

科学研究費助成事業（科学研究費補助金）研究成果報告書

平成25年6月7日現在

機関番号：15201
 研究種目：基盤研究(C)
 研究期間：2010～2012
 課題番号：22500047
 研究課題名(和文) フォーマル手法およびシミュレーション手法の統合によるハードウェア検証の効率化
 研究課題名(英文) Improving Hardware Verification Efficiency by Fusion of Formal Methods and Simulation
 研究代表者
 浜口 清治 (HAMAGUCHI KIYOHARU)
 島根大学・総合理工学研究科・教授
 研究者番号：80238055

研究成果の概要(和文)：フォーマル手法およびシミュレーション手法はハードウェア検証における2つの方式として実用的に用いられてきている。しかし、この2つの方式を組み合わせる方法については、まだ十分研究されていない。本研究では、カバレッジと呼ばれる量的基準に関して、シミュレーション手法で得られた結果をもとに、達成度が不足している部分に対してフォーマル手法を適用することにより、カバレッジの改善が見られることを、数千ゲート程度のブロックモジュールに対する実験によって示した。

研究成果の概要(英文)：Formal methods and simulation-based methods have been used for hardware verification in practical industrial designs. How to combine these two methods, however, has not been studied extensively yet. In this research, in terms of qualitative verification metrics, we show effectiveness of our new approach, in which based on the results of simulation runs, a formal method is applied for improving the coverage metrics. We also show some experimental results, in which for a block module of several thousands of gates, the proposed method is effective.

交付決定額

(金額単位：円)

| | 直接経費 | 間接経費 | 合計 |
|--------|-----------|---------|-----------|
| 2010年度 | 800,000 | 240,000 | 1,040,000 |
| 2011年度 | 1,500,000 | 450,000 | 1,950,000 |
| 2012年度 | 700,000 | 210,000 | 910,000 |
| 年度 | | | |
| 年度 | | | |
| 総計 | 3,000,000 | 900,000 | 3,900,000 |

研究分野：ハードウェア設計検証技術

科研費の分科・細目：情報学・計算機システム・ネットワーク

キーワード：設計検証技術，フォーマル検証，シミュレーションベース検証，SAT ソルバ

1. 研究開始当初の背景

大規模な集積回路に対する設計検証は、設計工程におけるボトルネックとなっている。これに対して、過去10年ほどの間に種々の方法論や手法が提案・実用化されてきている。代表的な方法論には、カバレッジ駆動検証やアサーション検証がある。近年の検証関連の方法論・技術

カバレッジ駆動検証は、状態カバレッジやトグルカバレッジなど種々の量的基準を定めておき、シミュレーションを行う際、これらの基準が100%に近づくようにテストパターンを増やして、検証を進める方式である。これは、現在設計現場で普通に用いられている方式である。アサーション検証は、設計が満足すべき論理条件(アサーション)を、設

計記述に埋め込んでおく方式である。シミュレーションベースではアサーションをチェックしつつ、さらに、各アサーションの活性度（たとえば、 $A \rightarrow B$ というアサーションで、 A が 1 となった回数）などを評価する。フォーマル検証の場合には、アサーションの成立・不成立を網羅的に調べる。とくにシミュレーションベースのアサーション検証は、標準化アサーション言語 (PSL, SVA など) の普及によって、良く用いられるようになってきている。

個別の手法としては、シミュレーションベースでは制約付きランダムテスト生成、フォーマル検証ベースでは、SAT (充足可能性判定) を利用したモデル検査手法、抽象化再詳細化手法などが開発されている。これらの手法は、RT (レジスタ転送) レベル設計から、C, C++, SystemC などによる高位の設計記述に、対象を拡大する形で研究が進められている。シミュレーション手法とフォーマル手法

シミュレーション手法とフォーマル手法は、しばしば相補的と言われ、小さいモジュールについてはフォーマル手法を用いて、大きなモジュールまたは複数のモジュールを接続した場合には、シミュレーション手法を用いて検証を行うのが良いとされる。しかしながら、フォーマル手法は実際に適用してみなければ、適用可能であるかどうかはわからないことも多く、どのように選択すればよいかについてはあまり明確な基準はない。

シミュレーションとフォーマル検証を組み合わせる方式については、研究開始時点では、ある特定のサイクルまではフォーマル検証を行い、次にランダムシミュレーションを、と次々と切り替えて検証を進めるセミフォーマル手法が提案・商用化されていた。また、各テストパターンに対して、それとよく似た多数のテストパターンをカバーするように、フォーマル検証の手法を使って、通常のシミュレーションを拡張する手法 (スーパーシミュレーション) なども知られていた。

2. 研究の目的

フォーマル検証とシミュレーションベース検証の2つの方式は、相補的と言われているが、実際にどう組み合わせるかについてはまだあまり研究されているとは言えない。本研究の目的は、1つの手法で得られた結果を他方の手法で利用する方法を見いだすことである。さらに、2つの方式をタイトに組み合わせることにより、より網羅性の高いシームレスな検証結果を効率よく得る手法の確立を目指す。

3. 研究の方法

(1) SAT ソルバから得られる制約を利用したランダムシミュレーション

本研究では、まず、フォーマル検証の過程で得られる中間的な情報を、シミュレーション手法に転用する手法について検討・実験を行った。具体的には、フォーマル手法のうち、有界モデル検査と呼ばれる手法を取り上げた。この手法では、内部のエンジンとして SAT ソルバが用いられるが、検証過程で SAT ソルバ内で枝刈りのために生成される「習得節」とよばれる論理式を取り出す。これはすでに探索を終えた空間に対する論理条件の否定を表現したものになっている。これからランダムシミュレーション用に論理制約を生成する。この論理制約はすでに探索を終えている部分を除外する論理条件となっている。この制約のもとでランダムシミュレーションを行うと、フォーマル手法で探索を終えた部分について、除外した形でのシミュレーションとなる。

上記のアイデアによる結果を次の「研究の成果」のセクションで示すが、特定の性質を持つ設計、具体的には到達することが難しいような条件を持つ設計では、効果があることが確認された。しかしながら、実用的な設計からなるベンチマーク回路に対して適用したところ、同様な論理条件を持つ設計では、SAT ソルバ内で生成される習得節が非常に多くまた複雑となり、ランダムシミュレーションを加速できるような制約条件の生成が困難であることが判明した。

(2) SAT ソルバによるトグル回数改善手法

そこで、本研究の後半では、シミュレーション手法とフォーマル手法を組み合わせる新たな方法として、ランダムシミュレーションにおけるカバレッジの改善を目的として、フォーマル手法を用いる手法を別途考案し、これを実装・実験した。ここでは、トグルカバレッジに着目する。信号線が 0 から 1 または 1 から 0 に変化することをトグルと言い、本研究では、トグル回数が少ない信号線について、トグル回数を増加させることをターゲットとした。ランダムシミュレーションを多数回行ってもトグルが発生しにくい信号線は、もともと活性化が難しいと考えられ、そのような信号線がトグルするような入力パターンは、トグル回数の少ない他の信号線も活性化できると期待できるためである。

具体的には、まず、ランダムシミュレーションを行って、各信号線についてトグル回数を計測する。つぎに、トグル回数の少ない信号線を選択する。これらの信号線に対するトグル発生論理条件を求めて、フォーマル手法によって、入力パターンを求める。これを次のステップのシミュレーションにおいて用いる。

上記の手法では SAT ソルバを用いるが、通常の SAT ソルバを複数回利用してパターンの生成を行うと、同一または似通ったような

パターンばかりが生成されて、信号線の活性化に有効でない。そこで、つぎつぎと充足解(設計への入力パターン)を生成するように SAT ソルバを改造し、それまでに生成されている充足解とは 0 と 1 の発生パターンができるだけ異なる解の探索を行うアルゴリズムを採用することとした。

4. 研究成果

(1) SAT ソルバから得られる制約を利用したランダムシミュレーション

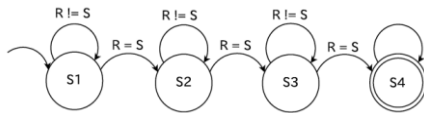


図 1: 実験に用いた順序機械

計算機実験では、図 1 に示した順序機械を用いた。R と S はレジスタ変数であり、複数ビットからなる。まず、この順序機械を順序回路として設計し、Verilog 記述を作成した。次にこの Verilog 記述を変換ツール v12mv によって、状態遷移関数の記述に変換し、さらに CNF(和積標準形論理式)に変換する。与えられたサイクル数分(ここでは 100 サイクル)、状態遷移関数記述の時間展開を行って、設計に対応する CNF 式を構成する。シミュレーションのための入力パターンが満足すべき論理条件をプロパティとして記述して、これも CNF に変換して設計記述に加える。この例では、 $R=S$ を全てのサイクルで満たしつつ、最終状態に到達するというプロパティを用いている。

こうして得られた論理式に対して、SAT ソルバを適用すると、途中結果として論理式つまり習得節を得ることができる。実際に得られる習得節はきわめて多数であるため、節のすべての変数が同じサイクルに属するものに限って用いることとした。得られた習得節をサイクル T に関わるものについて、論理積でつないだものを L^T と書くことにする。サイクルごとに「現在のサイクルが T であれば、 L^T 」という論理条件をランダムシミュレーション時の制約として加えた。

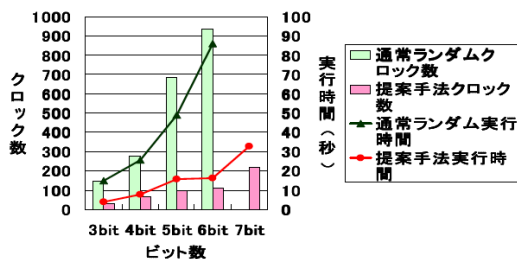


図 2: 図 1 の順序機械に対する結果

SAT ソルバには MiniSAT を用いており、シミュレーションには、商用ツールを用いた。変数 R と S のビット数を変えたときに、上

記のように制約を加えてランダムシミュレーションを行って、プロパティを満たすテストパターンが得られるまでのサイクル数(クロック数)と要した時間を図 2 に示す。これは 100 回行った場合の平均値である。提案手法の計算時間には、SAT ソルバの実行時間も含まれている。提案手法では最大約 3 倍の加速が確認できる。

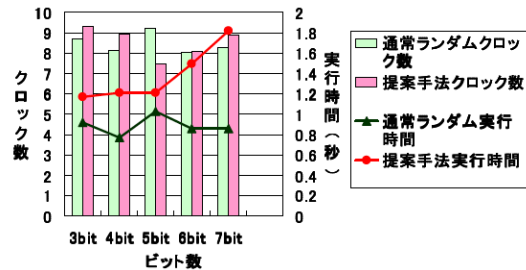


図 3: 遷移条件を変更した場合の実験結果

一方、 $R!=S$ と $R=S$ という条件をそれぞれ、 $R>S$, $R<S$ という条件に取り替えて、同じ実験を行うと、提案手法ではオーバーヘッドが大きくなってランダムシミュレーションの方が高速となる(図 3)。ランダムシミュレーションを行った場合に比較的満足しやすい条件が内部の遷移条件に含まれており、プロパティを満たすテストパターンが、それらの条件を満足すると得られるような場合には、SAT ソルバ適用時のオーバーヘッドが大きくなるため、提案手法は有効ではなくなると考えられる。

(2) SAT ソルバによるトグル回数改善手法

次の手順で実験を行う。(1) 検証対象とする設計に対して、一定時間ランダムシミュレーションを行って、各信号線におけるトグル回数を計測する。(2) トグル回数が少ない信号線を選択する。(3) 選択した信号線のいずれかがトグルする条件を論理式として生成する。(4) 指定されたサイクル数(クロック数)分、設計とトグル条件を時間展開して、CNF 式を生成する。(5) 0 と 1 のパターンができるかぎり異なるような充足解を探索して、入力パターンを作成する。(6) 生成された入力パターンを用いてシミュレーションを行う。

トグルの改善を評価するために次のように指標を導入している。ランダムシミュレーション終了時の信号線 i のトグル回数を I_i 、生成パターン入力後のトグルの増加数を R_i としたとき、信号線 i に関する評価値を $q_i = I_i/R_i$ と定める。これを選択した信号線について加算した値 $S_q = \sum_i q_i$ が評価値である。ここでは、選択する信号線については、トグル回数が平均値以下となる信号線を選択を考えることにし、 $S_{q_{ave}}$ と表記する。

本手法では(5)において、論理式に対してできるだけ異なる充足解を得ることが重要となる。このため、SAT ソルバ(MiniSAT)内の解探索手順を改変して用いている。SAT ソルバは0または1を割り当てる変数を順次選択していくことにより、解の探索を進める。変数を選択する際、各変数 u とそれまでに生成されている充足解の集合について、 u が1(0)であるものの個数 P^u (N^u)を求めて、 $|P^u - N^u|$ が最大となる u を選択し、この値が小さくなるように、0 または 1 を優先的に割り当てるようにする。

計算機実験では、(6)を行う際に、(i)アルゴリズムを改変せずにSAT ソルバで生成したパターン、(ii)上記で述べたようにできるだけ異なるように生成したパターン、(iii)ランダムに生成したパターンを入力して、トグル回数が改善されるかどうか評価値 S_q を使って評価した。

ベンチマーク回路としては、IWLS benchmarks 2005に含まれる ITC99の回路についてb01からb14を利用した。数十ゲートから数千ゲートの回路まで含まれているが、b12とb14はそれぞれ1412ゲート、6728ゲートとなっておりベンチマーク回路の中では大きい。実験ではまず1000クロック分(b12とb14については2000クロック分)のランダムシミュレーションを行った。その後、SAT ソルバを用いて50クロック分(b12,b14については、SAT問題を解くための計算時間が大きくなるため40クロック分)のパターンを20個生成している。ここでトグル条件を生成する際に選択する信号線の数は10とした。

上記の(i)-(iii)について計算機実験を行って $S_{q_{ave}}$ を比較した結果を図4に示す。(iii)のランダム生成の場合の値を1として各回路について比を示している。比較的規模の大きな設計に対しては、SAT ソルバを用いた手法は単純ランダムにパターンを生成する場合に比べてトグル回数を改善できていることが確認できる。

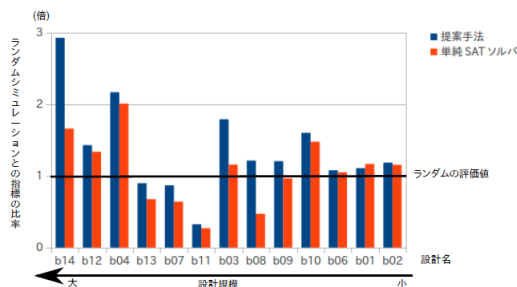


図4：評価指標 $S_{q_{ave}}$ の比較結果

ベンチマーク回路の中で最も規模の大きいb14についてさらに詳しい結果を図5に示している。この図では評価指標を計算する際に用いる信号線について、最初のランダムシミュレーションの際のトグル回数の平均以

下の場合だけではなく、平均の1/8以下の信号線、平均の2/8以下の信号線、... を選んだ場合について評価した結果を示している。ランダムシミュレーションでトグル回数が少なかった信号線について、確かにトグル回数が改善されることが確認できる。また、b14に関しては、ランダムパターン生成のみでは1度もトグルできなかったがトグル可能になった信号線も確認された。

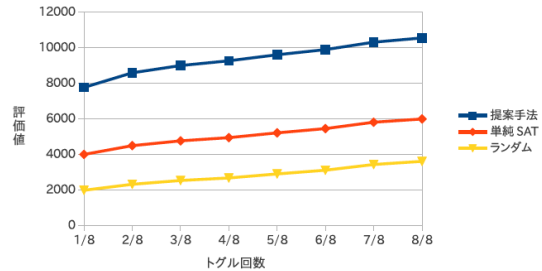


図5：b14に対する実験結果

以上の実験結果についてさらに各回路を解析し、本手法が有効な場合について考察する。効果があったのは、b14,b04,b03であるが、b14では状態遷移の条件が複雑であり、また、b04では状態遷移の条件に8ビットの変数が含まれている。b03は比較的小さな回路(271ゲート)であるが、状態遷移のために2クロックに渡って特定のパターンを入力する必要がある。このため、ランダムシミュレーションでは、トグルを発生しにくい信号線が存在していると考えられる。反対に提案手法の効果かなかったb11,b13では、状態遷移のための条件が単純であり、ランダムシミュレーションであっても、トグルを発生しやすくなっていると考えられる。まとめると、遷移条件が複雑で順序機械としての動作が複雑な回路に対しては本手法は効果を発揮すると考えられる。

計算時間に関しては、ランダムシミュレーションと比べて10倍以上遅くなる場合があり、計算量削減が今後の課題となっている。

5. 主な発表論文等

[学会発表] (計1件)

- ① 石木裕介(発表者), 浜口清治, 若宮直紀 : 動作レベル回路設計記述の等価性判定における複数の論理体系を利用した抽象化, DAシンポジウム2012, 2012年8月29日, 岐阜県下呂市水明館。

6. 研究組織

(1) 研究代表者

浜口 清治 (HAMAGUCHI KIYOHARU)
島根大学・総合理工学研究科・教授
研究者番号：80238055