

온톨로지 디버깅을 위한 MEXS 추출 및 저장 기법

김제민[○] 박영택
 송실대학교 컴퓨터 학과
kimjemins@hotmail.com, park@ssu.ac.kr

MEXS Extracting and Storing for Ontology Debugging

Je-Min Kim[○] Young-Tack Park
 Dept of Computer Science, Soongsil University

현재 온톨로지를 구축하는데 있어 OWL을 온톨로지 표현 언어로 많이 사용하는 추세이다. OWL 온톨로지의 내제된 정보(클래스간의 계층구조, 인스턴스의 정확한 타입)를 추론하기 위해, 현재 많은 온톨로지 추론엔진이 개발되어지고 있다. 그러나 대부분의 온톨로지 추론 엔진들은 단순히 추론 결과만 명시할 뿐, 그 과정을 표현하지는 않는다. 따라서 본 논문에서는 논리적으로 정당하지 못한 온톨로지를 디버깅 하기위한 MEXS(Minimum Expression Axiom Set) 추출과 저장에 대한 기법을 제안한다. MEXS를 추출하기 위해서는 온톨로지 내에서 논리적인 오류를 유발하는 Axiom들을 찾아내는 방법은 매우 중요하다고 할 수 있다. 이에 본 논문은 두 가지 부분에 초점을 맞추어 연구를 진행하였다. 첫 번째, 논리적으로 정당하지 못한 온톨로지가 주어졌을 때, 논리적 오류를 유발하는 핵심 Axiom을 찾아내고, 이와 연관이 되는 Axiom들을 찾아낸다. 두 번째, 논리적으로 정당하지 못한 온톨로지를 디버깅하기 위한 MEXS를 구성한다.

본 논문에서 제안하는 방식은 Tableau 알고리즘 기반의 MEXS 구성 기법과 이를 좀 더 최적화한 Hitting Set Tree 기반의 MEXS 구성 기법이다. Tableau 알고리즘 기반의 MEXS 구성 기법은 SHOIN 수준의 서술 논리 추론이 가능한 확장 Tableau 알고리즘을 기반으로 하고 있으며, Hitting Set Tree 기반의 MEXS 구성은 특히 대용량 온톨로지의 다수의 논리적인 오류를 일으키는 AXIOM을 효율적으로 찾아내기 위해 Hitting Set Tree를 적용한 방법이다. Tableau 알고리즘은 satisfiability 테스트를 실험적으로 Tractable하게 보여줄 수 있음을 증명하였고, 현재 많은 서술 논리 기반의 온톨로지 추론 기술은 Tableaux 알고리즘을 기반으로 하고 있다. 따라서 제안하고자 하는 MEXS 추출에 바탕이 되는 온톨로지 추론 방법에는 Tableaux 알고리즘 기반의 서술 논리 추론을 적용되었으며, 이를 기반으로 추론 도중에 해당 온톨로지에 논리적인 오류를 발생하는 Axiom, 또는 $C \sqcap \neg D$ 가 unsatisfiable하는데 원인을 제공하는 Axiom을 추출한다.

MEXS는 온톨로지 내에서 논리적 오류를 유발하는 Axiom들의 Justification의 집합이다. 논리적 오류가 있는 클래스가 존재할 때, 이 클래스는 Tableau 알고리즘의 확장 규칙의 반복적인 적용을 통해서 완전 그래프를 그려 나가다가, 결국 Crash를 발생시킨다. Justification은 Tableau 알고리즘을 적용했을 때의 초기 Axiom부터 Crash가 발생한 순간까지의 Tableau 알고리즘 진행 따른 Axiom의 상태를 기록하고 있다. 이를 위해 온톨로지에 존재하는 모든 Axiom에는 ID-number를 할당한다. 예를 들어 다음 과 같은 Axiom이 주어졌을 때,

1. $A \sqsubseteq B \wedge \exists R.E$
2. $B \sqsubseteq C \wedge D$
3. $C \sqsubseteq (\geq 1.R) \wedge E$
4. $B \sqsubseteq E$

A는 Tableau 알고리즘을 적용했을 때의 초기 Axiom에 해당하므로 [0]이라는 Justification을 갖게 된다. Unfold 규칙과 \wedge 규칙에 따라 그래프 노드에는 B 와 $\exists R.E$ Axiom이 추가되며, 각각 Axiom에는 해당 ID-number가 추가된다. 이때 현재 Axiom을 유도하는 이전 Axiom의 ID-number가 [0]이므로, 이전 ID-number는 삭제되고, 따라서 B 와 $\exists R.E$ Axiom의 Justification은 [1]이 된다. 다시 Unfold 규칙과 \wedge 규칙에 따라 그래프 노드에는 C 와 D Axiom이 추가되며, 이때 현재 Axiom을 유도하는 이전 Axiom의 ID-number가 [1]이므로, 각 Axiom에는 해당 ID-number가 추가된다. 따라서 C 와 D Axiom의 Justification은 [1.2]가 된다. 이러한 방식으로 Justification을 추가해 나감으로써, Chash가 유발되었을 때, 온톨로지 내에서 논리적 오류를 유발하는 Axiom들을 찾아낼 수 있으며, 이러한 Justification을 기반으로, MEXS를 구성할 수 있게 된다.

현재 발표된 Tableau 알고리즘은 Non-deterministic을 고려하지 않는다. 즉 알고리즘에 따라 그래프를 구축하면 서, Crash가 발생하면 알고리즘 진행을 멈추고, 현재 온톨로지가 논리적 오류가 있음을 메시지로 출력한다. 그러나 MEXS 안에는 온톨로지 내에서 논리적 오류를 일으키는 모든 Axiom들의 Justification을 저장하고 있어야 하므로 Clash가 발생하면, 현재 Clash를 유발하는 Axiom들의 Justification을 MEXS에 저장하고, Clash를 유발하는 그래프 노드의 또 다른 Non-deterministic을 확인하여, 가능한 모든 논리적 오류들을 유발하는 Axiom들의 Justification을 찾아내야 한다. 따라서 정확한 MEXS를 구성하기 위해서는 기존의 Tableau 알고리즘의 각각의 규칙들과 종료 조건을 수정할 필요가 있다. 수정된 종료조건은 다음과 같다.

- Tableau 알고리즘 진행시 모든 Non-deterministic을 고려했으며, 어떠한 Crash도 발생하지 않을 경우, 더 이상 적용할 Tableau 규칙이 없다면 종료한다.
- Tableau 알고리즘 진행시 모든 Non-deterministic을 고려하지 않았으며, 어떠한 Crash도 발생하지 않을 경우, 현재 기록된 최근 branching-point부터 다시 그래프를 구축한다.
- Tableau 알고리즘 진행시 모든 Non-deterministic을 고려하지 않았으며, Crash가 발생했을 경우, Crash를 유발한 Justification을 MEXS에 기록하고, 현재 기록된 최근 branching-point부터 다시 그래프를 구축한다.

Tableau 알고리즘 기반의 MEXS 구성 기법은 기존의 Tableau 알고리즘을 그대로 사용할 수 있기 때문에, 대부분의 Tableau 알고리즘 기반의 온톨로지 추론엔진에 쉽게 적용이 가능하다는 장점이 있다. 반면에 MEXS를 구성함에 있어서는 모든 Non-deterministic을 고려하기 때문에 탐색 그래프의 모든 거의 모든 노드들을 탐색하게 된다. Hitting Set Tree 기반의 MEXS 구성 기법에서 Justification을 생성하는 방법은 수정된 Tableau 알고리즘을 그대로 적용한다. 일단 Justification이 하나 구해지면(Non-deterministic을 고려하지 않음), Justification을 통해서 임의의 Axiom을 하나 제거한 후, 다시 온톨로지의 논리적인 정당성을 찾아낸다. 이때 Crash가 발생되면, Crash를 유발하는 Axiom들로 구성된 Justification은 새로운 노드로 생성한다. 즉 초기 Justification은 Hitting Set Tree의 루트 노드가 되고, 임의로 삭제된 Axiom은 edge가 되며, 새로 구성된 Justification은 하위 노드가 된다. 이러한 방법으로 트리를 구성해 나가다가, 더 이상 Crash를 유발되지 않거나, 이미 이전에 탐색한 적이 있던 경로면 하위노드 생성을 중단하고, 상위 노드로 Back-Tracking 한다. MEXS는 트리가 구축되면서 자연스럽게 생성되는데 이때 이전에 한번 저장되었던 Justification(하위 노드)은 다시 저장 되지 않는다,

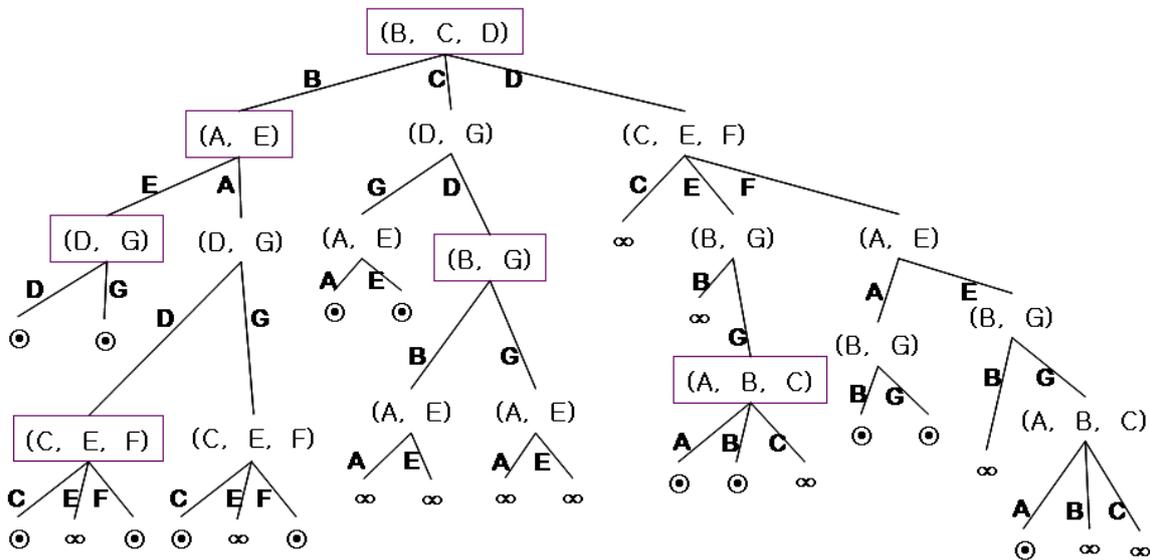


그림 1. Hitting Set Tree를 이용한 MEXS의 구성

MEXS(Minimum Expression Axiom Set) 추출과 저장 기법은 톨로지 내에서 논리적 오류를 유발하는 부분들만을 정확하게 지적해 주기 때문에 온톨로지 구축 자가 대용량 온톨로지를 디버깅하기 위해 많은 도움을 줄 수 있다. 또한 클래스간의 Subsumption 추론 과정을 표현하는데 좋은 자료가 된다. 현재 제안된 방법을 실험하기 위해서 기존 온톨로지 추론 엔진인 Pellet을 사용하였기 때문에 실험결과는 많은 부분에서 Pellet의 성능에 의존된다. 따라서 향후에는, 좀 더 MEXS를 효과적으로 구성하는 추론엔진개발과 더불어 MEXS기반의 온톨로지 디버깅 Planning 기법에 대한 연구가 진행될 것이다.