

NuSMVを用いたオントロジ検証

阿部 雄貴^{†1} 和泉 諭^{†2} 本間 圭^{†3} 富樫 敦^{†3} 高橋 薫^{†1}

^{†1} 仙台高等専門学校 ^{†2} 東北大学電気通信研究所/情報科学研究科

^{†3} 宮城大学大学院事業構想学研究科

1. はじめに

オントロジの記述は人手で行うことが大半のため、その中には構文的矛盾の他、意味的な間違いが含まれている可能性がある。そのため、記述したオントロジの内容を検証する必要がある。主なオントロジの検証方法として、専用のオントロジエディタ (例えば, Protégé[1]の利用) や推論機構 (例えば, Racer[2]の利用) を用いた検証があげられる。これらツールでは、階層関係や包含関係を中心としたオントロジの一般的な整合性や無矛盾性を、記述論理に基づいて検証することができる。しかし、各ドメインに応じた性質や条件を満たしているかどうか等、オントロジの内容を詳細に検証することは難しい。

本稿では、ソフトウェアや組み込みシステムの検証において有効なモデル検査の手法をオントロジの検証に応用する。具体的には、モデル検査ツールの一つであるNuSMV[3]を用いたオントロジ検証システムを提案する。

2. オントロジ検証システム

オントロジ検証システムの概要を図1に示す。図中、各ソフトウェアモジュールは以下の機能を有する。

- **Protégé**: 検証対象となるオントロジは専用のオントロジエディタProtégé[1]を用いて記述する。ProtégéはオントロジをRDF/XML形式で出力する。
- **検証モデル生成**: オントロジの検証をNuSMVで可能とするため、RDF/XML形式のオントロジをNuSMVで扱える検証モデルに対応付ける (変換する)。具体的な方法は次節で与える。
- **NuSMVモデルチェッカ**: 本システムの中核となるモデル検査ツールであり、検証モデルは専用言語で記述され、検証項目はLTL(線形時相論理)またはCTL(計算木論理)の時相論理式で与えられる。
- **検証インタフェース**: モデル検査の実行では、モデルだけでなく、検証項目を時相論理式でいかに表現するかの知識が必要である。しかし、オントロジの検証ユーザがそれらの知識を習得し、使用するの容易

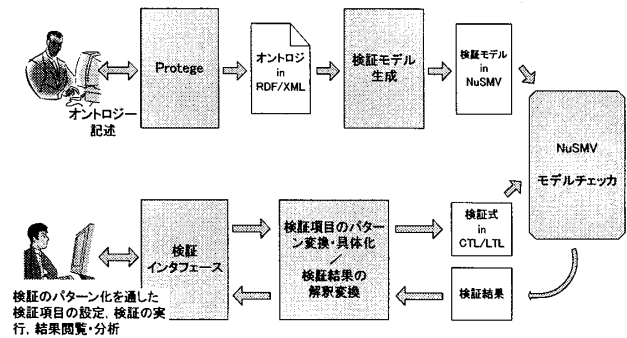


図1 オントロジ検証システムの概要

ではない。そこで、モデル検査の知識がなくても検証できる、ユーザに易しい検証インタフェースを提供する。

- **検証項目のパターン変換・具体化/検証結果の解釈変換**: 検証インタフェースから入力された検証項目をパターン変換等に対応する時相論理式へ自動的に変換・具体化し、NuSMVでの検証を実行可能とする。NuSMVでの検証結果は真偽の値で表され、偽の際には反例が出力される。この場合、検証結果を解釈変換し、オントロジのどこに間違いが存在するかを特定し、検証インタフェースを介して説明できるようにする。

3. オントロジのモデル化

本節では、モデル検査ツール NuSMV で扱うオントロジの検証モデルについて述べる。

【オントロジ】

オントロジはあるドメインにおけるクラスやインスタンス、及びそれらの間の関係 (プロパティ) を体系化したもので、

主語 S, 述語 P, 目的語 O の

トリプル $\langle S, P, O \rangle$ の集合

から成る。この時、S と O はクラスまたはインスタンスを表し、P はプロパティを表す。

【検証モデル】

NuSMV における検証モデルは状態遷移システム (クリプキ構造) であり、システムのある瞬間の振舞いを表す状態 q_1 と、振舞いの時間的な移り変わりによる次の状態 q_2 へのラベル無しの

状態遷移 $q_1 \rightarrow q_2$ の集合

An Ontology Verification using NuSMV

Yuki ABE^{†1}, Satoru IZUMI^{†2}, Kei Homma^{†3}, Atsushi TOGASHI^{†3} and Kaoru TAKAHASHI^{†1}

^{†1} Sendai National College of Technology

^{†2} Research Institute of Electrical Communication / Graduate School of Information Sciences, Tohoku University

^{†3} Graduate School of Project Design, Miyagi University

から成る。この時、ある状態から遷移可能な状態が複数ある場合、次状態への遷移は非決定的である。

オントロジと検証モデルはどちらもグラフ構造として表現できるが、検証モデルはラベル無し状態遷移系であるため、単純な対応付けでは元のオントロジが持つ関連性が崩れてしまうことが考えられる。従って、NuSMV 上でオントロジを扱う際には、いかにしてオントロジが持つクラスやインスタンスの関連性を保持した状態で検証モデルとして表現するかが重要となる。本研究では、以下のような規則でオントロジを検証モデルへ対応付ける(変換する)。

【変換規則】

トリプル $\langle S, P, O \rangle$ を、主語を表す状態 $q1 = (S, P)$ と目的語を表す状態 $q2 = (O, P)$ に分け、遷移 $q1 \rightarrow q2$ により表現する。また、あるトリプルの目的語 O が他のトリプル $\langle S', P', O' \rangle$ の主語 S' となる場合、遷移 $q2 \rightarrow q3$ (但し、 $q3 = \langle S', P' \rangle$) を付加する。(図 2 参照)

以上の規則より、元のオントロジが持つクラスやインスタンスの関連性を保持したまま検証モデルとして表現することができる。

さらに、オントロジの検証を考慮すると、クラスとインスタンスを区別することで、クラスレベルのみの検証や、クラスとインスタンスの関連性(クラスレベルの詳細な制約をインスタンスが満たしているか等)に特化した検証ができると考えられる。そこで、検証モデルの各状態において、その主語と目的語を詳細化するために、状態変数 C, I を追加する。また、主語・目的語の識別変数 $S0$ を導入し、各状態を $(C, I, P, S0)$ と表記する。ここで、 C はクラス、 I はクラス C のインスタンスを意味する。クラスのみ概念空間を表現する場合は $I = \text{“empty”}$ とする。

なお、プロパティ `rdfs:subClassOf` については、本来、下位クラスから上位クラスへ関係として記述されるが、本検証モデルで扱う際は、元のオントロジにおいて上位クラスから下位クラスへ関係 `hasSubClass` へと変換する。これは、オントロジは `owl:Thing` を最上位クラスとし、他の全てのクラスは `owl:Thing` の下位クラスとして構成されることから、検証モデルへ変換した際に、`owl:Thing` を頂点としたツリー構造となり、CTL 式により全状態を網羅した検証を可能とするためである。同様に `rdf:type` も `hasType` へと変換して扱う。

以上から本研究では CTL 式を用いた検証項目を検討しており、CTL 式を扱える代表的なモデル検査ツールである NuSMV を用いることとした。

4. 検証例

本節では家族オントロジを用いたオントロジの検証例を示す。まず、家族オントロジを検証モデルへと変換し、検証項目の設定を行う。図 3 に家族オントロジとその検証モデルの一部を示す。

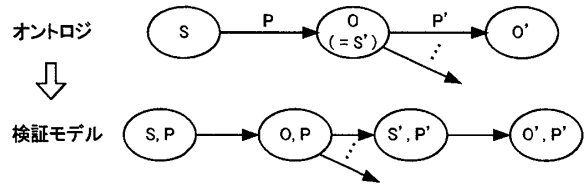


図 2 検証モデルへの変換

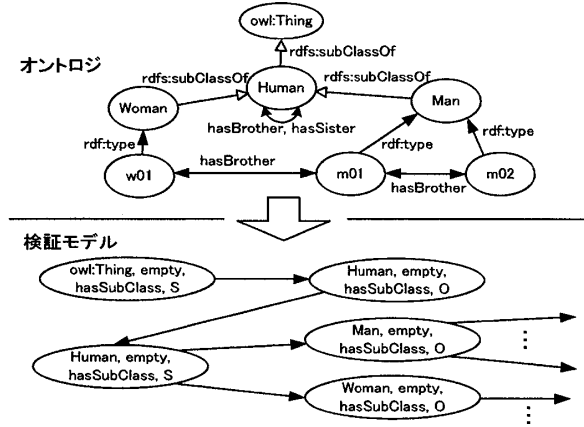


図 3 家族オントロジとその検証モデルの一部

【検証項目】 「兄弟は男性でなければならない」。これを CTL 式で表現したものを以下に示す。

$EF (C=Human \ \& \ I=empty \ \& \ P=hasBrother \ \& \ SO=S$
 $\rightarrow EX (C=Human \ \& \ I=empty \ \& \ P=hasBrother \ \& \ SO=O))$
 $\& (AG ((C=Man \ | \ C=Woman) \ \& \ I \neq empty \ \&$
 $P=hasBrother \ \& \ SO = S$
 $\rightarrow AX (C=man \ \& \ I \neq empty \ \& \ P=hasBrother \ \& \ SO=O)))$

上記の検証項目が満たされていない場合、女性がある男性の兄弟であるという記述が反例として出力される。以上の家族オントロジの検証モデルと検証項目 (CTL 式) を NuSMV に入力し、実際に検証を行った。検証結果として偽と判定され、男性 `m01` が兄弟として女性 `w01` を持つという反例が出力された。

5. まとめ

本稿では、モデル検査の手法を応用したオントロジ検証システムを提案すると同時に、オントロジの検証モデルへの変換と簡単な検証例を示した。

今後は、検証システムの開発を進めるとともに、オントロジ言語 OWL で定義された様々な特性を活かした検証ができるような検証モデルを検討する予定である。

参考文献

[1] The Protégé Ontology Editor and Knowledge Acquisition System, <http://protege.stanford.edu/>.
 [2] Racer Systems GmbH & Co. KG, <http://www.racer-systems.com/>.
 [3] A.Cimatti, et al., “NuSMV: a New Symbolic Model Verifier,” LNCS, Vol.1633, pp.495–499, 1999.