

TM-1079

並列オブジェクトモデルに基づく
LSI配線プログラム

伊達 博、瀧 和男

August, 1991

© 1991, ICOT

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03)3456-3191 ~ 5
Telex ICOT J32964

Institute for New Generation Computer Technology

並列オブジェクトモデルに基づく LSI 配線プログラム *

伊達 博 瀧 和男 †

(財) 新世代コンピュータ技術開発機構 ‡

E-mail: date@icot.or.jp, taki@icot.or.jp

LSI CAD における LSI 配線設計を並列処理により高速化することは、設計期間の短縮に効果が大きいと期待されている。我々は、既に分散メモリ型の大規模な並列マシンに適用できることを目標に、並列オブジェクトモデルに基づいた新しい並列配線手法を提案した。本論文では、それを KLI 言語を用いて分散メモリ型並列マシンの Multi-PSI 上に実装し、LSI の実データを用いて性能評価した結果をまとめる。また、大規模データに対処するためのメモリ使用特性とプログラムの書きやすさとの関係について検討する。

1 はじめに

LSI の集積度の向上にともなって、高速度な未結線を出さない高品質の自動配線設計システムの実現が強く望まれている。

我々は、並列処理を用いてこれらの問題を解決すべく研究を続けており、既に、配線設計の高速化を目的とした、新しい並列配線方式を提案した。そしてそれを分散メモリ型並列マシン、Multi-PSI 上に実装し、初期評価を行った結果を報告した [1]。

本稿では、LSI の実データを用いた性能評価実験結果に対する考察と今後解決しなければならない問題点についてまとめ、さらにこの並列配線プログラムの書きやすさとメモリ効率との関係について報告する。

以下では、本配線方式の特徴、並列化に伴う問題点と対処法、測定結果のまとめ、本プログラムのメモリ効率の順に述べる。

2 本配線方式の特徴

大規模な MIMD 型並列マシンを用いて問題を解く場合、大きな並列性を内在する必要がある。そのため、計算の粒度の小さい、並列オブジェクトモデル (COM) に基づく新しい配線問題のモデル化を試みた。

2.1 COM による並列プログラムの設計法

図 1 は、COM に基づいた並列プログラムの設計方法の概念図である。最初に問題が与えられると、

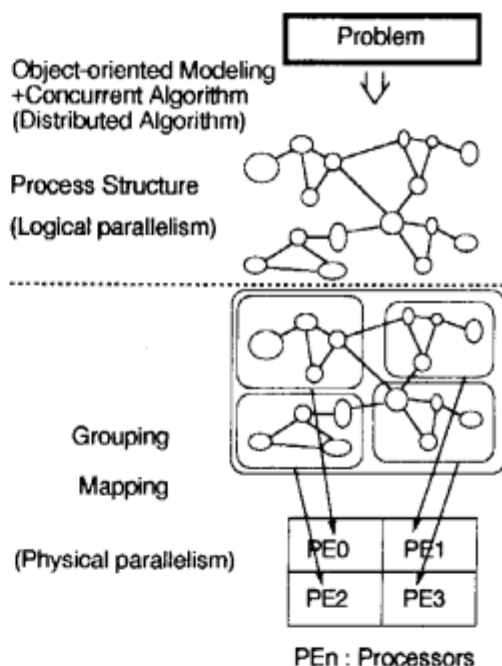


図 1: COM に基づく並列プログラムの設計法

それを複数の能動的に動作するオブジェクトがメッセージを交換しながら並列に解決するモデルに合わせて定式化し、同時に並列アルゴリズム (分散アルゴリズム) を設計する。それに基づき、プロセス構造とプロセスの動作が定義できる。この時点でアルゴリズムの論理的並列性の設計が完了する。

次にこれらのプロセスの中で頻りにメッセージを交換するものをグループ分けし、グループ単位でプロセッサに割り付ける。このことにより実行時の物理的な並列性が抽出される。このようにしてアルゴリズムの並列性設計の問題と物理的な並列実行

* A Parallel Router based on a Concurrent Objects Model

† Hiroshi DATE, Kazuo TAKI

‡ Institute for New Generation Computer Technology

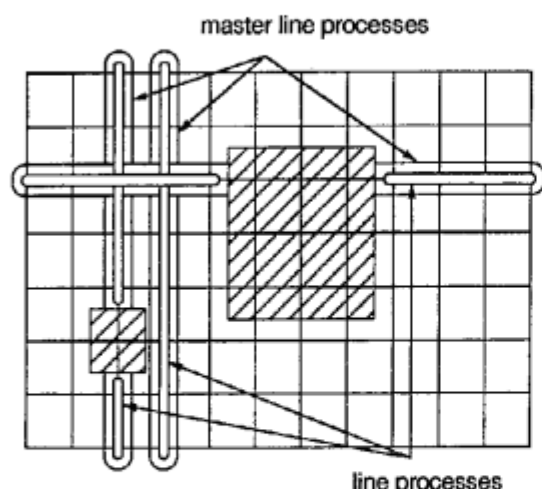


図2: マスター・ラインプロセスとライン・プロセス

への対応の問題を分離して考えられるので、効率的な並列プログラムの設計が可能となる。

2.2 並列配線プログラム

並列配線プログラムを設計するにあたり、基本アルゴリズムとして処理効率と配線品質の両方に優れた予測線分探索法[2]を選び、それを分散アルゴリズムとして設計し直した。具体的には、高い並列性を実現するため、図2に示すように、未配線、既配線を含めたすべての配線格子上の線分を互いに能動的に動作しうるオブジェクト（ライン・プロセス）としてモデル化した。そして、マスターライン・プロセスが、ライン・プロセス間のメッセージを仲介するというプロセス構造とした。この場合、配線アルゴリズムは、必然的にオブジェクトが互いにメッセージ交換しながら良い配線経路を決定してゆく分散アルゴリズムとなる。

また本プログラムでは、2通りの並列性を実現している。第1は、1対の端子間を配線するアルゴリズムの中で、予測線分探索法の先読み部分を並列化している。ここで先読み処理について簡単に説明する。予測線分探索法では、始点から目標点へ線分探索を行う過程で、始点から折れ曲がり一回で到達可能な点の中で、目標点に一番近い点を探す。これを先読み処理と呼ぶ。次にその折れ曲がり点までを結線し、その点を新たな始点とし、この一連の処理を繰り返して目標点までの配線経路を決定する。我々のプログラムでは、この先読み処理を次のように並列に行っている。

最初に始点を含むライン・プロセスは、それと直行する全てのライン・プロセスに対し到達可能な目標点に一番近い点（期待位置）の座標を要求する。

そしてすべての期待位置が返ってきたらその中から最も目標点に近い点を選ぶ。そして目標点に最も近い点を返したライン・プロセスとの交点までを結線し、その交点を始点とし、この一連の処理を繰り返す。

第2は、複数の配線処理を同時並行に実行することによる並列性である。個々の線分プロセスは並列に動作することができるので、探索・配線処理を複数ネットについて、同時に並行して進めることができる。

3 並列化に伴う問題点と対処法

並列に予測線分探索法を行おうとすると二つの問題点が生じた。一つは、デッドロックの問題、もう一つは、複数ネット間の競合の問題である。

3.1 デッドロックの回避

直交する線分のライン・プロセスがほぼ同時期に探索を行なった場合、双方のライン・プロセスは、直交する線分のライン・プロセスからの期待位置計算結果が全て返って来るのを待ち合せて、デッドロックに陥る場合がある。

これを回避するために、探索処理を行なっている間に届くメッセージを監視し、直交するライン・プロセスからの期待位置計算要求メッセージが届いたならば、それに対して探索処理の開始直前の状態に基づいた仮の値を計算して返すという処理を行なっている。また監視中に届いた新たな探索要求メッセージはバッファにためておく。これによりデッドロックを回避している。

3.2 複数ネット間の競合

複数ネットを同時配線すると、異なるネット間で同じ配線領域を取り合う場合が生じることがある。このような状況では、一つのオブジェクトに対するメッセージの順序化により、早いもの勝ちで配線が行われる。後から到着した配線要求は、達成されず、バックトラックが発生する。しかしこのようにしても配線率が低下する可能性は残る。

そこでこのような相互干渉がどの程度配線率を低下させるのかを実験によって調べた。

4 実験および評価

ここでは、以下の3点を明らかにするために二種類の実データを用いて実験を行った。表1に示すように、DATA1は、接続すべき端子が比較的全体に

表 1: データの仕様

データ名	DATA 1	DATA 2
格子規模	262×106	322×389
ネット数	136	71
IO 端子	ネット無し	ネット有り
提供元	日立製作所	NTT
特徴	端子が全体に分散	端子が局所的に集中

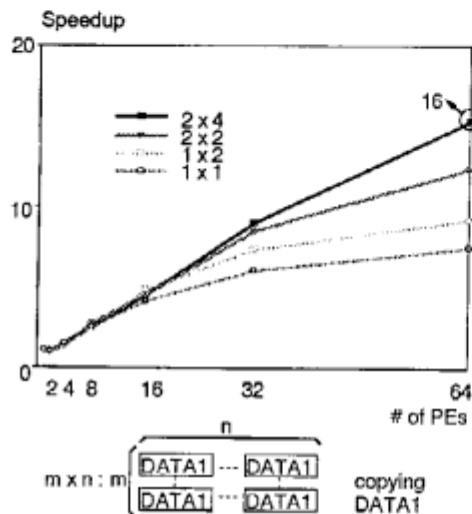


図 3: データ規模と相対性能

分散しているデータであり、DATA 2 は、局所的に端子が集中しているデータである。

4.1 問題の規模と台数効果

一般的に問題の規模が大きくなれば、並列に動作するプロセスの数が増えるので、台数効果も大きくなると考えられる。そこで、我々の並列配線プログラムに関して、データの規模と台数効果との関係を測定した。用いたデータは、DATA 1 を縦方向に n 個、横方向に m 個コピーしたものに対して測定した。コピーしたデータに配線プログラムを適用する場合、その配線可能空間が増えるので、計算量も増えることとなる。それぞれのデータに対する相対性能を一つのグラフとしてまとめたのが、図 3 である。このグラフからデータの規模が大きくなるに従って相対性能が向上することがわかる。今後、どの程度の規模のデータまで台数効果が向上するのかを調べる予定である。

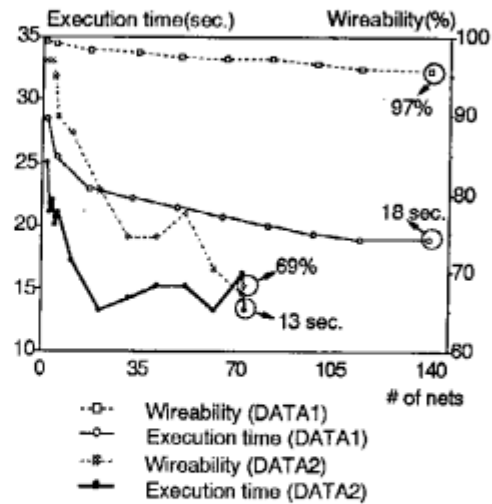


図 4: 配線率と並列度

4.2 配線率と並列度

3.2 で指摘したように、端子が局所的に集中するデータに対しては、多数のネットを同時に配線すると、配線ネット間の競合が起こり配線率が落ちる可能性がある。そこで DATA 1 と DATA 2 に関して、64 台のプロセッサを用い、同時に配線を実行するネット数 N と配線率 W との関係を測定した結果が図 4 である。ここで、二つの縦軸は、それぞれ実行時間と配線率である。また、横軸は、同時に配線を実行するネットの数を示している。すなわち、同時に実行するネット数が 1 ということは、並列処理の効果としては、期待位置の並列計算のみである。この図から、比較的端子が分散しているデータ (DATA 1) に対しては、配線率を余り落とさずにかなりの台数効果が得られることがわかった。しかし、局所的に端子が集中しているデータ (DATA 2) に関しては、複数ネットを同時配線すると配線率が低下してしまうことがわかった。また、このデータに関して、1 台のプロセッサを用いたときの実行時間は、142 秒であった。このことから、期待位置計算部の並列処理による効果は、約 5 倍である。

4.3 汎用計算機との性能比較

NTT LSI 研究所の北沢氏の協力により、IBM 3090/400 (15MIPS) 上に実装された予測線分探索法による配線プログラムを用いて、DATA 2 を配線させた結果、7.45 秒で 100% の配線率であった。Multi-PSI 上での処理速度は、ほぼ汎用大型機と同程度であることがわかった。

5 並列配線プログラムのメモリ効率

5.1 並列配線プログラムの問題点

COMに基づいてプログラム設計をする場合、2章で述べたようにアルゴリズムの並列性設計と実行時の物理的並列性抽出の問題を独立して考えられるのでプログラムの記述が自然にできるという利点がある。しかし、次のようなメモリ効率に関する問題点もある。

並列プログラムを実装する場合に使用されるメモリとしては、プロセスを表現するためのメモリと通信路を表現するためのメモリが必要である。COMに基づく並列プログラム設計においては、一般にオブジェクトの数が増大するとプロセッサ間にわたる通信路も増加するので、通信路を表現するメモリの量も増加する。

我々の並列配線プログラムの場合、ライン・プロセス間のメッセージの仲介をおこなっているマスターライン・プロセスは、直行する全てのマスターライン・プロセスと通信をしなければならない。よって配線格子規模の増加と共にプロセッサ間にわたる通信路も増えるので、現状のシステムでは、配線格子数が500×500程度で通信路を表現するメモリがオーバーフローしてしまう。

5.2 解決方法

前節の問題点を解決するためには、各プロセッサに図5に示すようなディストリビュート・プロセスをおき、プロセッサ間のストリームを管理させればよい。こうすれば、プロセッサ間にわたるストリームの数が常に一定となるので、通信路を表現するメモリはオーバーフローすることはない。ただし、プロセス間通信が一段余計にかかるので、処理速度が低下するとともに、プログラム構造が、もとの並列オブジェクトモデルを素直に反映したものではなくなり、プログラム開発には不利となる。今後、処理速度の低下とプログラムの複雑さの増加の度合いがどの程度なのかを実験により明らかにしていく予定である。

6 まとめ

並列オブジェクトモデルに基づく並列配線手法を分散メモリ型マシンのMulti-PSI上に実現し、LSIの実データを用いた性能評価結果について考察した。

速度向上に関する実験結果では、2台のプロセッサを使用した場合と比較して、64台のプロセッサを用いると約15倍の速度向上を実現した。データ規模

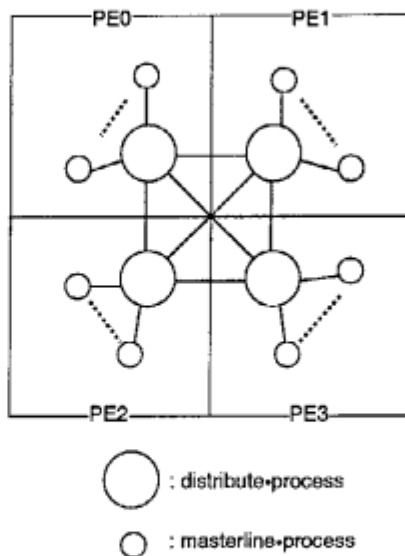


図5: プロセス構造

の増大につれて速度向上も大きくなる傾向を示しており、大規模データでは、さらに高い効率が期待される。

並列度と配線率に関する実験結果では、端子の位置が分散しているデータでは、台数効果及び配線率とも良好な結果を得た。しかし、端子の位置が局所的に集中しているデータに対しては、期待位置の並列計算の効果はあるが、複数ネットを同時配線すると、ネット間の競合により配線率が低下することが確認された。この問題に対しては、効果的な、ネットの同時投入量の制御、未結線ネットの自動再試行などを検討している。

また汎用計算機(IBM3090/400)とMulti-PSIを処理速度に関して比較した結果、ほぼ同程度であることがわかった。

最後に、本プログラムを大規模データに適用できるよう改良するために必要となるメモリ効率とプログラムの書き易さについて検討した。

参考文献

- [1] 伊達 博, 大塚能久, 瀧 和男, “並列オブジェクトモデルに基づくLSI配線プログラム”, 並列処理シンポジウム JSPP'91, 論文集, pp.381-388, 1991.
- [2] 北沢 仁志, “高配線率線分探索の一手法”, 情報処理, 第26巻第11号, pp.1366-1375, 1985.