

岸間 航                      新井浩志                      深澤良彰  
早稲田大学理工学部   千葉工業大学工学部   早稲田大学理工学部

## 1 はじめに

論理回路の大規模化に伴い、設計検証のためのシミュレーションに要する時間が設計のボトルネックとなっている。そこで我々は、要求駆動型のアルゴリズム [1] をもとにした並列論理シミュレーション手法に関する研究を行なっている。一般に用いられている事象駆動型のアルゴリズムでは、回路の入力の論理値変化（事象）が出力方向に伝搬していく過程を逐次評価していく。これに対して要求駆動型では、予め出力ピンと時刻の組で表現される観測要求が与えられ、この観測要求を回路の入力方向に伝搬していく。要求駆動型では回路の全論理値のうち必要な論理値のみを評価するため、事象駆動型よりもシミュレーションのイベント量を削減することが可能である。

我々は要求駆動シミュレーションをメッセージパッシングに基づく並列計算機上で実行するための手法として、多数の観測要求を分割し、各セル（プロセッサ）に分担させる方法を提案してきた。このとき、観測要求を時刻毎に分割するとシミュレーションの効率が良いことが分かっている [2]。しかしこの手法は、組合せ回路に対しては有効であることが確認されているが、順序回路に対しては必ずしも十分なシミュレーション速度が得られていない。

本報告では並列計算機 AP3000 を使い、一つのセル内における要求の伝搬処理を一時中断して、隣接するセルの処理結果が得られるまで保留することによって、順序回路の並列要求駆動シミュレーションを高速化するための手法を提案する。

## 2 順序回路におけるイベントの重複

順序回路を対象として並列要求駆動シミュレーションを実行した場合、各セル間でのシミュレーションイベントの重複が多く、高速化の妨げとなっている。これは、順序回路に含まれるフリップフロップの論理値を決定するのに回路の過去の時刻の論理値が必要となるためである。さらにフィードバックループがある順

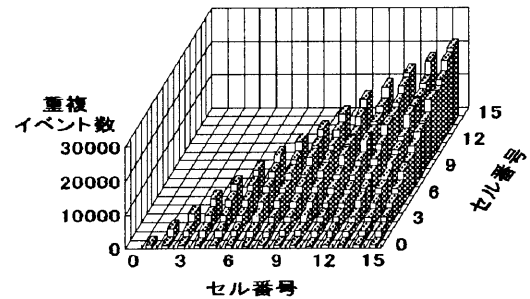


図 1: セル間のイベント重複量  
(保留処理なし、順序回路 s1238)

序回路の要求駆動シミュレーションでは要求の伝搬がループするため、ある時刻の論理値を求めるためにシミュレーションの開始時刻まで遡らなければならない。このため、遅い時刻の観測要求を担当するセルは、早い時刻の観測要求を担当するセルが処理するイベントの大部分を必要とする。

メッセージパッシングに基づく並列計算機である AP3000 の各セルは共有メモリを持たない。このため、遅い時刻の観測要求を担当するセルは早い時刻を担当するセルの評価結果を再利用することができず、シミュレーションを遅くする要因となっている。ISCAS'89 のベンチマーク順序回路 s1238 をシミュレーションした場合の各セル毎のイベントの重複量を図 1 に示す。

## 3 要求伝搬の保留処理

以下では、要求の伝搬を保留し、セル間の通信を利用して他のセルの評価結果を再利用することによって重複するイベントを削減する手法を提案する。この手法を「保留シミュレーション」と呼ぶ。

保留シミュレーションのイメージを図 2 に示す。この図において横軸はシミュレーション時刻を、縦軸は回路の入力側から出力側への信号の流れを示している。本保留シミュレーションでは、各セルにおいて観測要求の伝搬の際に遡れる時刻に制限を設ける。簡単にするため、この制限時刻はセル毎に一定とし、そのセルが担当する観測要求の最初の時刻より早い時刻<sup>†</sup>に設定する。これは、早い方の観測要求がすぐに制限時刻に

<sup>†</sup> 回路により変動するが、s1238 のシミュレーションでは 1 つ手前のセルが担当する観測要求の最初の時刻とした場合に最も効率が良い。

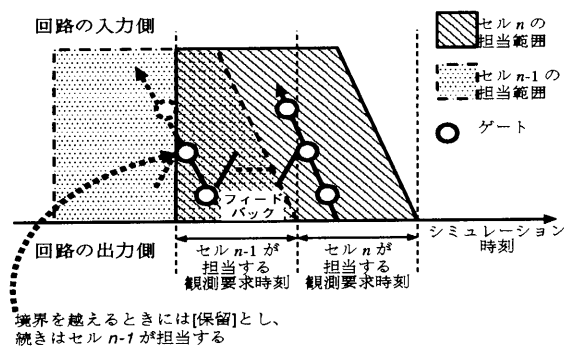


図 2: 「保留シミュレーション」のイメージ

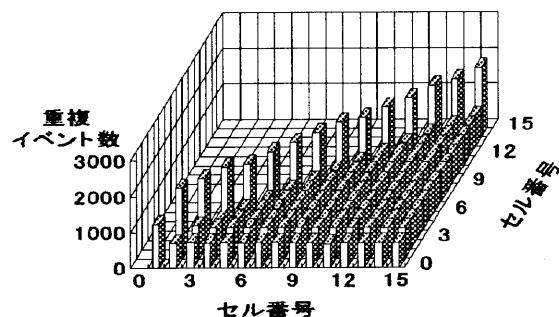


図 3: セル間のイベント重複量  
(保留処理あり、順序回路 s1238)

達してしまわないようにするためである。

そして、シミュレーション中に観測要求の伝搬が制限時刻に達した場合には、その観測要求の伝搬を一旦中止し、そのときの論理値を「保留」とする。同時に、保留となった信号線の名称と時刻を新しい観測要求として1つ前の時刻を担当するセルに送信し、1つ前のセルでシミュレーションの続きを実行させる。また、「保留リスト」を用意しておき、保留となった観測要求を保存しておく。

1つ前のセルでシミュレーション結果が得られたならば、結果を元のセルに送信する。保留リストに残った観測要求について再度シミュレーションを行えば、結果を得ることができる。

通信量を減らすために、各信号線毎に「保留フラグ」を設けておき、保留が発生したときにフラグがセットされていないければ、観測要求を送信してフラグをセットし、結果が返ってきたときにそのフラグをリセットする。フラグがセットされているときは、その信号線でさらなる保留が発生しても観測要求は送信しない。これは、保留が発生する時刻が殆んど一定なので、保留の観測要求に対する1回のシミュレーション結果でその近辺の時刻を含む論理値窓を得られるためである。

#### 4 「保留シミュレーション」の評価

ISCAS'89のベンチマーク順序回路 s1238 における保留シミュレーションの重複イベント数を図3に示す。

保留処理によって、セル間の通信を行わなかった場合(図1)より、セル間の重複イベントの組合せ総数を1/10に削減することができた。依然として隣り合ったセル間ではある程度のイベントの重複が見られるが、これはあるセルで保留となりその1つ前のセルでシミュレーションを行った結果のデータがセル間での重複として数えられているためである。また、各組合せとも重複が0にはならず多少残っているが、これは各ゲート共通のシミュレーション初期値(時刻0で論理値'X')が重複として数えられているためであり、実質的なイ

表 1: 総イベント量と実行時間の比較 (s1238)

テスト パタン長	総イベント量		実行時間(秒)	
	従来	保留	従来	保留
1024	593,982	174,608	7.220	2.272
2048	1,199,747	324,840	26.253	2.360

イベントの重複はほとんど存在しない。

また、セル間の通信を行わなかった場合(従来手法)と、通信を行った場合(保留シミュレーション)のそれぞれについてシミュレーションの総イベント量および実行時間を表1に示す。実行時間において、テストパタン長が1024タイムステップの場合、従来手法に比べて保留シミュレーションでは3倍強、テストパタン長が2048タイムステップの場合は約11倍の高速化を得ることができている。

#### 5 まとめ

順序回路に対する並列要求駆動シミュレーションにおいて、セル間の通信を利用してシミュレーションイベントの重複を削減し、高速化を図ることができた。今後、保留処理のアルゴリズムを洗練するなどしてより効率を高めるとともに、要求駆動アルゴリズムを用いたサイクルベースシミュレーション手法について検討していく予定である。

#### 参考文献

- [1] 中田 登志之: MIMD 型計算機の並列アルゴリズム — 電気系 CAD への応用を中心として —, 情報処理, Vol.33, No.9, pp.1056-1066(1992).
- [2] 張 欣綺, 樋口 寛之, 新井 浩志, 深澤 良彰: 分散処理環境における並列シミュレーションに関する研究, 並列分散処理研究推進機構, カテゴリー 4 科学技術計算応用技術成果報告書, pp.1-1~1-16(1999).