様式 C-19

科学研究費補助金研究成果報告書

平成２２年５月１４日現在

研究種目：基盤研究(C)
研究期間：2007～2009
課題番号：19500043
研究課題名（和文）　高位ハードウェア設計記述に対するモデル検査手法の研究

研究課題名（英文）　Study on Model Checking for High-Level Hardware Design Descriptions

研究代表者
浜口 清治（HAMAGUCHI KIYOHARU）
大阪大学・大学院情報科学研究科・准教授
研究者番号：80238055

研究成果の概要（和文）：モデル検査は計算機のハードウェアやソフトウェアの正しさを自動的かつ網羅的に検証する技術であり，近年利用がすすんでいる．しかし，複雑な演算を含む設計では，小規模な設計でも扱うことが難しい．本研究では，特にハードウェアの高位設計記述を対象に，複数の論理体系を組み合わせる手法により，不必要な部分の詳細を考慮せずモデル検査を行う方法を開発・実装した．ディジタル信号処理など，従来の手法では扱うことができなかった例に対するモデル検査に成功した．

研究成果の概要（英文）：Model checking is a technology for verifying the correctness of hardware or software designs, which has been widely used. Designs including complex arithmetic operations are hard to handle even for small designs. In this study, in particular, for high-level design descriptions, an approach that combines multiple logics has been developed and implemented, to abstract away unnecessary details. As a result, some digital signal processing designs have become tractable in term of model checking.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
２００７年度	1,500,000	450,000	1,950,000
２００８年度	900,000	270,000	1,170,000
２００９年度	1,000,000	300,000	1,300,000
年度			
年度			
総　計	3,400,000	1,020,000	4,420,000

研究分野：総合領域
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,フォーマル検証,第１階述語論理,モデル検査,高位ハードウェア設計

１．研究開始当初の背景

近年のハードウェア設計では，設計検証が設計工程の半分以上を占めるようになっている．検証コストの低減化のため，アサーションベース検証，カバレッジ指向テスト生成など多くの手法が提唱され実用投入されつつあるが，依然として，検証工程は設計におけるボトルネックである．また，近年，設計および検証効率を改善するため，ハードウェア設計を C や SystemC といったレジスタ転送レベルよりも高位の言語で記述するアプローチが実用的にも取り入れられるよ

うになってきているが，とくに，高位のハードウェア記述に対するフォーマル検証手法については， 必ずしも十分な研究が行われているとは言えない．デジタル信号処理などで複雑な演算を含んでいる場合には，小規模な設計でも従来の手法では，検証が困難である．

モデル検査は，設計対象に要求される論理的な性質(プロパティ)を入力として与えて，それが設計記述によって満足されるかどうかを網羅的にかつ自動的に調べる技術である．本来は，フォーマル検証技術のひとつであるが，シミュレーションベースの検証においても，検証カバレッジを改善するために用いることが可能である．具体的には,手作業またはランダムパタンによっては，生成することが困難な，特定の論理条件を満たす，シムレーション用パターンを見つけるために用いられる．

高位記述に対するモデル検査は,ソフトウェアに対するモテル検査に関係して,マイクロソフト・リサーチの SLAM (あるいは SDV), カーネキーメロン大学の MAGIC を始めとして様々なツールか開発されている．これらのアフローチでは，検証対象のプログラムに対して，何らかの意味で抽象化されたプログラムを作り出し，モデル検査を行う．抽象化により結果の正確さか損なわれている可能性がある場合に限り，順に抽象度を下げて，モデル検査を繰り返す．これらの手法は，基本的にブール式あるいは第 1 階述語論理のサブクラスの論理式など，いずれか１つの充足可能性判定アルゴリズムのみを用いることを前提としている．

２．研究の目的

本研究では高位ハードウェア設計を対象としたモデル検査手法について，特に複雑な演算を含んだ設計を，複数の論理体系を組み合わせて検証を行う手法を開発することを目的とする．

この際，特に設計記述上に含まれる算術演算などを第 1 階述語論理における関数記号によって表現して，可能な限り意味を解釈せずに記号とみなしたまま処理を行う．この方法で正確な結果が得られていない可能性がある場合に限り，整数演算あるいはブール関数のベクトル(ビットベクトル)としての意味を考慮してモデル検査を行う．

３．研究の方法

本研究の当初計画では，設計記述中の繰り返し構造を有限回展開することにより，モデル検査の問題を論理式の充足可能性判定問題に帰着して解決することを目指していた．すなわち，初期状態またはリセット状態から有限のサイクル数のみの範囲を対象とした限定モデル検査を研究対象としていた．

しかしながら，研究成果のセクションで述べるように，より一般に非限定のサイクル数を対象としたモデル検査アルゴリズムが考案できたため，これを実装する形で研究を進めることとした．

本研究では次の２つの手法について検討する．

1．関数記号（および述語記号）に複数の意味を与えておき，可能な限り記号の意味を考慮せずに 処理するアルゴリズムをベースとして利用しつつ，必要に応じて段階的に意味解釈を切り替え て，モデル検査を行う．

2．モデル検査の過程で動的な学習を行い，「同値制約」と呼ぶ記述中の等価関係を抽出して，これを利用して加速を図る

４．研究成果

(1) 項の高さ縮減による第一階述語論理の部分クラスに対する非限定モデル検査アルゴリズムの開発と実装

本研究では，第１階述語論理における関数記号または述語記号を使って，算術演算を抽象化することによって，モデル検査時の計算コストの削減を目指している．しかしながら，一般の第１階述語論理に対する論理式の充足可能性判定問題（あるいは恒真性判定問題）は決定不能であることが知られており，この論理体系を用いてモデル検査アルゴリズムを用いると停止性を保証できない．

これに対して，この論理体系の部分クラスである限量子を含まない等号付き第１階述語論理(以下，EUF 論理と記す)については，一般に論理式の充足可能性判定問題が決定可能であることが知られているが，この決定可能性は，静的(時間的な要素を含まない)論理式を対象にした場合に限られている．順序回路における状態遷移関数に相当する記述を，この EUF 論理を用いて与えて，モデル検査を行う問題はやはり決定不能となることが以前より知られていた．

このため，本研究では当初，限定モデル検査とよばれる，初期状態から入力として与えた有限サイクル数だけに注目して扱う手法を対象とすることを計画していたが，以下に述べる項の高さ縮減という手法によって，近似的ではあるが，非限定のモデル検査アルゴリズムが構築可能であることが明らかとなったため，詳細アルゴリズムの構築と実装を行っている．

ここで近似的というのは，モデル検査の結果，設計が与えられたプロパティに対して正しいと判定されれば，たしかに正しいが，正しいと判定されない場合に，本当に間違っているかどうかについては結論づけることができないことを意味している．この意味では，精密なモデル検査結果は得られないが，実用

的には十分な場合も多く，また，近似に用いているパラメータの値を順次増やしていけば，次第に正しい結果を与えるようになる．

本研究では，まず，インバリアント(不変条件)とよばれる性質に限った非限定のモデル検査アルゴリズムを考えた．不変条件は，すべての時点で成立することを要請する論理的な条件であり，記述内容は制限されるものの，たとえば，２つの異なる記述に対して，特定の時点に達したとき，２つの対応する変数に格納されている値が等しい，といった記述が可能である．

本手法では，EUF 論理で記述された状態遷移関数と，項の高さの上限を定める 0 以上の整数パラメータが与えられる．アルゴリズムはこの状態遷移関数をもとに，状態遷移グラフを生成する．通常の順序回路に対する状態遷移グラフとは異なり，各状態変数には，0,1 の値ではなく，関数記号を使った項（例えば，f(x)，g(y,f(z)) など)が格納される．さらに各状態には，初期状態からその状態へ到達するために必要な，到達可能条件が EUF 論理式として格納される．

このままでは，状態遷移グラフの生成の停止性を保証できない．たとえば，x’=f(x) という状態遷移関数が含まれていると，f(x)，f(f(x))，f(f(f(x)))，…と無限に長い(高さに制限のない)項が出現する．本アルゴリズムの近似手法では，これについて maxh によって項の高さに制限を加える．つまり，部分項を変数に置き換えることによって，出現する項の高さを maxh 以下に押さえる．

その上で，使われている変数は異なるが，形は同じになる項(たとえば，f(x,g(y,z)) と f(w,g(u,v)))を同じと見なして，状態遷移グラフ生成時に状態の併合を順次行う．出現する項の高さが制限されることから，結果として，このグラフ生成は有限時間で停止することが保証できる．生成された各状態で，成立すべき論理条件がチェックされるが，これは SMT ソルバー(Satisfiability Modulo Theory Solver)を用いて判定する．以上が本手法の基本的なアイデアである．

このアイデアを詳細化してアルゴリズムを設計・実装して，数値計算のための二分法(Bisection Method)およびデジタル信号処理のための ADPCM エンコーダの記述に対して適用した．この結果，二分法では約 50 秒，ADPCM エンコーダについては，約 160 秒(CPU 2.4GHz，メモリ 4GB 使用時．C++による実装)で検証結果が得られた．

これらの記述には，入力に依存して実行回数が定まるループが含まれているため，項の高さ縮減を用いない場合は，状態遷移グラフの生成が停止しない．また，EUF 論理を用いない従来手法(ブール論理を用いる)では，含まれている算術演算が複雑なため，計算コストが大きく，モデル検査を行うことができない．

これらは，本研究で開発した手法で始めて検証可能となった例である．

(2) 時相論理式に対するモデル検査アルゴリズムへの拡張

(1) で示した手法では，調べることができる性質はインバリアントに限られており，一般的な時間に関する性質を記述することはできない．例えば，ある特定の入力コマンドが与えられると，定められたサイクルののち，所定の演算結果が指定した変数に格納される，といった性質を直接的に記述することはできなくなる．ここでは，性質の記述言語の作成と(1)のアルゴリズムの拡張を試みた．

(1)のインバリアントは通常の EUF 論理の範囲で記述することができるが，時間に関係する論理条件を記述するためには，時相論理を用いる必要がある．第１階述語論理の体系をベースとした時相論理については，Cyrluk と Narendran による線形時間時相論理(Linear-Time Temporal Logic) に基づく Ground Logic や，Bohn らによる計算木論理(Computational Tree Logic，CTL)を第一階述語論理へ拡張した体系などが提案されてきている．しかし，これをベースにしたモデル検査は，いずれも決定不能となり，停止する手続きを構築することができない．また，時間に関する記述を含まない場合でも決定不能であることから，これらの論理と本研究で導入した近似手法を組み合わせた場合についても，やはり決定不能となることが予想された．このため，本研究ではあらたに，EUF 論理をベースした時相論理を定義することとした．

具体的には，命題論理に，時間に関する演算子(たとえば，「常に」を表現する G や「ーとなるまで…」を表現する …Uー など)を加えた計算木論理 CTL を，さらに拡張して EUF 論理式を記述できるようにして，これを EUF-CTL として定式化した．この論理ではたとえば，ループにはいった直後ある条件が成立していれば，やがてそのループから抜け出る，といった記述を書くことができる．

この EUF-CTL を前提にモデル検査のアルゴリズムを構築した．基本的には，CTL のモデル検査のアルゴリズムの適用を検討したが，(1)の状態遷移グラフ生成手法は，そのままでは利用することができないことが判明した．これは，EUF-CTL 式の部分式として出現する(時相演算子を含まない)EUF 論理式について，状態遷移グラフの各状態での真偽を，一般に一意に定めることができない(真の場合も偽の場合も両方考え得る)ためである．

この問題を解決するため，状態遷移グラフ

の生成時に，EUF-CTL 式中に部分式として含まれた EUF 論理式について，その真偽２つの場合にそれぞれに対応するように，各状態を分裂させる手法を考案し，アルゴリズムとして定式化した．このアルゴリズムが正しい結果を与えるかどうかは自明ではないため，正当性の証明を与えた上で，実装して実験を行った．

デジタル信号処理のための FIR フィルタ設計および前述の二分法について，いくつかの時相論理式を与えて実験を行った結果，FIR フィルタの場合は約 3 秒，二分法の場合は与える時相論理式によって約 200 秒から 340 秒程度の現実的な時間で検証に成功した(CPU 2.4GHz，メモリ 3GB 使用時 C++による実装)．

(1)と同様，これらは従来法では扱うことはできない例となっている．

(3) 複数の論理体系を組み合わせた効率的なモデル検査手法の開発

上記の(1)(2)の手法とともに複数の論理体系を用いて効率的なモデル検査を行う手法について研究を行った．

まず，以前の研究で研究代表者が提案していた「同値制約」の利用について検討を行った．先にあげた ADPCM や二分法の例題について，ランダムシミュレーションを行って各変数についてシンドローム(シミュレーションの結果，その変数に出現する値の列)を調べて，同値となる候補点を見つけ，当該部分の記述を切り出す手法により，確かに同値制約と呼ぶ規則を切り出せることを確認した．

しかしながら，この予備実験により，先行研究での主目標であった等価性判定とは異なり，検証対象記述に同値となる点が少ないことが明らかとなった．また，有効な同値制約は，以下の複数の論理体系の利用によって，自動的に仮定されて処理されることが多いこともわかった．以上より，本研究では同値制約の取り扱いについては，複数の論理体系の組合せ手法によって吸収することとした．

複数の論理体系の利用については，まず組み合わせる論理体系についての要件を検討した．その結果，複数の論理体系を組み合わせる場合には，セマンティクスレベルでみた際に，それぞれ包含関係にある論理体系を選択するのが妥当との結論に達した．

たとえば，f(x,y) という項は，EUF 論理では，x,y は整数，実数を問わず，任意の値をとる変数であり，関数 f は任意の 2 変数関数と解釈されて処理が行われる．一方，ビットベクトル論理では，各変数は決められた長さのビットのベクトルであると解釈され，f が加算に対応する記号であれば，２つのビットベクトルに対する加算関数と解釈されて処理が行われる．つまり，EUF 論理で可能な解釈は，ビットベクトル論理で可能な解釈を包含している，ということができる．

このように包含関係を有する２つの論理体系でモデル検査を行った場合，EUF 論理で評価を行って検証に成功すれば，ビットベクトル論理を用いても検証に成功するということが論理的に帰結できる．さらに，基本的には EUF 論理で解釈を行い，記述の一部の記号のみビットベクトル論理で解釈して，検証が成功すれば，やはり，すべてビットベクトル論理によって解釈した場合にも検証に成功することが論理的に帰結できる．

一般に EUF 論理は計算コストが低く，ビットベクトル論理は計算コストが大きい．与えられた記述は最終的にはビットレベル論理での解釈が与えられているとみることができる．計算コストを抑えるためには，EUF 論理を用いて評価することが望ましいが，EUF 論理は記号についてすべての解釈を許すため，悲観的な結果がでてしまうことが多い（たとえば，f が本来は加算の記号であっても，f(x,y) と f(y,x) を同値とみなせない)．

そこで，できるだけ EUF 論理を用いて，必要なところのみ，より詳細度の高い(EUF 論理に包含されている)論理体系を用いて処理すれば，計算コストを抑えることができて，検証可能な対象を拡大できると考えられる．以上が，本手法の基本的なアイデアである．

以上を踏まえて，包含する論理体系として，EUF 論理，線形算術論理，ビットベクトル論理の３つを選んで組み合わせる手法について検討を行った．具体的には，デジタル信号処理などで用いられている算術演算について，用いる論理体系を次のように定めた．

・+,-,<,>: 線形算術論理
・*,<<,>>: ビットベクトル論理
・/ などその他: EUF 論理

アルゴリズムは，まず，すべての演算について，EUF 論理を用いてモデル検査を行う．失敗した場合には，+,-,<,>については，線形算術論理を用いて，その他の記号は EUF 論理で解釈する．さらに，失敗した場合には，*,<<,>> をビットベクトル論理で解釈するように切り替えて順に評価を行っていく．ここでは，複数の論理体系を混合した論理式を生成して，SMT ソルバーによって評価を行う．

以上のアルゴリズムの実装を行い，ADPCM エンコーダの例について，記述に種々の変更を加えて，(1) のインバリアント検証の実験を行った．表１は，EUF 論理でインバリアント検証が成功する例に，あえて他の論理を組み合わせた場合の結果を示している(CPU 1.5GHz,メモリ 512MB 使用時．C++で実装)．すべての場合に実用的な時間で検証が成功しているが，組み合せる論理体系が多いほど計算時間も大きくなっていることがわかる．

表 1

改変演算	状態数	実行時間(秒)
乗算	159	169.1
シフト演算	159	169.3
乗算+シフト演算	159	175.8

表 2 には，3 つの論理体系を組み合わせて用いて始めて検証が成功する例に対する結果を示している．すべて ADPCM エンコーダの例であるが，それぞれ，乗算，シフト演算，乗算＋シフト演算の記述に改変を加えて，この部分についてはビットベクトル論理で評価しなければ，検証が成功しない．ここではビットベクトル論理のビット幅は 8 ビットとしている．

表 2

論理体系	状態数	実行時間(秒)
EUF	87	12.8
EUF+LA	87	26.6
EUF+LA+BV	87	42.8

なお，この実験では，状態遷移グラフの生成時にインバリアント判定を同時に行い，失敗した時点以降のインバリアントの評価を複数の論理体系を組み合わせる手法で行っている．すなわち，状態遷移グラフの生成を最初からやり直す方法はとっていない．これにより 10-20%計算時間が削減されている．

ここで示した例はすべての記号をビットベクトル論理で解釈すれば，理論上検証が成立する例であるが，計算量が大きくなりすぎるため現実的な時間での検証はできない．衣一方，EUF 論理を用いただけでは，結果が悲観的となって正しい結果を得ることができない．すなわち，本研究で提案した複数の論理体系の組合せによって，始めて検証が可能となる例となっている．

検証が失敗すると，利用する論理体系を増やして再検査を行うことになるが，現状では，算術演算ごとにどの論理体系で解釈するかを固定的に定めている．検証失敗の原因となっている項をみつけることができれば，その部分だけ解釈を変えることによって，さらなる効率化が期待できる．

１つの記号を複数の論理体系で解釈して検証を行う手法はこれまで研究されて来なかったが，本研究で示した結果は，実用的にも有用であることを示唆している．これまでに開発されている種々の効率化手法と組み合わせることで，従来扱うことができなかったより広い範囲の設計検証が可能と考える．

５．主な発表論文等
（研究代表者、研究分担者及び連携研究者には下線）

〔雑誌論文〕（計２件）

① Kiyoharu Hamaguchi, Kazuya Masuda, Toshinobu Kashiwabara, Approximate Model Checking using a Subset of First-Order Logic, IPSJ Transactions on System LSI Design Methodology, 査読有, vol.5, 2011. In printing.
② Hiroaki Shimizu, Kiyoharu Hamaguchi, Toshinobu Kashiwabara, Approximate Invariant Property Checking Using Term-Height Reduction for a Subset of First-Order Logic'', IPSJ Transactions on System LSI Design Methodology, 査読有, vol.4, pp.105-117, 2010.

〔学会発表〕（計４件）

① 増田和也，浜口清治，柏原敏伸，第一階述語論理のサブクラスに対する近似的モデル検査アルゴリズム，情報処理学会研究報告，査読無，2009-SLDM-142-9，2009 年.
② 浜口 清治：SMT ソルバーを利用した近似的な非有界モデル検査アルゴリズムにおける複数の論理体系の組み合わせ手法，組み込みシステムシンポジウム 2009 論文集，査読有，pp.41-48，2009 年.
③ Hiroaki Shimizu, Kiyoharu Hamaguchi, Toshinobu Kashiwabara Approximate Invariant Property Checking Using Term-Height Reduction for a Subset of First-Order Logic, 6th International Conference on Automated Technology for Verification and Analysis, LNCS 5311, 査読有, pp.318-331,2008.
④ 清水博章，浜口清治，柏原敏伸，第一階述語論理のサブクラスに対する項の高さ縮減を用いた不変条件の近似的検証アルゴリズム，情報処理学会研究報告，2007-SLDM-132，査読無，pp.19-29，2007 年.

６．研究組織
(1)研究代表者

浜口 清治（HAMAGUCHI KIYOHARU）
大阪大学・大学院情報科学研究科・准教授
研究者番号：80238055