



온톨로지 기반의 보일러 섯다운 절차 생성 : 지식표현 및 훈련시나리오 활용

박명남 · 김태옥 · 이봉우* · †신동일

명지대학교 화학공학과, *한국소방산업기술원

(2017년 8월 7일 접수, 2017년 8월 24일 수정, 2017년 8월 25일 채택)

An Ontology-based Generation of Operating Procedures for Boiler Shutdown : Knowledge Representation and Application to Operator Training

Myeongnam Park · Tae-Ok Kim · Bongwoo Lee* · †Dongil Shin

Department of Chemical Engineering, Myongji University, Yongin 17058, Korea

*Department of Hazmat Technical Standards, Korea Fire Institute, Yongin 17088, Korea

(Received August 7, 2017; Revised August 24, 2017; Accepted August 25, 2017)

요약

대규모 플랜트에서 조업자 안전훈련 모델의 전제조건은 조업에 관련된 다양한 위험의 상세분석 및 지식표현으로 얻어진 운영절차의 범용성과 정확성이다. 본 연구에서는 조업절차의 생성을 위해 인공지능 플래닝 기법을 고려하여 조업자의 일반행위와 조치행위 그리고 기술용어 등을 분류하고, 지식의 공유 및 재사용을 고려하여 플랜트의 운영과정과 관련된 조업행위 및 용어의 확장을 지식표현 온톨로지 형태로 정의하였다. 또한 조업의 일반적인 행위의 구체화를 위해 Hierarchical Task Network (HTN)기반의 행위계획을 적용하여 목표와 실행이 가능한 수준까지 분할하여 여러 상황에 따른 절차를 생성하도록 설계하였다. 이후 실제 보일러 설비의 사례연구를 통해 조업조건과 운전상태 그리고 장치들 간의 운전목적에 따라 구성설비의 역할을 분류하고, 비상정지절차를 생성하였으며, 제안한 방법의 실제 플랜트 적용 가능성을 확인하였다. 체계적인 지식표현에 기초한 지식베이스 구축은 일반적인 플랜트 운영절차 및 조업자 안전훈련 시나리오의 생성에도 활용이 가능할 것이며, 향후 자동생성 등에도 활용될 수 있을 것으로 판단된다.

Abstract - The preconditions of the usefulness of an operator safety training model in large plants are the versatility and accuracy of operational procedures, obtained by detailed analysis of the various types of risks associated with the operation, and the systematic representation of knowledge. In this study, we consider the artificial intelligence planning method for the generation of operation procedures; classify them into general actions, actions and technical terms of the operator; and take into account the sharing and reuse of knowledge, defining a knowledge expression ontology. In order to expand and extend the general operations of the operation, we apply a Hierarchical Task Network (HTN). Actual boiler plant case studies are classified according to operating conditions, states and operating objectives between the units, and general emergency shutdown procedures are created to confirm the applicability of the proposed method. These results based on systematic knowledge representation can be easily applied to general plant operation procedures and operator safety training scenarios and will be used for automatic generation of safety training scenarios.

Key words : operating procedure, ontology, process safety, boiler shutdown, ai planning

†Corresponding author: dongil@mju.ac.kr

Copyright © 2017 by The Korean Institute of Gas

I. 서론

고도화된 장치산업에는 정상 및 비정상 운전과 관련하여 많은 운전관련 절차들이 존재한다. 짧게는 수분에서 수일간 이뤄지는 일련의 절차들은 수많은 장치들과 여러 단위공정들이 순차적으로 연결되어 여러 개의 단위작업들을 가지고 있다. 이러한 장치산업의 안전한 조작을 위해 조업절차의 합성방법은 플랜트운전 관리기술을 확보 측면에서 대안이 되어 왔으며, 전문가 시스템을 통해 구현하려는 노력이 있어 왔다. 최근에는 인공지능 기술의 발달로, 추론을 사용한 정성적 및 정량적 지식 베이스를 통해 지식표현기법을 이용한 연구가 시도되고 있으며, 온톨로지 지식베이스를 이용하여 기계의 가동성을 높이고, 1차 논리에 기반을 둔 추론을 통한 지식표현으로 조업절차의 각 도메인에 적합한 표준방법론을 정립하고, 유효한 조업절차의 자동생성이 요구되는 경향을 보이고 있다. 이전 연구에서 운전절차서는 주로 조업절차의 합성(Operating Procedure Synthesis, OPS)방법으로 프로세스 영역에서 수행되어 왔다.

Aylett 등[1]은 화학공장의 물질흐름을 고려하여 밸브 시퀀스를 위한 플래너의 사용과 통합을 제안하고, 운영절차 합성에 AI 플래닝 도구를 적용하였다. 또한 Hou 등[2]은 조업자의 운전지식과 추론과정을 일반화하고, 계층분류하여 보일러의 섀다운 절차생성 방법을 고려하였으며, Lee 등[3]은 운전절차를 생성하기 위해 대상플랜트의 경험적 지식을 일반화하고, 공정의 동적 변화를 휴리스틱 함수로 모사하여 운전목표를 순서에 따라 하위 트리로 구분하는 방법을 제안했다. 이러한 운전절차 합성방

법은 공정의 각 구성요소 사이의 관계가 정의된 지식베이스를 이용하여 지식의 재사용을 추구하고, 플랜트 구조의 변경 등에 의한 절차서의 수정에 걸리는 시간을 줄이는 방향으로 발전하여 왔다. 그러나 여러 운전목표 사이에 상호작용을 보이는 복잡한 공정, 중복문제의 분할이 복잡해질수록 지수적으로 증가하는 조합폭발(combinatorial explosion) 문제를 수반하여 해답을 발견 못하는 단점을 가지고 있다.

본 연구에서는 온톨로지 기반의 보일러 비상정지 절차에 관한 지식표현을 인공지능 플래닝 중에 하나인 HTN 플래닝(Hierarchical Task Network Planning) 방법을 적용하여 task나 activity를 수행하기 위해 실행 가능한 subtask로 분할, 운전절차에 포함된 지식표현 및 범위의 확장, 그리고 재활용이 가능한 온톨로지 기반의 섀다운 절차를 생성하고, 구축 방법과 그에 따른 의미를 논하였다.

II. 이론

2.1 온톨로지

온톨로지 엔지니어링은 유사한 개체와 각 개체 간의 관계를 제공하는 개체의 속성 또는 클래스를 정의하여 체계적인 도메인에 대한 형식적인 개념화를 말한다[4]. 온톨로지는 개념화된 관심 도메인에 대한 정형화된 항목으로, 컴퓨터의 추론기능을 허용하며 시스템 통합 프로세스를 간소화하게 하는 유망한 기술이다. 온톨로지는 지능형 시스템 개발에 도움이 되며, 1990년대부터 프레임과 1차 로직(Cyc 온톨로지, Ontolingua 온톨로지[5])을 기반으로 하는 AI 모델링 기술로 이용되었고, 최근에는 표현논리를 기반으로 한 다른 형태의 지식 표현 기술이

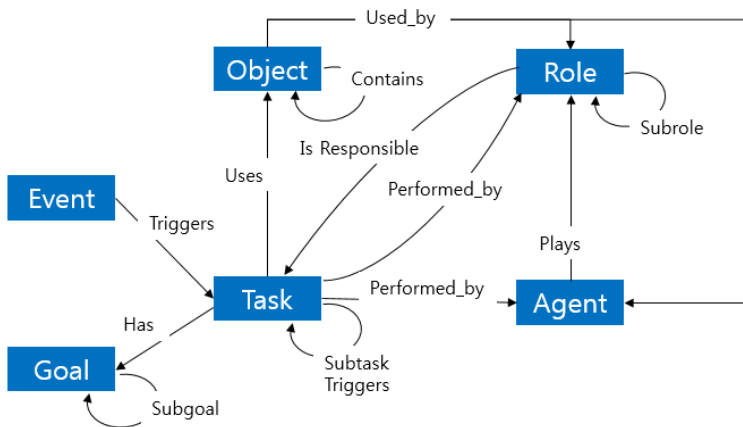


Fig. 1. An ontology for task analysis[10].

온톨로지 구축하는데 개발되었으며, OIL(Ontology Inference Layer), DAML(DARPA Agent Markup Language)+OIL[6]과 OWL(Web Ontology Language)[7]과 같은 새로운 표현논리 언어[8]가 시멘틱 웹 언어로 등장하였다[9].

분류체계는 각 하위 집합의 공통된 특성을 기반으로 하는 도메인 하위집합의 그룹으로 되었고, 특성은 온톨로지에 의해 지정되며, 존재론적 특성의 선택된 하위 집합에 의해 결정되는 도메인에 대해 하나 이상의 분류가 있을 수 있다.

Fig. 1과 같이 온톨로지는 통합 지식의 재사용성, 데이터 공유, 상호 운용성, 문맥 인식 및 의미론적 데이터 마이닝 같은 문제를 해결하는데 사용된다. Welie 등[10]은 task와 subtask의 분해를 통해 개체간의 상호관계를 분석하였고, task를 특정목표에 도달하기 위해 에이전트가 수행하는 활동으로 규정하였으며, goal은 task의 환경 또는 시스템이 원하는 상태로 보았다. role은 에이전트가 수행하는 의미있는 task들의 모음이며, object는 물리적 또는 비-물리적 독립체, agent는 동작하는 개체, event는 task 환경의 상태변화로 분석하였다. 이러한 온톨로지 구축을 위한 표현논리 언어 중 RDF(Resource Description Framework) 또는 OWL 등을 활용하여 온톨로지 개발이 가능하며, OWL은 정보를 사람에게 제공하는 대신 정보의 내용을 처리해야하는 온톨로지 기반 응용 프로그램으로 설계되어 있다. 본 연구에서는 대표적으로 Protege의 온톨로지 에디터를 사용한다.

2.2 온톨로지에서 RDF(S)/OWL의 지식표현

RDF[11]는 Resource Description Framework의 약자이다. W3C가 제정한 웹 리소스를 기술하는 메타 데이터를 생성하기 위한 국제 규약 중 하나인 RDF 데이터 모델은 의미론적 네트워크 형식론을 따르며, 3가지 요소로 구성된다. 즉, 자원(RDF에 의해 표현되는 모든 유형의 데이터), 성질(자원을 설명하는데 사용되는 속성이나 관계의 정의) 그리고

서술(특정 리소스의 성질에 값을 할당)로 구성된다. 이때, RDF KR(knowledge representation) 온톨로지는 클래스와 클래스별 특성 및 하나의 인스턴스를 포함하며, 주요 구성요소의 정의는 다음과 같다 [12].

- Class rdf:Statement : 주어, 성질 및 객체를 포함하는 3가지 클래스를 정의
- Class rdf:Property : 클래스의 속성 정의
- Class rdf:Bag, rdf:Seq and rdf : Alt는 집단의 클래스를 정의하며, 각기 무순, 정렬, 또는 하나의 선택되어야 하는 자원의 집합을 나타낸다.
- Class rdf:List:XMLLiteral : 잘 구성된 문자 그대로의 XML값의 클래스를 정의하는 데이터 유형
- Properties rdf:predicate, rdf:subject 및 rdf:objects : 각각의 클래스 속성, 주제의 자원 및 객체 자원을 정의
- Properties rdf:type : 자원이 속한 클래스 정의
- Properties rdf:value : 데이터 값이 구조화된 자원인 경우 일반적으로 문자열인 속성의 값 정의.

Fig. 2는 RDF KR 온톨로지의 클래스 분류를 나타낸다. KR 온톨로지에서 가장 중요한 것은 rdfs:Resource는 RDF 문, RDFS 컨테이너, RDFS 클래스, RDF 속성 및 RDFS 원 문자로 분류하며, rdf:Bag, rdf:Seq 및 rdf:Alt 클래스는 rdfs:Container의 하위 클래스이다.

RDF 구문은 things, property 및 value로 표현된다. 이때, thing은 웹 리소스이고, property는 속성, value는 속성 값으로 나타내며, RDF 표현 구조는 subject, predicate, object로 구성된 triple 형태로 표현되어 웹 페이지에서 subject는 URIs(uniform resource identifier)가 되고, predicate는 uri설명, object는 웹 페이지의 제목으로 해석될 수 있으며, 제약 사항은 RDF schema에 명시된다.

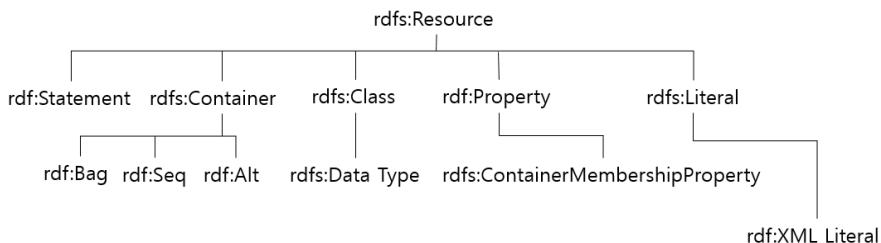


Fig. 2. Class taxonomy of the RDF(S) KR ontology [12].

```

<rdfs:Class rdf:ID="OperatingProcedure">
  <rdfs:comment> Boiler operating procedure by abnormal</rdfs:comment>
  <rdfs:subClassOf rdf:resource="#Boiler"/>
</rdfs:Class>
<rdf:Property rdf:ID="ControlRoom">
  <rdfs:domain rdf:resource="#Boiler"/>
  <rdfs:range rdf:resource="#BoilerShutdownProcedure"/>
</rdf:Property>
    
```

Fig. 3. Generated boiler ontology RDF(S) file from Protege editor(adapted from Asuncion et al.[12]).

```

<owl:Class rdf:ID="OperatingProcedure"/>
<owl:Class rdf:ID="Boiler">
  <rdfs:subClassOf rdf:resource="#OperatingProcedure"/>
</owl:Class>
<owl:Class rdf:ID="ControlRoom">
  <rdfs:subClassOf rdf:resource="#OperatingProcedure"/>
  <owl:ObjectProperty rdf:ID="hasParent">
    <rdfs:subPropertyOf>
      <owl:ObjectProperty rdf:ID="hasChild">
    </rdfs:subPropertyOf>
  </owl:ObjectProperty>
</owl:Class>
    
```

Fig. 4. Boiler ontology RDF/OWL file.

Fig. 3을 살펴보면 운전절차 클래스와 보일러의 비정상적인 운전에 따른 정지절차, 행위장소 등의 관계를 제시한다. 이러한 정의에서 RDF(S) KR은 온톨로지의 기본요소인 rdf:Property와 결합된다. 등록정보인 rdf:ID 및 rdf:resource도 사용되며, 운전절차와의 비정상적인 운전상태 및 행위장소와의 관계는 RDF(S) KR 온톨로지의 기본요소인 rdf:Class, rdfs:comment, rdfs:subClassOf, rdfs:domain 및 rdfs:range 등으로 rdf:Property와 결합된다. 등록정보 rdf:ID 및 rdf:resource도 사용된다.

웹 온톨로지 언어로서 OWL은 클래스와 속성의 개념 및 그들 사이의 관계가 보다 명료하게 정의되도록 정리한 것으로, 클래스의 구성원들 간에 관계를 기술하고, 이러한 명제의 집합으로 이뤄진 구문적으로 정의되지 않는 것은 논리적 유추가 가능하게 한다. Fig. 4는 OWL로 표현된 온톨로지로서 boiler는 operating procedure의 서브 클래스이며, hasParent와 hasChild의 특성을 갖는 것을 정의한다.

2.3 행위계획 HTN planning (Hierarchical Task Network Planning)

HTN planning은 상태전이시스템에 의해 조건에 제한을 둔 예전의 플래닝과 결부된 인공지능(artificial intelligence) 플래닝 기법이다[13]. 이 플래닝의 기본 아이디어는 초기상태를 설명하고, 달성 할 목

표를 task 네트워크, 원시 및 복합 task 네트워크로 구성된 도메인 지식을 포함한다. 이때, task 네트워크는 task가 원시적이거나, 자세하게 하위 task로 분류된 경우에 실행할 수 있는 task 계층을 나타낸다. 또한 플래닝 프로세스는 초기 task 네트워크를 분해하는 순서로 시작되고, 모든 복합 task가 완전히 분해될 때까지 진행된다. 즉, task에 해당되는 솔루션을 찾게 되는데, 이 솔루션은 초기 환경상태에 적용할 수 있는 일련의 기본적인 task와 동일한 플래닝을 가지게 된다.

HTN planning은 $P=(d, I, Op, Me)$ 로 나타낼 수 있다. 이때, d는 초기 task를 의미하며, 플래닝을 세우는데 필요한 제약조건을 말한다. 그리고 I는 초기 상태를 말하며, Op는 이용 가능한 운영집합이며, Me는 사용가능한 방법에 대한 집합이다. 각 방법(Me)은 복잡한 task를 하위 task로 분해하는 방법을 설명하고, 각 Op는 기본 task를 구현하는 방법을 설명한다. 그리고 HTN 플래닝은 복잡한 task를 하위 task로 분해하는 방법을 사용하며, 일반적으로 도메인에서 task를 수행하는데, 표준방법을 제공한다.

Fig. 4는 특정 CEP 매크로 흐름에 대한 밸브절차와 CEP 운전계획을 구성한 것으로, HTN 플래너는 전제조건을 충족하기 위해 상위 task를 전제조건을 만족하도록 하위 task로 분해하여 행위계획을 나타낸 계략적인 개요이다. 최상위 스키마는 모든 하위

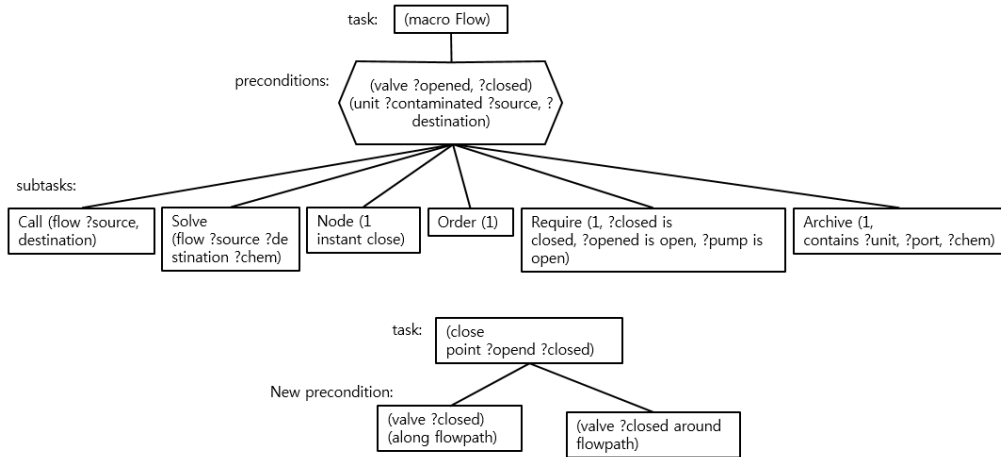


Fig. 5. Example of flow macro planning operator(adapted from Nau et al.[14]).

task에 대해 사전에 지정한 선행 전제조건(precondition)을 통해 정렬된 도메인을 나타내고, 하위 스키마는 도메인이 부분적으로 정렬되어 다른 하위 task 사이에 추가하는 방법을 표현한다.

계층적 task 구조에서 HTN 플래닝에 기반을 둔 독립적인 도메인 플래닝 방법으로 단계별 실행될 순서대로 각 플래닝의 단계를 생성하므로 실행 순서 단계별 상태를 알 수가 있다[14]. 이는 불확실성을 제거함으로써 복잡한 추론을 줄여 실질적인 표현을 통해 플래닝 시스템에 쉽게 통합할 수가 있다.

III. 온톨로지 기반의 셋다운 절차의 자동 생성

3.1 조업절차 자동생성을 위한 온톨로지 구축

온톨로지 구축방법에는 여러 가지가 있으며, 현재도 새로운 방식의 지식 모델링 기법이 연구되고 있다. 가장 일반적인 Stanford 대학에서 제안한 Ontology Development 101[15]을 참고하여 보일러의 운전절차의 자동 생성을 위해 운전정지절차를 목적으로 개발하였다.

첫째, 조업조건과 운전상태는 일반적인 플랜트 조업자의 직무 동사를 구분하여 일반행위와 조치행위 그리고 기술용어로 분류하여 인스턴스 값으로 표현하였고,

둘째, 플랜트를 구성하고 있는 장치들을 운영목적 별로 하위 task로 분할하고, 조업절차를 위해 구분한 각 클래스 별 전제조건 상관계로 나타냈으며,

셋째, 제한조건, 운영상황 및 조업절차에 따른 목표와 하위 목표들로 나누기 위해 계층적 관계를 도식화 하였다.

조업 운전정지 절차생성에 필요한 지식을 체계적으로 표현하기 위해 Peeters[16]가 제안한 클래스 계층 구조를 참조하여 각 클래스 간에 상위 온톨로지로 개별 OWL파일로 구분하였다. 이러한 구분의 의미는 조업절차에 관련된 지식을 구조화하고, 클래스 별로 관계를 나누어 다른 조업절차 영역에서도 쉽게 확장하고 재사용이 가능하도록 구분하기 위함이며, 조업절차 생성을 위한 영역추론에 사용된다.

Fig. 6은 온톨로지를 Protege 오픈소스 편집기를 사용하여 설계되었고, 계층적으로 순서가 지정된 클래스 간의 슬롯을 통해 parent와 child 관계를 구분하였다. 슬롯에서는 각 클래스 간에 추론을 위해 parents, child, constraint, decomposition, name 그리고 precondition 등을 정의하고, 각각의 슬롯에 대해 주석 및 특징, 인버스, 클래스와의 도메인 집합, 클래스 범위는 Fig. 7과 같이 클래스 간의 추론 관계를 표현한다. 또한 각 클래스의 의미는 Table 1과 같다.

조업자의 인지 및 지각 등을 표현한 일반적 행위 또는 조치행위에 관해 중복 또는 다중의 값으로 쓰이는 경우에 대해서는 슬롯의 값으로 정의하였다. 예를 들면, 행위가 일어나지 않는 상황에서는 0, 설비의 "정지 시킨다"에서는 1로 표현하는 precondition 슬롯의 다중 값으로 사용하였다. 해당되는 보일러의 펌프 또는 연료 공급 인스턴스 등은 클래스 사례가 갖는 값으로 "정지시킨다"는 보조설비인 펌프 등에 허용된 클래스에서 인스턴스 형으로 슬롯에 정의된다.

Fig. 8의 운전절차 구조는 SmartObject 구문에 using으로 정의된 Slot은 SmartObject와 클래스 그

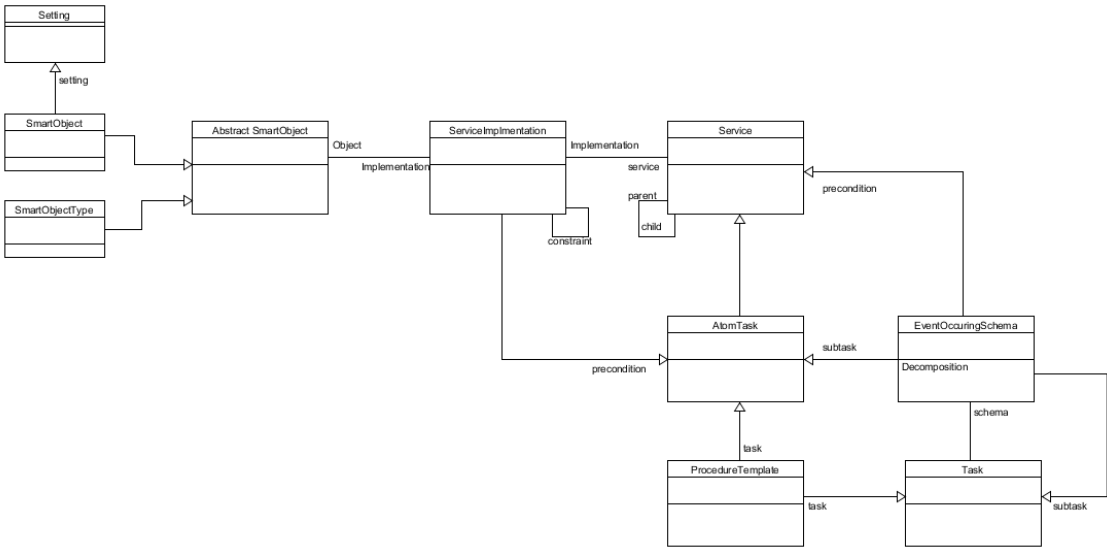


Fig. 6. UML(Unified Modeling Language) diagram of upper-ontology domain knowledge concepts(adapted from Peeters[16]).

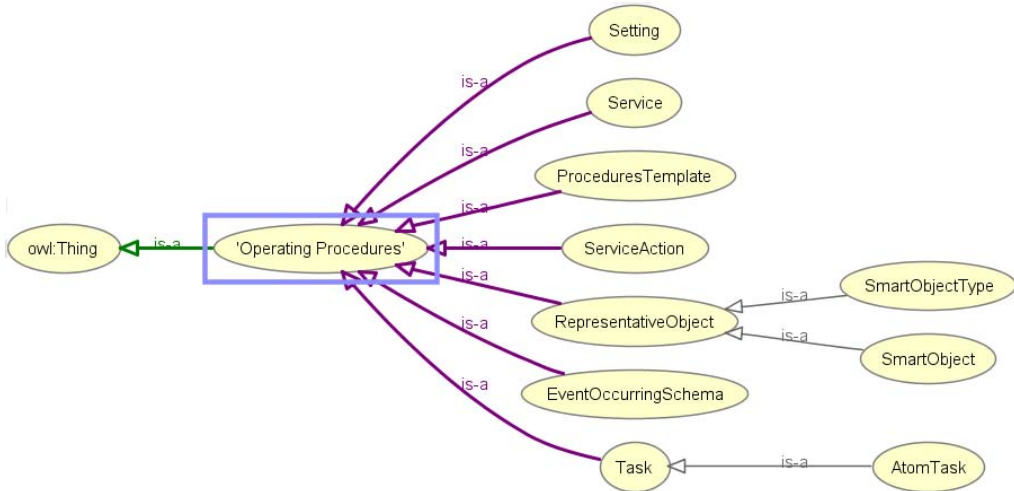


Fig. 7. Asserted hierarchy in Protege.

롭으로 제한되고, ServiceAction 클래스를 참조하게 된다. 생성 시나리오의 선택 메커니즘은 항상 대상 task에 적용된 템플릿 작업으로 분해되어 시작된다. 이 task는 작은 단위로 완전 분해 되었을 때만 다른 템플릿을 고려하게 된다.

조업절차를 생성하기 위해 의미론적 프레임 워크에 의해 추론된 정보를 확장하기 위한 규칙이 필

요하다. 상위 클래스인 OperatingProcedures는 ProcedureTemplate, RepresentativeObject, SmartObject의 하위 클래스 관계를 가지고 있고, 각각의 하위 클래스는 hasParents, hasChild로 데이터 형식은 문자열로 정의하여 각각의 클래스 별로 규칙이 적용되어 있다. 여기서 쓰인 SWRL은 OWL 기반의 규칙언어로 개별 클래스 별로 추론 규칙을 작성하고

온톨로지 기반의 보일러 섀다운 절차 생성 : 지식표현 및 훈련시나리오 활용

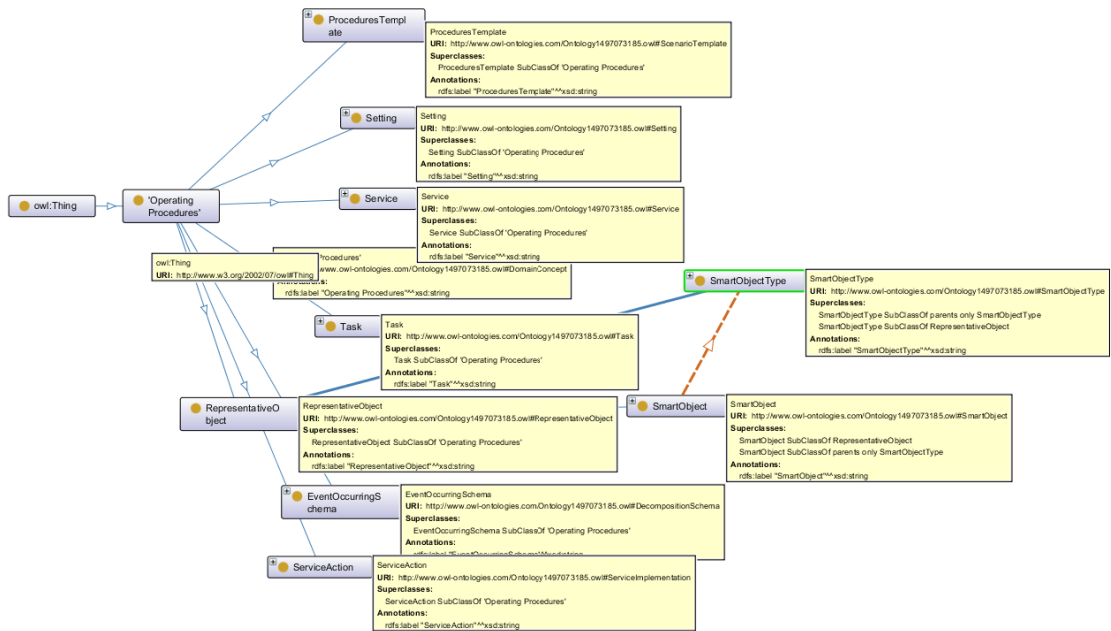


Fig. 8. Interactive relationships of procedures: OWL Ontology from OntoGraf.

Table 1. Definition of class about operating procedure

Class names	Individual class definition	Examples
owl:Things	Protege에서 처음 클래스의 계층 구조를 생성시 default인 OWL 파일 정의	-
OperatingProcedure	온톨로지 조업 절차 도메인	이상진단, 비상대응, 조업작업절차
RepresentativeObject	조업 관련 운영 및 기능 습득 목표	-
SmartObject	조업 절차 하위 목표	운전조건 습득, 장치 사용, 조업방법
SmartObjectType	하위 목표 수단, 방법, 수치	의사소통, 장치 컨트롤, 현장조치, 계기 모니터링
ServiceAction	조업 절차 생성시 단계별 목표 및 행위	정상운전방법, 비상운전방법
EventOccurringSchema	조업 절차 사건계획, HTN 방법 행위 분해 스키마	조치/방법/중단/계속행위
ProcedureTemplate	요구되는 상위 또는 하위 목표 달성을 위한 조업 절차 템플릿	-
Task	조업자의 개별 목표를 이루기 위한 행위 정의	수행/확인/반복/중단/계속
Setting	조업 절차 생성을 위한 조업자의 행위 및 용어 설정	-

새로운 지식을 유추하는데 사용될 수 있다. SWRL 규칙에서 '∧'는 연결을 나타내고, "?x"는 변수, "!"

는 함축을 표현한다. 변수에 "?"가 없으면, 상호 관계가 없는 개별 클래스를 말한다.

3.2 보일러 조업절차서 온톨로지 도메인 및 범위결정

온톨로지 기반의 보일러 조업절차를 생성하기 위해서는 먼저 대상범위를 선정할 필요가 있다. 보일러 비상운전정지 절차는 필요한 지식과 숙련 기술을 제공함으로써 과업에 대처할 수 있는 조업자의 과업 숙련도를 결정하는 요소로, 일반행위는 Table 2의 직무를 위한 일반행위에서 장치정보에 관해 “신규장치 사용을 위한 훈련” 일반 또는 조치행위에 대한 “익숙지 않는 상황의 처리”, 조치행위에 관한 “긴급절차서 사용에 관한 훈련”, “자동 제어 및 보호 시스템을 사용한 작업” 등이 물리적 숙련 기술을 분석하기 위한 기술용어를 활용함으로써 인식적 숙련, 의사결정 숙련, 그리고 장치의 세련된 조작 기술에 관한 훈련 또한 필요하다. 조업절차를 위한 시나리오는 여러 가지 이산사건의 조합을 통해 절차적 순서로 구성하고 조업범위 내에서 이루어지는 동작들을 구성하게 된다. 즉, 1) 조업의 초기상태의 선택(정상, 비상), 2) 조업목표 선택, 3) 조업

또는 조치 행위절차 선택, 4) 시나리오 상에서 발생하는 사건의 계획, 5) 일반 또는 조치행위 선택, 그리고 조업자의 행위를 수행하기 위해 실제 발전소의 유형에 따라 각 발전소의 정상/비정상, 출력 증발/감발 등의 운전상태에 따라 연속, 연결 그리고 분기방식으로 변화한다.

3.3 HTN 플래닝을 활용한 보일러 운전정지 절차

온톨로지를 단순화한 도식으로 스템의 상태에 따른 HRH, SH, MP, LP 등이 steam state stage의 하위 클래스라는 관계(isa HRH HRH)로 표시된다. 그리고 HRH 클래스와 인스턴스 HRH 1의 관계는 (클래스 HRH1 HRH)로 정의할 수 있다. 간단한 불완전한 온톨로지 모델에서 쉽게 절차서와 같은 예제 등을 추출이 가능하다고 판단하며, 몇가지 가정과 전제조건, 그리고 지식표현 등을 실행하면 자동생성이 가능한 온톨로지 모델이 될 것이라 판단된다. 그리고 HTN 탐색은 다음과 같이 구성된다. 즉, 1) 초기 상태의 집합을 표현, 2) 기존 온톨로지 저장

Table 2. Classification of operator task verb for boiler operating shutdown procedure

구분	직무동사 정의	유형의 예
일반행위	<ul style="list-style-type: none"> • 장치 운전해 담당하는 조업자 작업행위 • 운전정보 입력 및 획득행위 • 운전정보 처리 및 의사결정행위 	<ul style="list-style-type: none"> • 수행/확인/점검/찾아보기 • 조치/반복/중단/계속
조치행위	<ul style="list-style-type: none"> • 운전비상시 대응을 위한 조업자 수행행위 	<ul style="list-style-type: none"> • 조치/제거/차단/요청/참조/알림
기술용어	<ul style="list-style-type: none"> • 대상 플랜트의 장치 및 기술분류용어 • 장치의 동작상태 	<ul style="list-style-type: none"> • 펌프/열교환기/보일러/밸브/필터 • 정상/과열/과압/법람/진동/이탈

Task: Shutdown heaters OP1 from Ma1 to Au1 Precondition: OP1 class: Operation room Ma1 class: main equipment Au1 class: aux. equipment Ma1 in B1 Au1 in B1 B1 class: Boiler OPC1 class: Control OPC1 at OP1 SubTask: load OP1 OPC1 operation OPC1 Ma1 Au1 unload OP1 OPC1	Task: Shutdown ?p1 from ?a1 to ?u1 Preconditions: ?p1 class: operation room ?a1 class: normal equipment ?u1 class: normal equipment differ ?a1 ?u1 ?a1 in ?c1 ?u1 in ?c2 ?c1 class: boiler ?c2 class: boiler Same? ?c1 ?c2 ?pc1 class: control ?pc1 at ?a1 Subtask: load ?p1 ?pc1 drive ?pc1 ?p1 ?u1 unload ?p1 ?pc1
--	---

Fig. 9. Generating simple heaters shutdown sequence from plan cases.

된 도메인과 관련 배경지식 기반, 3) 설계자가 정의한 새로운 도메인 등이며, 이러한 관계를 통해 원시 목적과 task보다 더 포괄적이고 정확한 계획을 얻을 수 있다.

Fig. 9는 절차를 생성하기 위한 HTN 플래닝 입력의 한 예로, Shutdown heaters 클래스의 변수 개체를 추상화하여 간단한 모델을 생성하였다.

3.4 보일러의 구조

보일러 플랜트는 발전용 보일러와 산업용 보일러로 나뉜다. Fig. 10과 같이 발전용 보일러는 전기를 생산하는 것이 주 목적이며, 산업용 보일러는 공정용 증기, 난방, 전기 사용 등을 목적으로 단독 또는 복합적으로 사용할 수 있게 구성되어 있다. 이러한 보일러는 사용 연료, 순환 방법, 증기압 및 연소 방식에 따라 구분되어 진다. 그러나 보일러의 정상 운전 정지절차는 비슷하며, 이러한 보일러는 공기 공급부, 보일러 급수, 연료 가스 또는 오일 공급부, 버너 장치 및 드럼 장치로 구성되어 있다. 보일러 급수는 이코노마이저를 통해 예열되며, 상부드럼으로 흘러 버너의 열에 의해 증기로 변환된다. 증기는 과열기를 통해 고압 및 고온으로 과열되어 스팀이 되며, 스팀은 스팀터빈으로 공급된다. 버너에서 연료를 효율적으로 연소시키기 위해 공기 예열기를 거쳐 강제 팬을 통해 노 내로 이송된다.

3.5 보일러 운전정지 절차

보일러 설비의 운전정지는 여러 가지 제약조건

을 가진 정교한 운전절차가 필요하며, 세 가지 범주로 구분 할 수 있다.

- Normal Shutdown : 계획 정비에 따른 운전정지, 설비의 개선 및 계획운전 정지
- Emergency Shutdown : 해당 설비의 안전 제한을 초과한 변수 또는 설비의 고장, 예기치 못한 시스템의 오류에 의한 정지
- Partial Shutdown : 정상 또는 비상 종료로 인한 설비의 일부를 정상운전 상태로 변경

보일러의 운전정지 절차로 여러 상황에 따른 조업절차 중 일부를 HTN 방법을 통해 task 중심으로 분할하여 분해 스키마로 구성하였다. 일반적으로 보일러 설비의 운전정지는 작업자의 운전 매뉴얼에 제시되어 있는데, 비상상태의 경우 상황 파악 및 운영결과 평가를 토대로 절차를 수행하게 된다. 정형화되고 일반화 된 지식에 의해 보일러 플랜트의 가동중단을 위한 표준절차를 표현할 필요가 있으며, 다양한 상황에서 보일러 플랜트에 적용될 수 있다.

3.6 온톨로지 보일러 운전정지 절차

일반적인 보일러의 정상운전 절차의 생성을 위한 행위지식을 표현하기 위해 일반행위 동사 등을 구분하면, 지각, 인지, 수행, 기능 등으로 분류할 수 있다. 보일러의 용어를 정리하고 개별 조업자의 휴리스틱을 적용하여 수행 순서를 임의 조합과 구성, 행위의 시작과 확인 그리고 종료로 정의하였다.

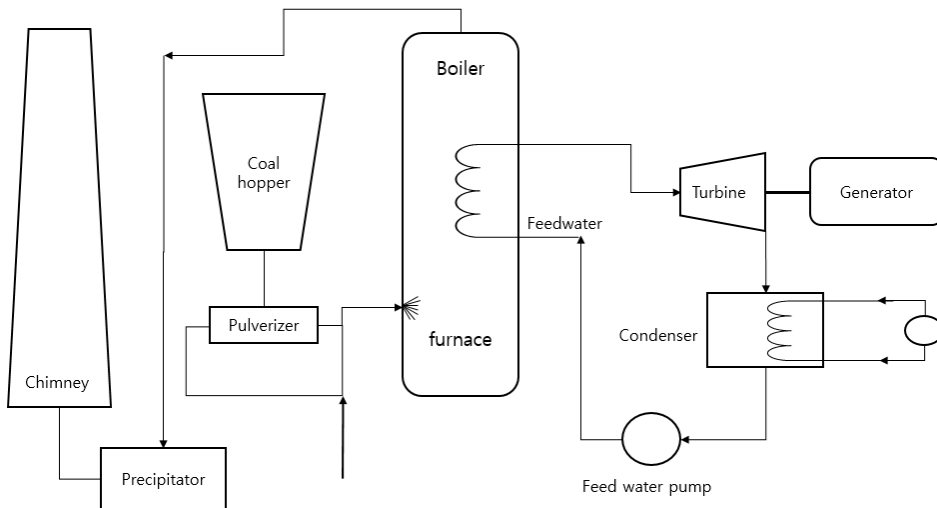


Fig. 10. Fossil power plant diagram.

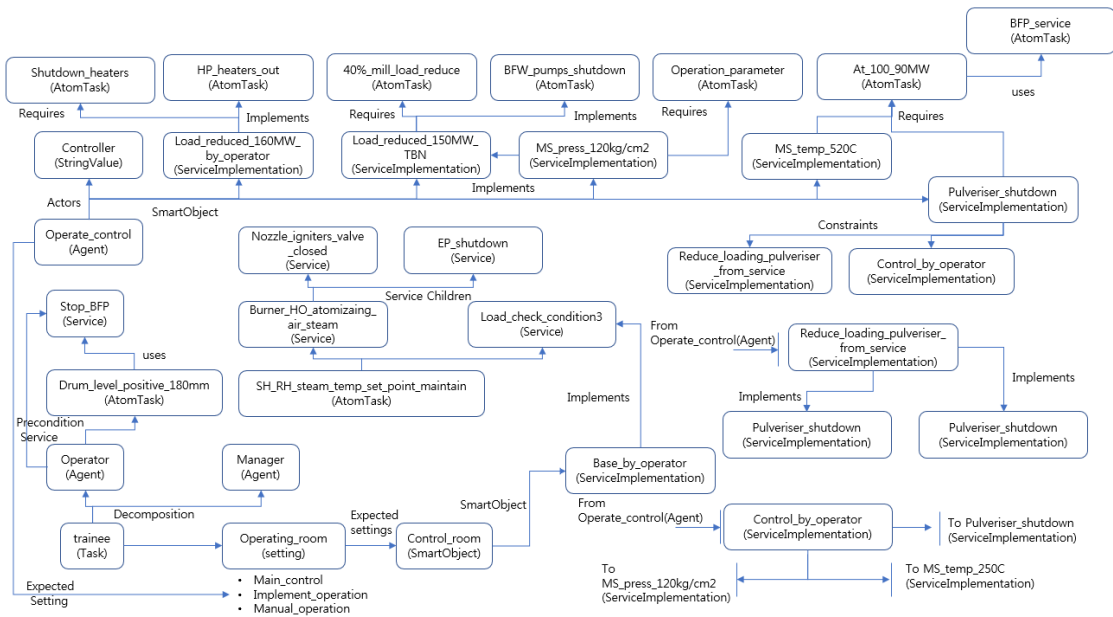


Fig. 11. HTN Planning of boiler shutdown sequence including instances.

```

EventOccurringSchma(String id, String name, String taskId,
ArrayList<String> decomposition, ArrayList<String>
preServiceId, ArrayList<String> preObjectIds) {super(id, name)
_taskId= taskId
_decomposition=decomposition
_precond_services=preServiceIds
_precond_objects=preObjectIds}

//return, task 분할에 의한 하위 task
Array ArrayList<String> getDecomposition()
ArrayList<String> copy = new ArrayList<String>();
for(String t : _decomposition)
copy.add(t);
return copy;

//return 분할스키마에 적용할 task
String getTaskId() return_taskId

//return, 분할 스키마에 적용되기 전 요구되는 서비스
Array<String> getAllServicePreconds()
Array<String> copy=new ArrayList<String>()
for(String t:_precond_services)
copy.add(t)
return copy

//return 분할 스키마에 적용되기 전 요구되는 object
Array<String> getAllObjectPrecond()
Array<String> copy=new ArrayList<String>()
for(String t:_precond_objects)
copy.add(t)
return copy
    
```

Fig. 12. Decomposition schema Java code.

따라서 Fig. 11에서 일반적인 조업자를 상대로 하는 것이 아닌, 대상 플랜트에 종사하는 관계자를 대상으로 절차서의 기술은 전문영역지식을 기반으로 용어들이 구체화 할 필요가 있다. 이때, 개념도 설계를 위해 slot은 Implementation, requires, uses, precondition등을 정의하여 클래스에 해당하는 인스턴스 간의 관계를 정의하였다.

조업절차를 생성하기 위해 대상 플랜트의 보일러 운전정지절차의 생성된 결과를 확인해 보면 task 또는 AtomTask(행위) using SmartObject(대상, 또는 도구) for ServiceAction(목표)의 온톨로지의 RDF/XML을 통해 출력된 결과를 확인해 볼 수 있다. 그리고 슬롯의 정의에 따라 클래스간의 인스턴스를 입력 후 프로그램의 실행을 통해 다양한 절차가 생성이 되며, 다른 관련 지식을 반영하여 재사용하여 온톨로지를 구축하면, 여러 상황의 온톨로지 재사용에 대한 결과를 얻게 된다.

3.7 HTN Planning 알고리즘

조업절차를 생성하기 위한 HTN 플래닝을 적용한 개별 도메인들은 추상화된 상위 온톨로지 클래스에서 하위로 세부적이고 구체화하는 과정이 필요하다. 상위 레벨에서 하위 레벨로 구체적인 인스턴스 값들을 입력하는 과정으로 다른 레벨에 있는 온톨로지에 있는 지식과의 일관성을 유지해야 한다.

Fig. 12는 HTN 방법의 task 분할 알고리즘으로 task분할과 전제조건을 표현하였다. 이때, 알고리즘으로 주요 실행방법은 다음과 같다.

1. 코드에서 인식한 온톨로지 파일 ObjectSelector, timeLine, Task decomposer parsing
2. ProcedureTemplate의 문자열로 선언된 인스턴스 값 목록 순서 정렬
3. Task의 문자열로 선언된 인스턴스 값 목록
4. Setting의 문자열로 선언된 인스턴스 값 목록
5. ProcedureTemplate에 있는 critical task 인덱스 조회, 새로운 분할 task 목록

IV. 결과 및 토의

4.1 화력발전 보일러의 적용사례 비교

Table 3은 일반적으로 사용되는 매뉴얼의 보일러

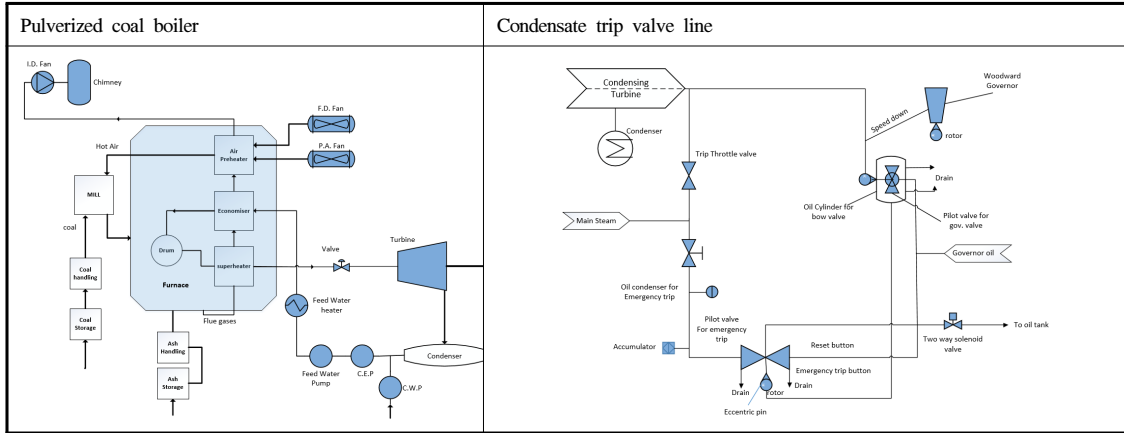
운전정지 절차와 온톨로지 기반의 생성된 정지절차를 비교하였다. 기존의 OPS를 살펴보면 보일러 셧다운을 위한 지식표현은 운전상태 표현과 조작을 행하기 전의 예비조건 등을 표현하였다. 이러한 지식표현은 중간목표들을 생성 실행하도록 되어 있으나, 많은 시간이 소요되고, 시간적 제약에 따른 조건들을 세부적으로 적용해야 하는 문제와 조업자에게 운전절차의 지식 습득에 대한 제약사항이 따른다.

벨브 시퀀스의 합성방법은 boolean logic의 and, or와 같은 기호를 목표상태까지 연속적인 조작들을 적용하기 때문에 시간적으로 비효율적이고, 초기상태에서 최종 목표상태에 대한 하나의 운전절차를 찾는 문제점이 있다. 이러한 단점을 보완하기 위한 시도로 목표상태에 도달하는데 불필요한 조작을 제외하는 탐색조건 토대로 경험적인 규칙을 사용하는 운전절차 방법은 두 개 이상의 운전조건이 필요 하는 경우에는 적용하기 어렵다.

Table 3. Comparison between classical and automated boiler shutdown procedure

Boiler emergency shutdown procedure	
Boiler abnormal by electric power failure	Boiler emergency shutdown procedure using ontology
배기가스 온도 250℃ 초과, 배기가스 세정 시스템 급수 불량 및 필터 입구 250℃초과 하는 경우 · Step 1. 긴급 상황 발생시, 연료 공급 중단, 배기가스 온도 250℃ 초과시 비상공급장치 스프레이 타워 사용, 스크러버 과열 방지 위해 냉각 밸브 열림 · 배기가스 세정 시스템은 비상 발전을 통한 ID 팬 재가동 때 까지 작동 · 배기가스의 흐름 정상흐름의 20%미만, 남아 있는 연료의 완전 연소까지 ID 팬 약 2시간 가동 · 보일러 급수 중단, · Step 2. 냉각 단계, ID 팬 작동 정지 후 보일러 배출구를 통해 남아 있는 연소 가스 배출(잔열 약 3MW)	<ol style="list-style-type: none"> 1. Furnace_operation_condition using spray_tower(2) for flue_gas_above_250C, 2. Supplementary_valve_closed(0) for Burners_HO_atomizing_air_steam, Drum_level_positive_180mm using Operate_Control(1) for stop_BFP, 3. Furnace_ID_fan_running using Operate_Control(1) for flue_gas_cleaning_system, 4. Continuous_reduce_firing_rate(3) for temp_press_drop_within_load, APH_lubeOil_pump_stop_interlock using Operate_Control(1) for oil_temp_40C, 5. Furnace_ID_fan_stopped using Aux_power_supply for furnace_cooling_condition, 6. Drum_level_maintain_auto using Operate_Control(1) for boiler_water_temp_94C, Check_boiler_expansion_contraction(4) for remove_heating_furnace
Condenser abnormal trip	Condenser emergency shutdown procedure using ontology
· Shaft의 eccentric pin 돌출, trigger lever 작동 · Pilot valve의 spindle 연결 상태 이탈 · Oil emergency trip cylinder 내부 가압으로 인해 Oil 드레인, pilot valve 하부에 공급 · Oil emergency trip cylinder의 압력 저하, piston이 spring 압력에 의해 하강, valve 닫힘 · Governing valve의 pilot valve 올라가고, oil cylinder의 하부 공간에 oil drain · Governing valve 닫힘, trigger lever로 인한 밸브 닫힘, · Electric trip signal 동작, turbine shutdown	<ol style="list-style-type: none"> 1. Condensate_outlet_flow_check using Inspection(6) for condensate_vaccum_press_abnormal, 2. Trip_throttle_valve using trigger_lever(5) for pilot_valve_close, 3. Check_boiler_expansion_contraction using operate_Control(5) for boiler_ShutDown_sequence, 4. Governing_valve_control(7) for condensate_extraction_flow_condition, 5. Slip_check_PurgeValveOpen using Operate_Control(5) for oil_temp_40C, 6. Trip_throttle_valve using Operate_Control(5) for governing_valve_close, electric_trip_signal(8) for turbine_shutdown

Table 3. continue



{((close a)(close x)), ((open b)(open y)), ((close b)(close y)), ((open c)(open z)), ((close c)(close z)), ((open b)(open y)), ((close b)(close y)), ((open a)(open x))}

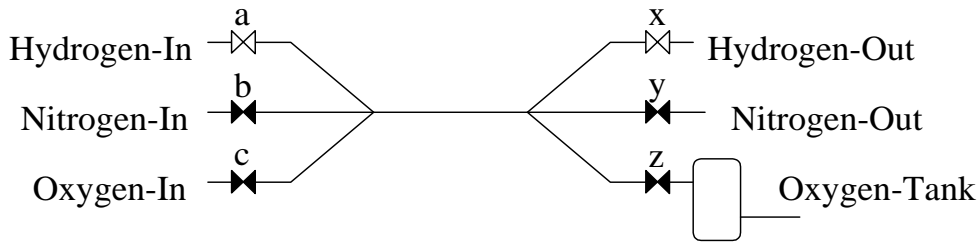


Fig. 13. Bottleneck problem[17].

온톨로지 기반의 보일러 운전절차는 Fig. 14과 같이 RDF(S)파일로 생성되어 공통의 지식 통합문제와 용어의 불일치를 해결하는 접근을 통해 설비의 운전절차의 지식 표현의 표준화를 시도한 점이 큰 의미가 있으며, 향후 확장을 통해 Web의 온라인 공간에서와 기계언어로의 해석 그리고 웹에 흩어져 있는 관련 지식의 정보를 취합 및 관련 지식을 공유하기 수월하며, 다른 설비에 관한 지식정보와 쉽게 결합할 수 있는 장점을 가지고 있다. 또한 RDF schema를 통해 domain, range, resource 등의 의미 표현의 제한이 없으며, 다른 설비의 운전절차 또한 RDF파일로 쉽게 병합이 가능하다.

4.2 조업훈련 시나리오의 활용

HTN 기반의 플래너를 통해 task의 실행 가능한 세부수준인 subtask까지 분할하여 class와 subclass와 매핑, 그리고 partially order된 형태로 실행 가

능한 primitive subtask로 연산하도록 할 수 있다. 조업절차 생성을 위해 9개의 class(subclass 2개 포함)로 계층구조를 설계하고, 각각의 instance로 매핑한 HTN 플래닝 방법은 행위순서에 대한 심각도(severity) 레벨을 구분하고, SWRL rule을 생성하여 적용하면, 지식 추론을 통해 수준별 훈련생의 이동 및 임무 수행에 대한 플래닝 기반의 시나리오의 자동생성이 가능하다.

간단한 예를 들면, 조업자 안전훈련 시나리오를 생성하기 위한 Table 4의 SWRL 룰을 작성하면 그 의미는 다음과 같다. 즉, 컨트롤 룸에서 훈련생의 작업선택의 분할에 따른 심각도 10 이상이 된다면, 레벨 1의 임계값을 초과하므로, incident type은 "action_Trainee_level 1"로 추론하게 되고, "work_Field_valve_station"의 위치에서 "manual_working_at_valve_change"의 행위를 획득하게 된다.

위의 예에서 살펴보았듯이 온톨로지 기반의 HTN

```

////////////////////////////////////
//
// Classes
//
////////////////////////////////////

-->
▼<!--
  http://www.owl-ontologies.com/Ontology1497073185.owl#AtomTask
-->
▼<owl:Class rdf:about="http://www.owl-ontologies.com/Ontology1497073185.owl#AtomTask">
  <rdf:subClassOf rdf:resource="http://www.owl-ontologies.com/Ontology1497073185.owl#Task" />
  <rdf:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">AtomTask</rdf:label>
</owl:Class>
▼<!--
  http://www.owl-ontologies.com/Ontology1497073185.owl#DecompositionSchema
-->
▼<owl:Class rdf:about="http://www.owl-ontologies.com/Ontology1497073185.owl#DecompositionSchema">
  <rdf:subClassOf rdf:resource="http://www.owl-ontologies.com/Ontology1497073185.owl#DomainConcept" />
  <rdf:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">EventOccurringSchema</rdf:label>
</owl:Class>
▼<!--
  http://www.owl-ontologies.com/Ontology1497073185.owl#DomainConcept
-->
▼<owl:Class rdf:about="http://www.owl-ontologies.com/Ontology1497073185.owl#DomainConcept">
  <rdf:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">OperatingProcedure</rdf:label>
</owl:Class>
▼<!--
  http://www.owl-ontologies.com/Ontology1497073185.owl#RepresentativeObject
-->
▼<owl:Class rdf:about="http://www.owl-ontologies.com/Ontology1497073185.owl#RepresentativeObject">
  <rdf:subClassOf rdf:resource="http://www.owl-ontologies.com/Ontology1497073185.owl#DomainConcept" />
  <rdf:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">RepresentativeObject</rdf:label>
</owl:Class>
▼<!--
  http://www.owl-ontologies.com/Ontology1497073185.owl#ScenarioTemplate
-->
▼<owl:Class rdf:about="http://www.owl-ontologies.com/Ontology1497073185.owl#ScenarioTemplate">
  <rdf:subClassOf rdf:resource="http://www.owl-ontologies.com/Ontology1497073185.owl#DomainConcept" />
  <rdf:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">ScenarioTemplate</rdf:label>
</owl:Class>
▼<!--
  http://www.owl-ontologies.com/Ontology1497073185.owl#Service
-->
▼<owl:Class rdf:about="http://www.owl-ontologies.com/Ontology1497073185.owl#Service">
  <rdf:subClassOf rdf:resource="http://www.owl-ontologies.com/Ontology1497073185.owl#DomainConcept" />
  <rdf:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Service</rdf:label>
</owl:Class>
▼<!--
  http://www.owl-ontologies.com/Ontology1497073185.owl#ServiceImplementation
-->
▼<owl:Class rdf:about="http://www.owl-ontologies.com/Ontology1497073185.owl#ServiceImplementation">
  <rdf:subClassOf rdf:resource="http://www.owl-ontologies.com/Ontology1497073185.owl#DomainConcept" />
  <rdf:label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">ServiceAction</rdf:label>
</owl:Class>

```

Fig. 14. RDF/OWL expression of boiler shutdown ontology.

Table 4. Example SWRL rule to application operator safety training

SafetyTrainingCase(?) ^ hasWorkPlaceName(?x, Control_Room) ^ hasTraineeLevelNumber(?x, ?y) ^ swrlb:greatThan(?y, 10(temporal selectivity No.))	hasIncidentType(?x, action_Trainee_level 1)
SafetyTrainingCase(?) ^ hasIncidentType(?x, action_Trainee_level1) ^ hasPosition(?x, Work_Field_valve_station)	hasProcedure(?x, manual_working_at_valve_change)

플래닝은 지식을 구조화되고, 의미론적인 방식으로 저장 관리하여 효율적인 행위기반의 시나리오 생성에 활용할 수 있다. 이러한 결과를 통해 Aachen공대의 공정 온톨로지 OntoCAPE도 융합한 공정 안전분야의 온톨로지 향후 확장 구축에 공유할 기회가 크다 여겨진다.

V. 결론

본 연구에서는 보일러 운전정지 절차를 중심으로 온톨로지 기반의 체계적인 플랜트 운전지식 표현절차를 제안하였다. 이를 위해 온톨로지 오픈소스 에디터인 Protege를 사용하여 9개의 클래스와 172개의 인스턴스로 구성해 조업절차 중 섰다운 지식표현 계층도를 구성하였으며, 보일러 설비에서 공급 전기누전 상황과 복수기 라인에서의 갑작스런 비상상황 발생 2가지 경우를 대상으로 섰다운 절차서 결과를 얻었다. 이러한 결과는 온톨로지 보일러 운전정지 절차의 구조성과 구체성을 통해 구현되었으며, 이를 실제 대규모의 복잡하고 분산된 조업환경에 적용 시 에이전트들의 지식기반의 개념 계층도를 통한 지식과 정보 교환이 수월하게 될 것으로 기대된다. 또한 기존의 운전정지 절차서와 비교한 결과, 조업행위에 대한 구체적인 지식표현 계층도의 수립이 수월하며, 결과에 대해 쉽게 이해할 수 있음을 알 수 있었다.

또한, 조업운전절차서의 실시간 자동생성을 위해서는 공정 데이터의 모니터링과의 연계 노력이 필요하다. 특히, 제안방법은 공정상의 지식통합 문제 및 용어의 불일치를 해결가능하게 하며, 클래스별 Internationalized Resource Identifiers (IRIs)가 정의되어 시맨틱 웹을 통한 확장된 개념의 다른 조업절차 생성지식에도 활용될 수 있다. 또한 HTN 플래닝 알고리즘에 의해 생성된 절차서는 제각각 경험에 근거한 지식과 서로 다른 제조사의 확실적인 절차서에 비해 조업지식의 공유를 가능하게 하고, 지식표현의 표준화의 근거가 될 수 있을 것으로 판단된다. 본 연구는 보일러의 운전정지절차 생성방법을 중심으로 다루었지만, 제안방법은 다른 조업절차의 지식 표현과 추론을 통한 지식의 확장도 가능하게 하며, 다중 조업자 안전훈련 자동시나리오 생성에도 도움이 될 것으로 판단된다.

감사의 글

본 연구는 국토교통부 플랜트연구개발사업의 연구비지원(14IFIP-B085984-04)에 의해 수행되었습니다.

REFERENCES

- [1] R. S. Aylett, J. Soutter, G. Petley, P. W. H. Chung, and A. Rushton "AI Planning in a Chemical Plant Domain", *ECAI*, 622-626, (1998)
- [2] Bo Kyeng Hou, Kyu Suk Hwang, Soo Hyoung Choi, Dongil Shin, and En Sup Yoon, "An Operating Procedure Synthesis Method for Shutdown of Boiler Plants", *IFAC Proceedings*, **34**(27), 351-356, (2001)
- [3] Sung Gun Lee, Sang il Han, Yoon Sup Byun, Kyu Suk Hwang, Young Han Kim, Dongil Shin, and En Sup Yoon, "An Automatic Synthesis System for Start-Up Operating Procedures of Boiler Plants", *Korean J. Chem. Eng.*, **22**(4), 503-511, (2005)
- [4] Horridge, M., *A Practical Guide To Building OWL Ontologies Using Protege 4 and CO-ODE Tools*, Edition 1.3, *The University of Manchester*, (2011)
- [5] Lenat, Douglas B., Guha, R. V., Pittman, D., and Shepherd, M., "Cyc: toward programs with common sense", *Communications of the ACM*, **33**(8), 30-49, (1990)
- [6] Fensel, D., Horrocks, I., Van Harmelen, F., Decker, S., Erdmann, M., and Klein, M. "OIL in a nutshell", *International Conference on Knowledge Engineering and Knowledge Management*, 1-16 Springer, (2000)
- [7] Dean, M., Schreiber, G., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D. L., and Stein, L. A., *OWL web ontology language reference*, W3C Working Draft, (2003)
- [8] Baader, F., et al. (Ed.), *The Description Logic Handbook: Theory, implementation and applications*, 2nd Ed., Cambridge University Press, (2010)
- [9] Asuncion Gomez-Perez, Mariano Fernandez-Lopez and Oscar Corcho, *Ontological Engineering*, Springer, (2004)
- [10] Martijn van Welie, Gerritt C. van der Veer, and Anton Eliens, "An Ontology for Task World Models", *Design, Specification and Verification of Interactive Systems*, Vienna, 57-70, (1998)
- [11] Ora Lassila and Ralph R. Swick, *Resource description framework (RDF) model and syntax specification*, W3C RDF Model and Syntax Wor-

- king Group, (1999)
- [12] Asuncion Gomez-Perez, Mariano Fernandez-Lopez, and Oscar Corcho, *Ontological Engineering with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*, Springer (2003)
- [13] M. Ghallab, D. S. Nau, and P. Traverso, *Automated Planning: Theory & Practice*, Morgan Kaufman Publisher, (2004)
- [14] Dana Nau, Tsz-Chiu Au, Okhtay Ilghami, Ugur Kuter, J. William Murdock, Dan Wu, and Fusun Yaman "SHOP2: An HTN Planning System", *Journal of Artificial Intelligence Research*, **20**, 379-404, (2003)
- [15] Natalya F. Noy and Deborah L. McGuinness, *Ontology Development 101: A Guide to Creating Your First Ontology*, Stanford Knowledge Systems Laboratory Technical Report KSL-01-05, Stanford University, (2001)
- [16] Marieke Peeters, *Personalized Educational Games: Developing agent-supported scenario-based training*, Ph.D. Thesis, Dutch Research School for Information and Knowledge Systems, (2014)
- [17] Bo-Kyeng Hou and Kyu-suk Hwang, "Development of Algorithm for Operating Procedure Synthesis of large-scale chemical plants", *Theories and Application of Chemical Engineering*, **5**(2), 2641, (1999)