

要求駆動アルゴリズムによる

1 L-6

論理回路の分散並列シミュレーション手法

黄 山* 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

1. はじめに

従来より、論理シミュレーションを並列化するための方法として、論理回路をいくつかの部分回路に分割し、その部分回路毎にセルを割り当てる手法が用いられてきた。また近年では、事象駆動アルゴリズムを基本に、分散離散系シミュレーション手法を用いて、各セルが異なる時間軸で動作するバーチャルタイム法[1]が研究されている。しかし、この方法では、事象が必ずしも時間順に発生するとは限らないため、過去に発生した事象をキャンセルして再評価する。いわゆるロールバックと呼ばれる処理が必要になる。

本報告では、要求駆動型のシミュレーションアルゴリズム[2]を基礎にして、分散環境に適した大規模論理回路のシミュレーション手法を提案する。要求駆動型のアルゴリズムを用いた並列処理により、一定の条件下では、通信をゼロにすることにより、高速化することが可能であることを示す。

2. 要求駆動型アルゴリズム

2.1 要求駆動とは

一般に論理シミュレーションにおいて、すべての信号線のすべての時刻の論理値を知る必要はない。要求駆動型のアルゴリズムは、この事実を利用して、一部の信号線の特定の時刻の論理値を高速に求める方法である。要求駆動型アルゴリズムでは、信号名と時刻から構成される観測要求を、出力ピンから入力ピンに向かって伝搬させる。そして、観測要求が入力ピンまたはすでに論理値が判明している信号線に達した後は、その要求に対する観測結果を入力ピンから出力ピンの

方向に返す。ここで観測結果とは、観測要求時刻を挟んだ開始時刻・終了時刻と、その間の論理値である。

2.2 並列シミュレーションへの適用

今回我々が提案する手法では、回路分割は行わず、回路全体の情報を各セルに持たせる。シミュレーション対象回路は、組合せ回路とする。各セルは、全シミュレーション時間と全出力ピンに対する観測要求を分担して処理する。これにより、セル間の通信や同期を必要としないシミュレーションが可能になる。しかし、一般には並列化によって生じるオーバーヘッドがあるため、n台のセルを用いてn倍の高速化が得られるとは限らない。本シミュレータの効率に影響を与えるオーバーヘッドは主に以下に示す2種類がある。

(1) データ通信によるオーバーヘッド

ホストとセルの間でデータ通信するためのオーバーヘッドが必要となる。ホストからセルへは全回路データと全入力パターンを受け渡す。セルからホストへは各セルが担当する観測要求に対する観測結果を受け渡す。

(2) 重複によるオーバーヘッド

セル間で重複してシミュレーションしてしまうことによるオーバーヘッドであり、主に以下の二種類がある。

a) 同一時刻、異なる出力ピン間の重複

ファンアウトがまったく存在しない回路であれば、シミュレーションの重複は発生しない。しかし、一般にはファンアウトが存在するため、複数の出力ピンから入力ピンに向かって観測要求を伝播させた時に、重複する部分回路が発生する。

b) 異なる時刻、同一出力ピン間の重複

一般の回路には再収束ファンアウトが存在する。再収束の各パス上での遅延時間差があると、同一出力ピンの異なる時刻の観測要求に対して、入力ピン側では同一時刻のシミュレーションが必要になる可能性がある。また、二つの観測要求時刻の間で信号

Parallel and Distributed Logic Simulation based on Demand Driven Algorithm

Shan HUANG*, Hiroshi ARAI**, Yoshiaki FUKAZAWA*

*School of Science and Engineering, Waseda University

**Faculty of Engineering, Chiba Institute of Technology

線に論理値変化がない場合も、同一の観測結果を元にして、重複した評価が行われてしまう。

あるのに対して a) の条件は満足していると考えられるため、a) の方法が有効であると考えられる。

2. 3 オーバーヘッド削減の条件

以下では、2.2 節で述べた各オーバーヘッドを削減するための条件について述べる。

- (1) ホストとセルの間の通信や通信のための処理にかかる時間をなくすることはできないが、シミュレータ本体に与える入力パタンの量が十分大きければ、全シミュレーション時間に占める本オーバーヘッドの割合が小さくなる。
- (2) シミュレーションの重複によるオーバーヘッドは、各セルへの観測要求の割り当て方法と密接に関連している。観測要求の割り当て方法としては、以下の二種類が考えられる。
 - a) 同一時刻の観測要求を同一セルに割り当てることにより、2.2 (2) a) に示した、「同一時刻、異なる出力ピン間の重複」によるオーバーヘッドを軽減できる。例えば、全回路の全タイムステップ (0 から N まで) をシミュレーションする場合、これをセルの数 (P) で割って、 N/P のタイムステップを各セルに担当させる。しかし、シミュレーションの重複を抑えるためには、入力パタンのイベント間隔は N/P より遥かに小さいという条件が必要となる。
 - b) 同一ピンの観測要求を同一セルに割り当てることにより、2.2 (2) b) に示した、「異なる時刻、同一出力ピン間の重複」を軽減できる。しかし、回路のファンアウトによるオーバーヘッドが小さいという条件が必要となる。一般には、b) の条件を満足させることは困難で

3. 評価

API1000 並列計算機を用いて、ISCAS85 ベンチマーク回路 C880 について要求駆動型アルゴリズムの有効性を評価した。テストパターンは、750 タイムステップのもの、1500 タイムステップのものをランダムに生成して用いた。パタンのイベント間隔は 0~10、ゲート遅延は 5 タイムステップとした。2.3 (2) a) の割り当て方法にしたがってタイムステップを各セルに割り当て、観測要求を自動的に生成しながら、シミュレーションを行った。

この結果より、テストパターンが長ければ長いほど、データ通信によるオーバーヘッドの割合が小さくなり、シミュレーションの重複も減るため、効率が高くなっていることが分かる。

4. おわりに

並列計算機上に要求駆動型アルゴリズムを実装し、並列シミュレーションにおける有効性について考察した。現実の回路においては、2.3 節で述べた条件を十分考慮しても、セル間の重複を完全になくすることは不可能である。これからの研究では、さらにセル間の通信と同期を用いて、セル間で重複してシミュレーションされる部分を最少限に抑える手法を検討する予定である。

参考文献

- [1] 中田登志之、MIMD 型計算機の並列アルゴリズム、情報処理、Vol.33, No.9, pp.343-pp.346(1995).
- [2] S.P.Smith, M.R.Mercer, and B.Brock: Demand Driven Simulation, BACKSIM, 24th, ACM/IEEE Design Automation, Conf., pp.181~287(1987).

表1 要求駆動型の評価結果

	入力 パターン	セル数				
		1	2	4	8	16
通信オーバー ヘッド (秒)	パターン1	1.111	0.978	0.937	0.956	1.098
	パターン2	1.398	1.230	1.134	1.143	1.236
全処理時間 (秒)	パターン1	18.222	8.005	3.907	2.417	1.928
	パターン2	45.938	19.009	9.006	5.243	3.457
速度向上率 (倍)	パターン1	-	2.28	4.66	7.54	9.45
	パターン2	-	2.41	5.10	8.76	13.29

パターン1: 750 タイムステップまで (観測要求数: 4352)

パターン2: 1500 タイムステップまで (観測要求数: 7042)