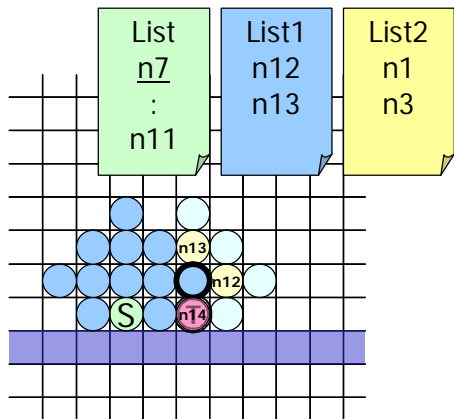


図 6.3 終点 T 発見までの経路探索結果

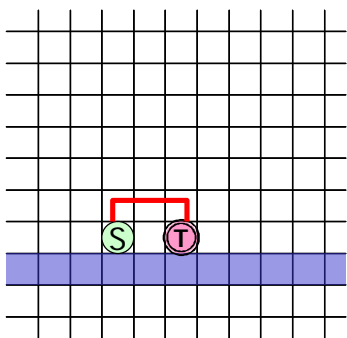


図 6.4 配線結果

### 3. 新クロストーク回避配線法

微細加工技術の進展で今後さらに集積度、動作周波数が上昇すると、一グリッド程度の距離による寄生容量の削減では不十分になるため、配線間隔をさらに広げる必要がある。

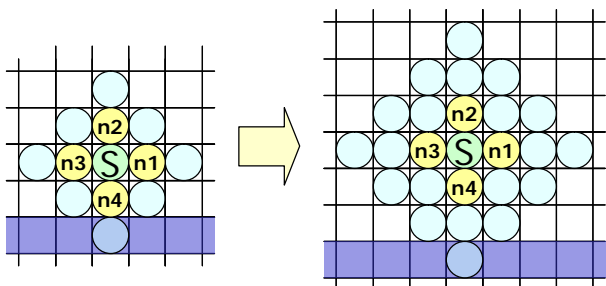


図 7 回避距離増大による検索領域の拡大

図 7 において暗線部が回避配線、S は始点であり探索中心点、n1 ~ n4 は S の隣接点、白抜きのが回避配線の検索領域を表す。

図 7 にみられるように、1 グリッド離すクロストーク回

避配線法を単純に拡張した場合、対象となる配線からの距離に応じた候補点を各候補リストへ振り分ける際の識別作業は膨大となるため、処理時間が大幅に増えてしまう。そこで著者は、より高速に配線間を任意幅離す新クロストーク回避配線アルゴリズムを提案する。同手法は下記の2点の改善をおこなっている。

1. 複数の「候補リスト」により回避対象となる配線からの距離  $d$  を任意に設定可能
2. 候補点の候補リストへの割付する分別処理が不用となり処理が高速

### 3.1 アルゴリズム

提案する新クロストーク回避配線法は迷路配線法による経路探索処理を行う前に、前処理としてマップへのラベル付けを必要とする。ラベル付けはまず回避配線からの回避距離  $d$  を設定する。その後図 8 のように、回避配線の周囲  $d$  グリッド分にだけ回避配線からの距離情報をラベル付ける。図において、S は始点、T は終点、S-T の 3 グリッド下にある暗線部は回避配線、1 ~  $d$  はラベル付けされた回避配線からの距離情報を表す。

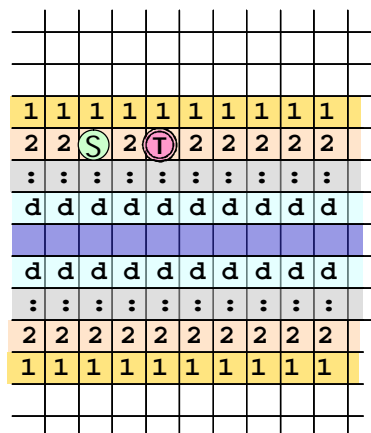


図 8 ラベル付けの様子

ここから経路探索処理へと移る。提案手法の経路探索よりはただ1つの点を除いて迷路配線法と同じアル



ゴリズムである。迷路配線による経路探索処理との相違点は、前処理によってラベル付けされたマップを用いる点にある。一グリッド離すクロストーク回避配線法ではこの経路探索処理において、回避配線を検索する識別処理を組み込んでいることが処理時間増加の原因であった。

しかし新クロストーク回避配線法は、マップにラベル付けされた回避配線からの距離情報を経路探索と同時に読み込む。この為、回避配線を検索することなく、適切な候補リストへ候補点(経路の候補となる点)を振り分けることができる。このことから回避距離が処理時間に影響することはほとんどないと考えられる。また候補リストには従来法同様に優先順位があり、優先順位の高い候補リストが空のときのみ例外的に次の候補リストを利用する。

#### 新クロストーク回避配線法アルゴリズム

1. 回避距離  $d$  を設定
2.  $d$  に従った数(候補リスト[0] ~ 候補リスト[ $d$ ] までの  $d+1$  個)だけ候補リストを用意
3. 回避配線の周囲  $d$  グリッドにラベル付け
4. 始点を検索リストへ登録
5. 検索リストに登録された全てのグリッドに対して以下の処理を行う
  - 5.1 終점에隣接するならトレースバックして 8 へ
  - 5.2 隣接点が配線可能(=空白点)なら候補リスト[0]へ登録し、TB コードを印す
  - 5.3 回避配線からの距離  $r$  がラベル付けされていれば、候補リスト[ $r$ ]へ登録し、TB コードを印す
6. 候補リスト[0] ~ 候補リスト[ $d$ ] に対して以下の処理を繰り返す
  - 6.1 候補リスト[ $r$ ] が空でないなら候補リスト[ $r$ ] の点を検索リストへ移し 5 へ
  - 6.2 空なら  $r+1$  して次の候補リストへ
7. 全候補リストが空なら未配線として終了
8. 結果を出力して終了

図 9 新クロストーク回避配線法アルゴリズム

以上のことから新クロストーク回避配線法は、迷路配線と同程度の処理時間で配線を完結でき、回避配線から  $d$  グリッド離れた配線が可能で、それが不可能であれば例外処理により次善の経路を見つけ出す特徴を持つ。図 9 にアルゴリズムの要約を示す。

図 8 の配線問題において、新クロストーク回避配線法の振る舞いを図 10.1 ~ 10.3 に示す。図において、検索リストを List, 候補リストを List0 ~  $d$  とし、始点を S, 終点を T,  $n1 \sim n6$  を経路探索の中心に隣接する探索対象となる点、矢印を TB コード、下から 2 グリッド目にある暗線部を回避配線、マップ上の数字をラベル付けされた距離情報とする。

図 10.1 始点 S を中心として経路探索した結果を示す。まず、候補リストが  $d+1$  個用意される。その後経路探索により  $n1, n3$  はラベル付けされた距離情報「2」を読み込み候補リスト[2]へ。同様に  $n2$  は候補リスト[1],  $n4$  は候補リスト[3]へ登録される。

適切な候補リストへの登録が終了した後、 $n1 \sim n4$  は TB コードが印される。続いて、次の経路探索の為に候補点が検索リストへ登録される。このときは、最も使用優先度が高い候補リスト[0]が空の為に、次に優先度の高い候補リスト[1]に登録された  $n2$  が検索リストへ移される。このときその他  $n1, n3, n4$  はそれぞれの候補リストに残ったままになる。ここまでの様子を図 10.2 に示す。

また最後に、図 10.3 に上記の経路探索を繰り返し確定する配線経路を示す。S-T 間の実線部が配線経路を表す。

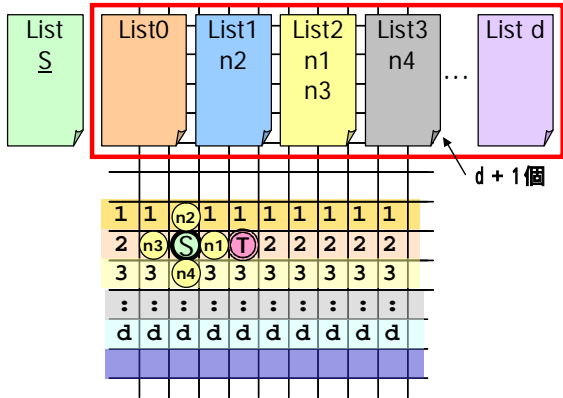


図 10.1 候補リストの準備と登録

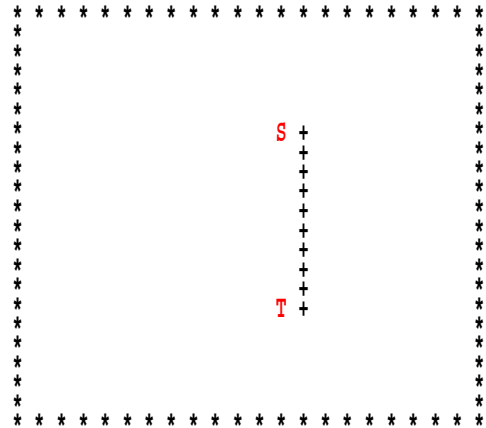


図 11.1 回避対象に接した端子間の配線問題

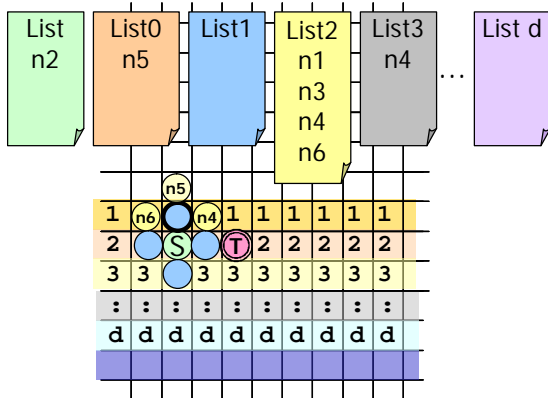


図 10.2 候補点を検索リストへ登録

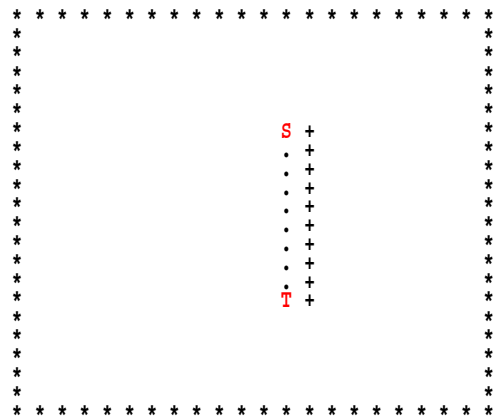


図 11.2 迷路配線法による配線経路

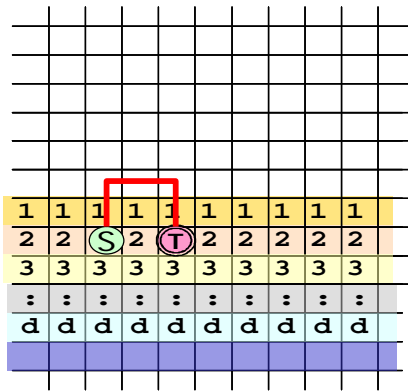


図 10.3 配線結果

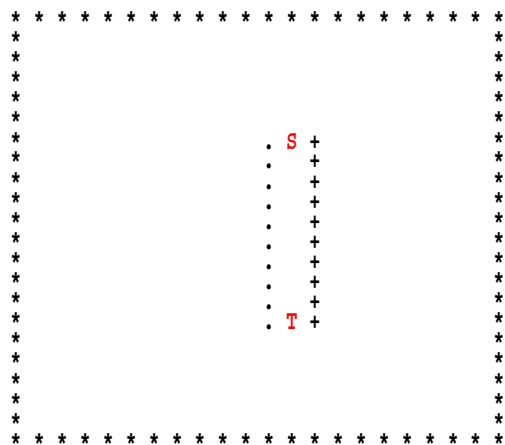


図 11.3 一グリッド離れた配線経路

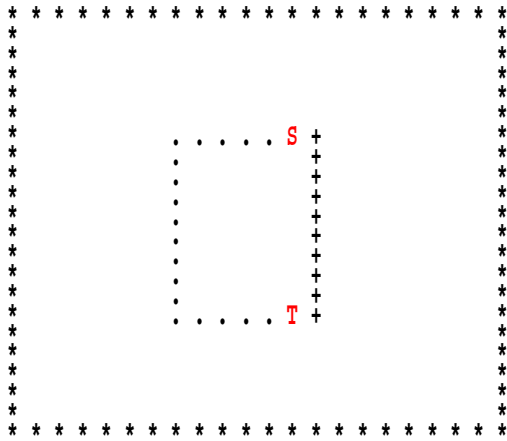


図 11.4 提案手法による配線経路

#### 4. 実験

始点終点が回避配線に隣接する場合(図 1.1), 配線途中にボトルネックをもつ場合(図 12.1)の2種類の配線問題を用意し, 迷路配線法, 1 グリッド回避する配線法, および提案配線法の比較実験をおこなった. 提案手法については回避距離  $d$  を変えた配線経路も比較する. また, これとは異なるボトルネックをもつ配線問題(図 13.1)について, 配線領域を拡大した幾つかの問題に迷路配線法と提案手法を適用し処理時間の比較とその規模依存も評価した. なお, 各図において「.」は配線, 「\*」は境界線, 「+」は回避配線, 「S」は始点, 「T」は終点を指す. また, a, b, c および d は TB コードでそれぞれ左, 右, 上および下の方向を意味する. さらにマップ上の記された数字 1~7 は回避対象の配線からの距離をそれぞれ表すものとする.

配線手法の実装は C 言語(Cygwin)を使い, 時間計測で使用した計算機の OS は Windows-XP, CPU は Celeron(2..53GHz)である.

##### 4.1 回避配線に隣接する端子間の配線

始点・終点が回避対象の配線に隣接する配線問題  
 図 11.1 に対して迷路配線法により得られた配線経路を  
 図 11.2 に, また一グリッド離れたクロストーク回避配線

法による配線経路を図 11.3 にそれぞれ示す. これらに対して提案手法は, 回避距離  $d$  を任意に設定することができる.

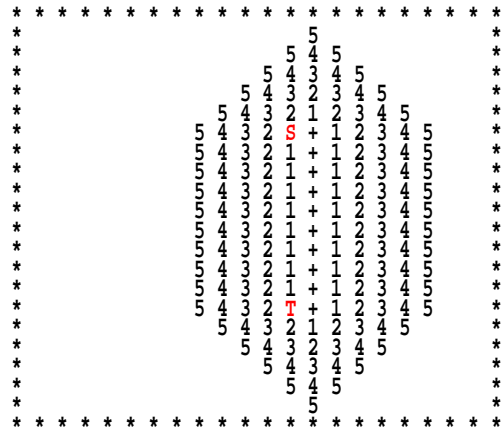


図 11.5 実際のラベル付けの様子

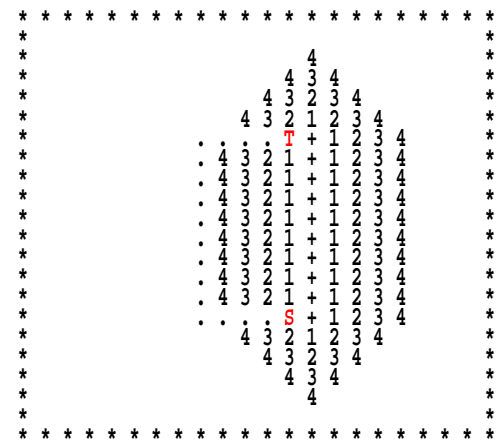


図 11.6 d=4 による配線

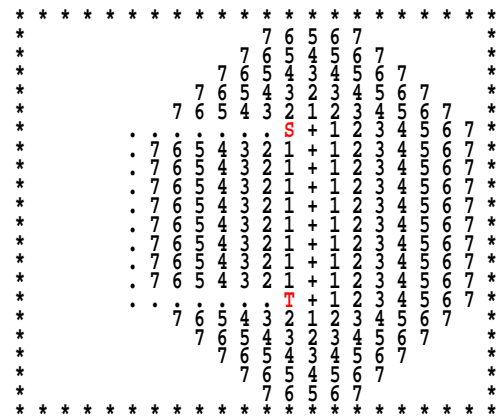


図 11.7 d=7 による配線

回避距離  $d = 5$  としたときの配線経路を図 11.4 に示す。回避距離 5 グリッド分、隣接する回避配線から離れた上で始点から終点までの配線経路が得られている。

いま配線間の距離、配線幅を同じ長さとしたとき、迷路配線における寄生容量を  $C$  とすると、図 11.3 の配線では、寄生容量を  $C' = 1/3C$  に削減することができクロストーク現象を緩和することができる。一方、 $d=5$  と任意に離せる提案配線ではさらに、 $C'' = 1/11C$  となり 1 桁以上緩和できる。今後高速動作が予想される LSI において強力な回避法として使えそうである。

提案アルゴリズムの経路探索の振る舞いを見るためマップに付されたラベルを図 11.5 に示す。この問題では始点・終点が回避配線に接している。また回避距離 5 であるため、始点・終点はラベル付け領域内になる。このとき提案アルゴリズムは、始点から非ラベル付け領域までの経路をまず検索し、その後に終点付近まで近づき、ラベル付けされた領域を通して(他の経路がないため)終点までの経路を見つけ出している。本手法は、ラベル付けされた配線領域も有効に利用していることがわかる。

参考のため図 11.6 と図 11.7 に、 $d=4$  および  $d=7$  のときの配線経路をそれぞれラベルとともに示す。指定された距離に応じて、ラベルを判断し、図 11.5 と同様に経路が求められていることが確認できる。

#### 4.2 ボトルネックを持つときの配線

次に、図 12.1 に示す配線問題に適用する。この配線問題の特徴は、回避配線と障害領域が始点・終点間に狭い部分(ボトルネック)を作っている点である。配線禁止ブロックを設置してクロストークを回避する手法では、禁止ブロックがボトルネックを塞ぐため遠回り、あるいは未配線となる。

しかし、 $d=3$  とした場合の提案手法では図 12.2 に見られるような配線経路を見出すことができた。

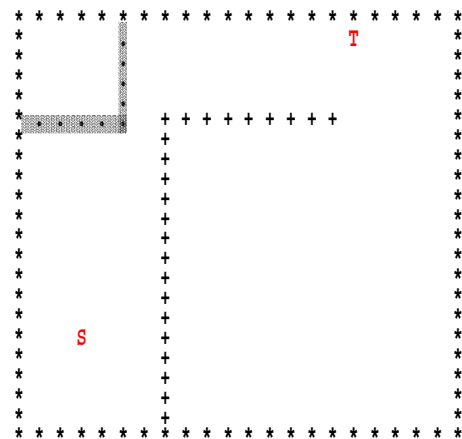


図 12.1 ボトルネックを持つ配線領域

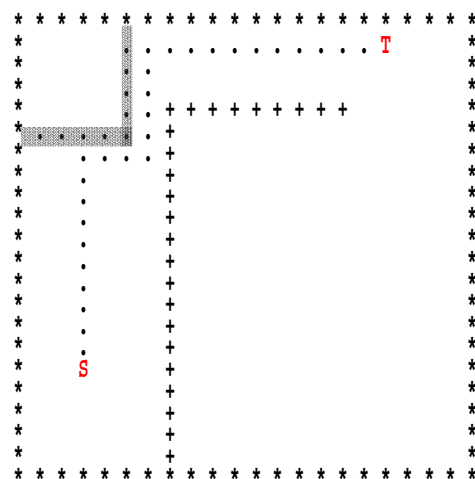


図 12.2 提案手法で得られた配線経路

提案手法の経路探索の振る舞いを見るためマップに付されたラベルを図 12.3 に示す。

始点・終点は、回避配線から距離 3 以上離れているため、経路探索はラベル付け領域外で行われる。図 12.4 にラベル付け外の経路探索まで行ったときの配線領域の図を示す。(図中 a, b, c および d は TB コードでそれぞれ左, 右, 上および下を表す)

ボトルネック部があるため、ラベル付け以外の領域では経路が見出せないことがわかる。

図 12.4 は、本提案手法がラベル付け領域にまで検

索範囲を広げてボトルネック領域を通過し,再び回避ネットから距離を置いた経路探索が行われている様子を示している.このように本手法は,ボトルネックがある場合であっても,経路が存在すれば配線探索を行うことができる.

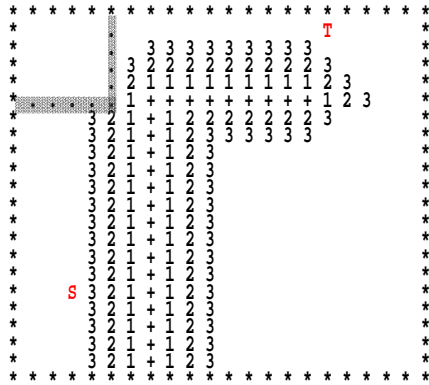


図 12.2 ラベル付けの様子

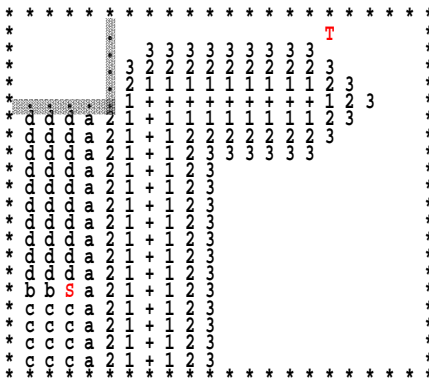


図 12.3 TB コードの設定 1

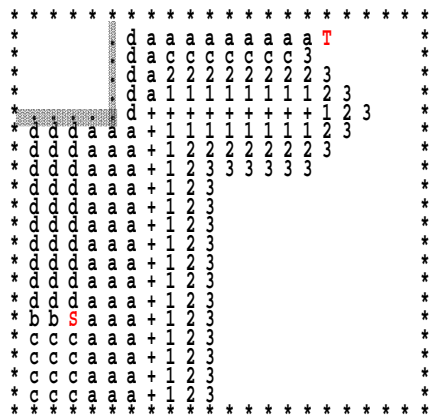
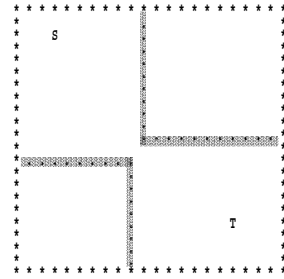
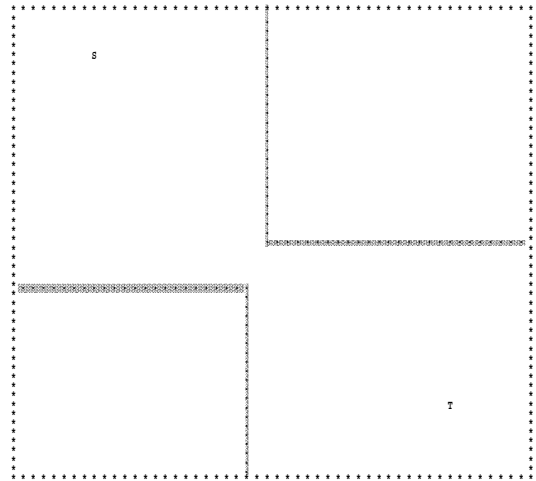


図 12.4 TB コードの設定 2



(a) サイズ 20x20



(b) サイズ 50x50

図 13.1 規模の異なる配線問題

図 12.2 に相当する配線を,グラフィックツールを用いて描いた図を示す.

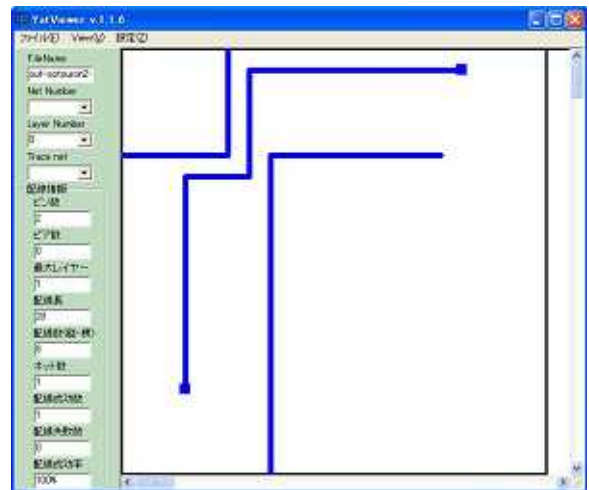


図 12.6 グラフィックツールによる配線結果表示

### 4.3 処理時間の比較

迷路配線法と提案配線法 ( $d = 3, 5, 10$ ) についてボトルネックを規模が異なる (サイズ  $20 \times 20, 30 \times 30, 40 \times 40, 50 \times 50$ ) 配線問題に対して時間比較を行った。図 13.1(a)に  $20 \times 20$ , (b)に  $50 \times 50$  の各配線問題を示す。

処理時間の結果を表 1 に示す。表 1 において、「迷路配線」に迷路配線の処理時間を、また「1」「5」「10」には  $d = 3, 5$  および  $10$  としたときの提案配線法の処理時間 (単位ミリ秒) を記す。

表 1 各配線法の総実行時間及び平均時間

Types \ Size	20*20	30*30	40*40	50*50
Maze	4.93	5.29	6.04	6.96
d=3	4.59	5.10	5.90	6.76
d=5	5.04	5.20	6.08	6.89
d=10	4.65	5.25	6.04	7.12

Unit:Time(ms)

$20 \times 20$  の配線問題では、迷路配線の処理時間が 4.93 ミリ秒であり、提案法の処理時間は  $d = 3$  で 4.59 ミリ秒、 $d = 5$  では 5.03 ミリ秒、 $d = 10$  では 4.65 ミリ秒である。すなわち、迷路配線および提案手法の処理時間は、ほぼ同等で、提案手法について  $d$  の依存性もほとんどみられない。

次に、表 1 から配線領域サイズと処理時間についてみてみると、迷路配線法では、 $20 \times 20$  が 4.93 ミリ秒、 $30 \times 30$  が 5.29 ミリ秒、 $40 \times 40$  では 6.04 ミリ秒、 $50 \times 50$  では 6.96 ミリ秒である。また、提案手法は、 $d=3$  においてそれぞれ 4.593 ミリ秒、5.10 ミリ秒、5.90 ミリ秒、6.76 ミリ秒となり、やはりほぼ同等な規模依存性を持っていることがわかる。また、その傾向は、 $d$  に依存しないとも言えそうである。

図 13.2 は、表 1 を横軸にサイズ、縦軸に処理時間と

してグラフ化したものである。迷路配線法を実線で、提案手法の様々な  $d$  を破線で描いた。

提案手法がほぼ迷路配線程度の処理時間でおこなえることが確認できる。

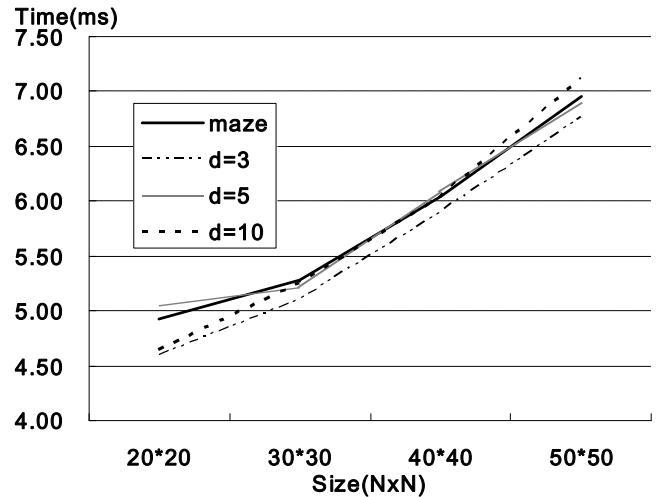


図 13.2 実行時間の回避距離  $d$  とサイズ依存

新配線法は迷路配線法に比べ、処理時間の増加が最大でも+2%程度であり、迷路配線法と同程度の処理時間で処理を完結することが確認できる。

以上の傾向について、グラフでまとめたものを表 3 に示す。縦は処理時間、横軸はマップサイズを示す。同じマップサイズについては左から、迷路配線法、新配線法 ( $d = 3, 5, 10$ ) の結果を示す。

## 5. まとめ

本論文において我々は、レイアウト設計の配線処理において、クロストーク回避を目指した配線法を提案した。提案手法が持つ特徴は、

1. 回避配線との回避距離を任意に設定できる。
2. 禁止ブロックによるクロストーク回避の手法の欠点であった未配線を例外処理により防止する。つまり、配線経路があれば必ず配線を完結でき

る。

3. 配線完結までの実行時間は迷路配線法によるものと同等。

という特徴を持つ。

また、本論文で行った実験の様に、配線領域に比較的余裕のある配線問題では回避距離  $d$  によっては、迷路配線法よりも若干ではあるが高速に配線を完結できることが確認された。今後、より実的な多層・多端子による配線問題に適用し評価する予定である。

### 謝辞

この論文の作成にあたり、熱心にご指導して下さいました豊永先生に敬意を表し、深く感謝致します。また多大な協力をしてくれた同研究室の皆様にも感謝します。

### 参考文献

- [1]谷本俊介 “指定ネットとの隣接を回避する迷路配線法” 平成 19 年度高知大学理学部卒業論文
- [2]竹谷啓, 張汎, 豊永昌彦 “クロストーク回避迷路配線の一手法” 平成 19 年度電気関係学会 四国支部連合大会 1-3 (2007 9 月 29 日)。
- [3] 国際半導体技術ロードマップ (International Semiconductor Roadmap)2006 年版
- [4]JIRI SOUKUP “Circuit Layout” IEEE, VOL.69, NO.10, OCTOBER 1981
- [5]JIRI SOUKUP “Fast Maze Router”IEEE,1978.15th Conference on Volume, pp.100-102, June19–21, 1978
- [6]Muhammet Mustafa Ozdal “Escape Routing For Dense Pin Clusters In Integrated” DAC 2007, June 4-8, San Diego, California, USA

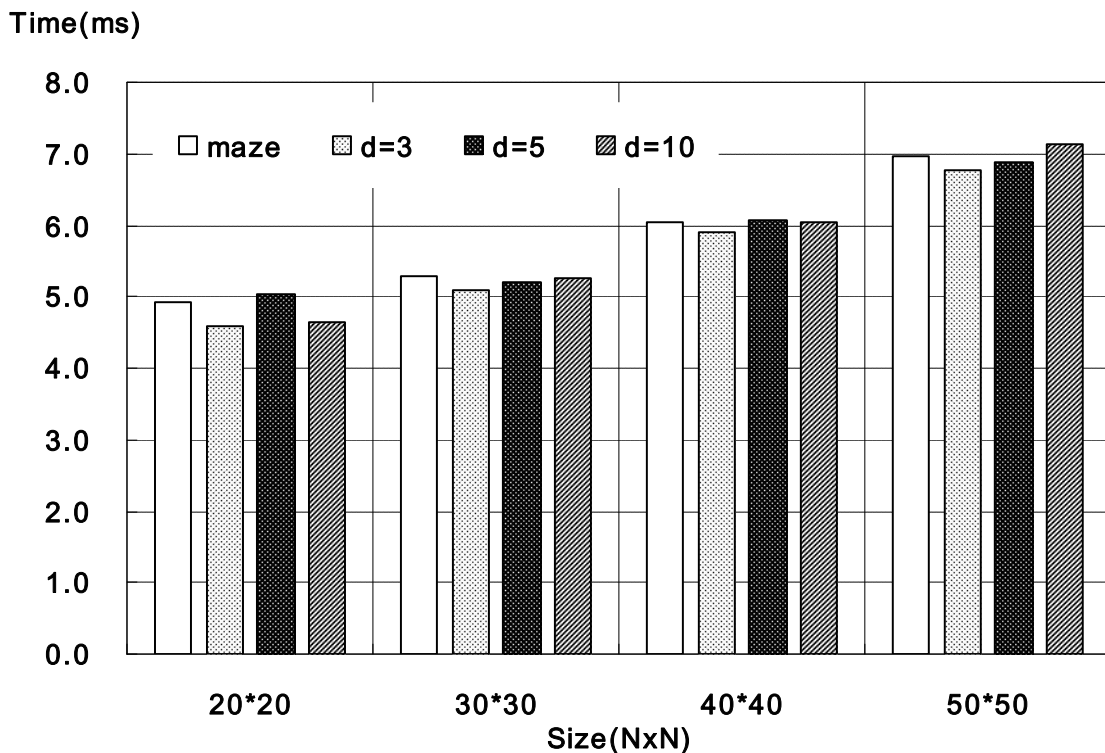


図 14 各配線法の平均実行時間の比較