B-015

モデル検査を用いたオントロジの検証

Ontology Verification using Model Checking

和泉 諭 † 本間 圭 ‡ 富樫 敦 ‡ 高橋 薫 ††
Satoru Izumi Kei Homma Atsushi Togashi Kaoru Takahashi

1. はじめに

セマンティック Web において，情報はオントロジを用いて表現される．オントロジの記述は人手で行うことが大半のため，その中には矛盾や意味的に一貫性が保たれてない記述が存在する事が考えられる．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，オントロジエディタ Protégé[1] と推論機構 Racer[2] を用いた検証があげられる．文献 [3] では家族オントロジを例とし，上記ツールを用いたオントロジの階層関係の整合性の検証や推論ルールによる家族関係の補完を行っている．

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，Description Logic 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きめ細かく検証することは困難である．

本稿では，従来は様々なシステムの振る舞いを検証するために用いられるモデル検査ツールを用いて，オントロジの検証を行う．

2. オントロジの検証モデル

本節では，モデル検査ツール上で扱うためのオントロジの検証モデルについて述べる．オントロジの検証モデルを作成する際には，検証モデルとオントロジの特性を考慮する必要がある．

モデル検査ツールにおける検証モデルは状態遷移システムであり，システムのある瞬間の振舞いを表す状態と，振舞いの時間的な移り変わりを表す状態遷移から成る．一方，オントロジは < 主語，述語，目的語 > のトリプルの集合から構成され，主語と目的語はオントロジのクラスまたはインスタンスを表し，述語はオントロジのプロパティを表す．

本稿では以下のように，プロパティを主体としてトリプルを状態と状態遷移に対応づけ，オントロジの検証モデルを定義する．

状態

オントロジの検証モデルにおいて，状態 S はクラスまたはインスタンス CI とプロパティ P の組み合わせで表現する．

$$S =< CI, P >$$

この時，CI は P の主語，または目的語とする．

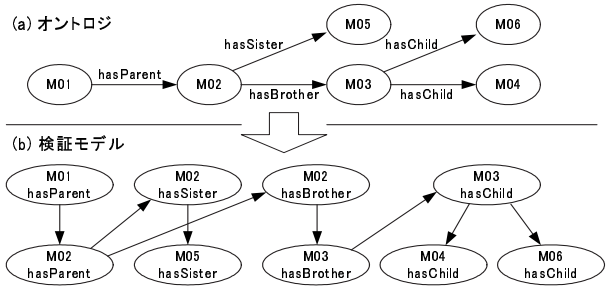


図 1: **オントロジとその検証モデル**

状態遷移

状態から状態への遷移は，ある状態において，CI が P の主語であるか，述語であるかによって，以下のように場合分けする．

(1) CI が P の主語である場合

オントロジ中に CI を主語，P を述語とするトリプル $< CI, P, O >$ が存在するとき，オントロジの検証モデルにおける状態 $S_i =< CI, P >$ から $S_{i+1} =< O, P >$ に遷移する．

(2) CI が P の目的語である場合

オントロジ中に CI を目的語，P を述語とするトリプル $< S, P, CI >$ が存在し，かつ CI を主語，P' を述語とするトリプル $< CI, P', O >$ が存在するとき，オントロジの検証モデルにおいて，状態 $S_i =< CI, P >$ から $S_{i+1} =< CI, P' >$ に遷移する．

図 1 にオントロジとその検証モデルの例を示す．このように，プロパティとクラス（またはインスタンス）の組み合わせで状態を表現することで，次節で示すようなプロパティの特性や意味を考慮した検証が可能となる．

3. 検証例

本節では，文献 [3] で述べられている家族オントロジを例にして，実際にモデル検査ツールを用いて，オントロジの内容の検証を行う．図 2 に例として扱う家族オントロジとその検証モデルを示す．また，モデル検査ツールは NuSMV[4] を用いる．

検証を行う場合，まずオントロジの検証モデルを作成し，NuSMV に入力する．この際，状態を表現するものとして，2 節で述べたクラス（またはインスタンス）とプロパティに加えて，そのクラス（またはインスタ

† 東北大学電気通信研究所/情報科学研究科
Research Institute of Electrical Communication / Graduate School of Information Sciences, Tohoku University
‡ 宮城大学大学院事業構想学研究科
Graduate School of Project Design, Miyagi University
†† 仙台電波工業高等専門学校
Sendai National College of Technology

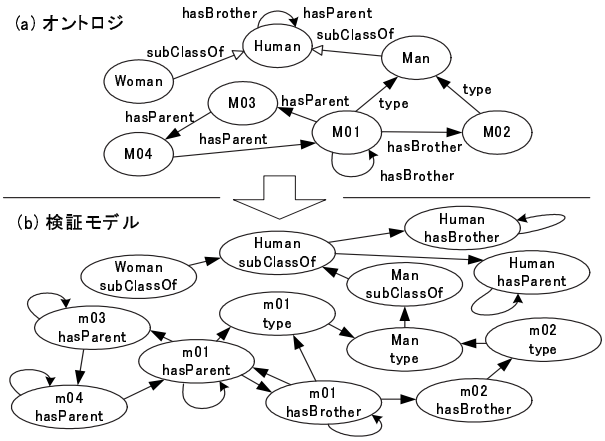


図 2: **家族オントロジと検証モデル**

ンス）がプロパティの主語と述語のどちらに相当するのかを表現する状態変数を加える．そして検証内容を LTL 式（Linear Temporal Logic）で表し，NuSMV に入力して検証を行う．本稿では以下に示す 2 種類の検証を行った．

検証事例 1：自分自身を兄弟としていないか

まず，ある男性 m01 が自分自身を兄弟として記述されていない事を検証する．もし上記の検証内容が満たされていない場合，すなわち m01 が自分自身を兄弟として記述している場合，家族というドメイン知識を考えると，自分自身のことを兄弟とは言わないため，それは矛盾した記述であると言える．上記の検証内容を LTL 式で表現すると以下のようになる．

```
!F ((CI=m01 & P=hasBrother & CIType=subject)
→ X (CI=m01 & P=hasBrother & CIType=object))
```

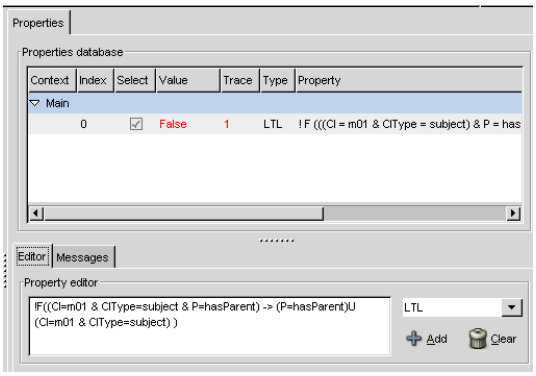
上記 LTL 式を NuSMV に入力し，それを検証モデルが満たしているかを確認した．この場合，検証結果として LTL 式で表現した検証内容をモデルは満たしていないことがわかった．すなわち，m01 が自分自身を兄弟としている記述が存在していることが示された．これは図 2 に示す元のオントロジから明らかである．

検証事例 2：自分自身を祖先としていないか

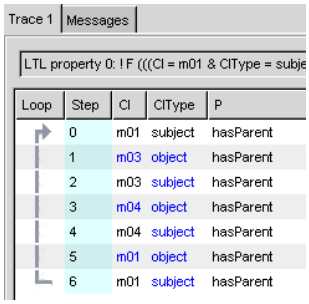
次に，m01 が自分自身を祖先として記述されていない事を検証する．検証内容の LTL 式を下記に示す．

```
!F ((CI=m01 & P=hasParent & CIType=subject)
→ (P=hasParent) U (CI=m01 & CIType=subject))
```

図 3 に上記内容の検証結果を示す．図 3(a) では，LTL 式を入力し，その内容をモデルが満たしているかを判定した結果を示している．判定結果として LTL 式で表現した検証内容をモデルは満たしていない（`False`），すなわち，オントロジに m01 が自分自身を祖先としている記述が存在していることが示された．図 3(b) では，その反例を出力している．同図から m01 は m03 を親として，m03 は m04 を親として，さらに m04 は m01 を親と記述していることが確認できた．



(a) **LTL 式の入力とその判定**



(b) **反例の表示**

図 3: **自分自身を祖先としていないかの検証例**

4. おわりに

本稿では，オントロジの内容についてモデル検査ツールで検証を行うため，オントロジの検証モデルを定義した．さらに家族オントロジを例として，実際にモデル検査ツールを用いてオントロジの内容の検証を行った．モデル検査ツールを用いることで，従来のオントロジ検証ツールでは検証が困難であった，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，各ドメインの意味に反していないかどうかを検証することができた．

今後は，オントロジの検証モデルを詳細化し，オントロジ言語 OWL で定義された様々な語彙の特性を考慮した検証事例について検討する予定である．

参考文献

[1] The Protégé Ontology Editor and Knowledge Acquisiton System, `http://protege.stanford.edu/`.

[2] Racer Systems GmbH & Co. KG, `http://www.racer-systems.com/`.

[3] C. Golbreich, “Combining Rule and Ontology Reasoners for Semantic Web, ” LNCS, Vol.3323, pp.6–22, 2004.

[4] A. Cimatti et al., “NuSMV: A New Symbolic Model Verifier, ” LNCS Vol.1633, pp.495–499, 1999.