

협업적 제품 설계를 위한 온톨로지 기반 시맨틱 조립체 모델링

양 형 정^{*} · 김 경 윤^{**} · 김 수 형^{***}

요 약

협업적 제품 설계 환경에서는 여러 설계자가 작업에 참여하기 때문에 설계자간의 원활한 의사 소통이 필수적인 요소이다. 즉, 제품 설계에 내재되어 있는 다른 설계자의 설계의도를 파악할 수 있어야 하고, 각 설계자마다 사용하고 있는 상이한 모델링 용어에 대한 의미적 처리가 필요하다. 시맨틱 웹에서 온톨로지는 의미 정보를 명시적으로 표현할 수 있기 때문에 데이터와 서비스에 대한 통합되고 일관된 접근을 가능하게 한다. 따라서 제품을 여러 설계자가 설계하는 협업적 환경에서 온톨로지를 사용한다면 조립체내의 내포된 공학적 관계들, 공간적 관계들, 그리고 집합 관계들이 명확하게 표현되고, 추론을 통해 설계자의 의도가 파악되어 설계자간의 의사소통이 원활하게 이루어질 수 있다. 본 논문에서는 조립체를 온톨로지를 이용하여 표현함으로써 협업적 환경에서 조립체 설계정보를 공유할 수 있는 시맨틱 조립체 모델링 프레임워크를 제안한다. 시맨틱 조립체 모델링 프레임워크에서는 조립체 관계 모델(Assembly Relationship Model, ARM)을 OWL(Web Ontology Language)와 SWRL(Semantic Web Rule Language)로 표현한다. 이를 통해 조립체 내에 내포된 공학적 관계들, 공간적 관계들, 그리고 집합 관계들이 명확하게 표현되고, 추론되어 설계 의도가 파악된다.

키워드 : 온톨로지, 조립체, 협동적 제품설계, OWL, SWRL

Ontology-based Semantic Assembly Modeling for Collaborative Product Design

Hyung-Jeong Yang^{*} · Kyung-Yun Kim^{**} · Soo-Hyung Kim^{***}

ABSTRACT

In the collaborative product design environment, the communication between designers is important to capture design intents and to share a common view among the different but semantically similar terms. The Semantic Web supports integrated and uniform access to information sources and services as well as intelligent applications by the explicit representation of the semantics buried in ontology. Ontologies provide a source of shared and precisely defined terms that can be used to describe web resources and improve their accessibility to automated processes. Therefore, employing ontologies on assembly modeling makes assembly knowledge accurate and machine interpretable. In this paper, we propose a framework of semantic assembly modeling using ontologies to share design information. An assembly modeling ontology plays as a formal, explicit specification of a shared conceptualization of assembly design modeling. In this paper, implicit assembly constraints are explicitly represented using OWL (Web Ontology Language) and SWRL (Semantic Web Rule Language). The assembly ontology also captures design rationale including joint intent and spatial relationships.

Key Words : Ontology, Assembly, Collaborative Product Design, OWL, SWRL

1. 서 론

각각 다른 환경에서 개발되는 조립체의 정보를 일관된 프로임워크에서 접근할 수 있는가? 각 개발자가 필요한 정보

만 접근할 수 있는가? 다른 개발자의 설계 의도를 파악할 수 있는가? 본 논문에서는 위와 같은 문제를 해결하는 협업적 제품 설계를 위한 온톨로지 기반 조립체 모델링 프레임워크를 제안한다.

최근 제품 개발 환경은 컴퓨터 네트워크의 구축으로 지역적 개발에서 벗어나 전역적인 환경에서 제품의 협업적 설계와 개발이 가능해졌다[5, 22]. 제품 개발이 점차적으로 지식 집약적이고, 협업적인 추세를 보임에 따라 제품에 대한 지식의 표현, 접근, 재사용에 관한 연구가 더 중요해 지고 있

* 본 연구는 2005년도 누리 산학 협력 사업 지원(전자정보기전인력양성사업)에 의해 수행되었음.

^{*} 중신회원: 전남대학교 전자컴퓨터공학부 전임강사

^{**} 정 회 원: Wayne State University 조교수

^{***} 정 회 원: 전남대학교 전자컴퓨터정보통신공학부 부교수
논문접수: 2006년 2월 24일, 심사완료: 2006년 3월 6일

다[20, 32]. 제품의 협업적 설계에서는 여러 설계자들이 서로 다른 설계 틀을 이용하거나 각 설계자마다 서로 다른 용어를 사용하는 경우가 빈번하다. 따라서 제품 설계자 간의 설계 정보 공유를 위해 다른 설계자의 설계의도를 파악할 수 있어야 하며, 서로 다른 용어에 대한 의미적 처리가 필요하다. 또한 표준화된 데이터 형식이 요구되며, 설계 과정 동안에 필요한 의사 결정을 위해 지식의 제공이 필요하다.

한편 시맨틱 웹은 온톨로지를 통해 의미 정보를 명시적으로 표현함으로써 데이터와 서비스에 대해 통합되고 일관된 접근을 가능하게 한다. 온톨로지는 웹의 자원을 정확하고 공유할 수 있는 형태의 용어로 표현하고, 그 결과 자원에 대한 접근성과 기계 판독성을 높인다. 따라서 제품 설계에 온톨로지를 적용함으로써 제품 설계자들은 단순히 설계 정보를 전달하는 차원에서 벗어나 다른 설계자의 설계의도를 파악할 수 있고, 의미적으로 같은 제품의 설계 정보를 공유할 수 있을 뿐 아니라 설계와 제품 개발 단계에서 발생할 수 있는 지식들, 예를 들어, 명세, 설계 규칙, 제약 사항 등도 파악할 수 있다.

기계적 제품은 제품 공정상 하나의 부품으로 이루어진 경우가 드물고, 여러 개의 부품이 연결되어있는 조립체의 형태가 대부분이기 때문에 효율적인 조립체의 설계는 제품 가격을 결정하는데 중요한 요인이 된다[2]. 조립체의 공정은 많이 부분에서 자동화 되었지만 여전히 제품의 품질과 생산력의 부분에서 큰 영향을 미치고 있다[3, 7]. 따라서 제품 개발의 전과정에서 조립체를 관리하고 조작할 수 있도록 조립체 설계를 인지하고, 필요한 정보만을 전달하는 프레임워크가 필요하다.

강인한 조립체 모델을 생성하기 위해 조립체의 기하학적 형태 배열과 이에 따른 물리적 영향의 이해가 선행되어야 하지만 기존의 솔리드 모델이나 시뮬레이션 소프트웨어들은 제품의 기하학적인 정보를 가장 중요한 요소로 인식하고 있고, 설계 의도나 물리적 공정 원리 등을 표현하는 형태학적 특성은 고려하고 있지 않다. 따라서 제품에 대한 정보를 완벽하게 제공하지 못하고, 모델의 의미적 내용에 따라 반응하지 못함으로써 강인한 조립체 모델로서의 역할을 제대로 수행하지 못하고 있다[9, 16].

본 논문에서는 조립체를 온톨로지를 이용하여 표현함으로써 협업적 환경에서 조립체 설계 정보를 공유할 수 있는 시맨틱 조립체 모델링 프레임워크를 제안한다. 조립체 온톨로지는 조립 부품간의 관계나 설계 의도와 같은 조립체에 대한 정보를 정확하고, 기계가 이해할 수 있도록 표현한다. 본 논문에서는 조립체의 공학적 관계를 표현하기 위해 [11]에서 제안되었던 조립체 관계 모델인 ARM(Assembly Relationship Model)을 온톨로지를 이용하여 강화하여 표현하였다. 조립체 내에 내포된 공학적 관계들, 공간적 관계들, 그리고 접합 관계들은 OWL(Web Ontology Language)[26]와 SWRL(Semantic Web Rule Language)[27]로 표현되며, 추론을 통해 접합 의도, 공간적 관계 등의 설계 의도가 파악된다.

본 논문의 구성은 다음과 같다. 2장에서는 조립체의 정형

화와 조립체 정보 공유에 관련된 기존의 연구를 살펴보고, 3장에서 조립체 관계 모델(Assembly Relational Model, ARM)의 특징과 문제점을 살펴본다. 4장에서는 온톨로지 기반 조립체 모델링 방법을 제안하고, 5장에서는 조립체 지식의 공유를 통한 협업적 제품 개발 프레임워크를 보인다. 6장에서는 결론과 향후 연구를 살펴본다.

2. 관련 연구

조립체 설계 정형화와 관련하여, Deneux[33]는 복합적 조립체 설계에서 조립체 정보를 표현하는 조립체 피쳐의 필요성에 대해 논하였다. Holland and Bronsvort[34]는 단일 부품 피쳐 모델의 개념과 조립체 모델링을 위한 조립체 피쳐 모델을 제안하였다. 그러나 이들의 조립체 피쳐의 개념은 기계적으로 결합된 조립체만을 대상으로 하기 때문에 다른 방법의 결합 과정에서 요구되는 조립체 모델링에는 적용될 수 없다는 문제점이 있다. Whitney, *et al.*[28]는 완전히 제약된 조립체와 부조립체를 중심으로 조립체 설계의 정형화를 제안하고 있다. 이 연구는 일반적인 조립체 설계를 위한 방법을 제안하고 있지만, 공간관계나 결합 과정에서 나타나는 조립체의 상호 영향에 대해서는 논의되지 않고 있다.

Rémondini *et al.*[23]은 기하학적 모델과 분석 모델 사이를 연결할 수 있는 조립체 연산자를 제안하고 있다. 그러나 이 연구는 기계적 분석 모델의 기하학적인 면만을 강조하고 있고, 조립체의 결합방법에 관한 정보를 획득할 수 있는 정형화를 제시하지 않았다. Fu, *et al.* [6]는 기하학적 피쳐를 표현하기 위해 그래프기반 문법을 사용하였다. 이 방법은 단일 부품의 기하학적 피쳐만을 지원하고 있기 때문에 조립체와 기하학적 피쳐들사이의 결합관계를 표현하는데 한계가 있다.

Kim *et al.*[11]는 [1][19]에서 제안한 공간 관계와 피쳐들에 기반하여 정형화된 조립체 모델을 제안하였다. 이 논문에서는 조립체와 조립체 내의 결합 관계를 표현하고, 메타 모델인 조립체 관계 모델인 ARM(Assembly Relation Model)을 생성한다. 조립체 관계 모델은 XML 형식으로 관계를 명확하게 표현한다. 그러나 이 방법은 내포된 제약사항과 관계들을 표현하는데 한계를 보이고 있다. 본 논문에서는 이상의 연구들에서 보인 단점을 해결하기 위해 조립체를 온톨로지를 기반으로 모델링하는 방법을 제안한다. 온톨로지를 기반으로 조립체를 모델링함으로써 내포된 관계를 추론하여, 설계자의 설계의도를 정확하게 파악할 수 있다.

온톨로지는 Tim Berners-Lee[25]가 제안한 시맨틱 웹에서 용어를 정의하고 용어들 사이의 관계를 정형화함으로써 데이터가 공유되고, 재사용 될 수 있도록 지원한다. 온톨로지는 다양한 도메인에서 개발되었다. CYC 온톨로지[4]는 음성인식, 데이터 마이닝, 자연어처리 등에 이용하기 위해 200,000개의 일반 용어들을 포함하는 온톨로지로 구축되었다. 미 표준 과학원에서 제조 도메인에서의 온톨로지를 구축하는데 기본 온톨로지인 CYC를 선택하였고[24], 이 연구

의 결과로 프로세스 명세 언어[8]가 개발되었다. 그러나 CYC 온톨로지는 일반 용어를 대상으로 구축되어있기 때문에 본 논문에서 대상으로 하고 있는 기계 설계 도메인에 이용되기에는 한계를 보인다.

공학 도메인에서 [17]은 공학적 요구사항과 관련된 공학적 작업으로부터 지식을 획득하기 위한 KAD(Knowledge Aided Design) 시스템을 객체지향 방법으로 개발하였다. 이 연구에서는 특히 의사소통, 추적, 완전함, 일관성, 문서 생성, 수정사항관리 등이 중점적으로 논의되었다. 개념적이고 세부적인 설계 수준에 적용된 온톨로지에 관한 연구로는 [12-15] 등이 있다. 이들 연구는 기능적 설계를 표현하기 위한 온톨로지를 개발하여 실제 응용에 적용하였지만 복잡한 기계적 현상을 찾아내는데 적합하지 않다.

이상과 같이 기존에 개발된 온톨로지들은 일반 용어를 대상으로 하였거나 특정 도메인을 대상으로 구축되었기 때문에 기계 설계 도메인에 이용되기는 부적합하다. 또한, 실제 응용을 위해 구축된 온톨로지들도 기계적 조립체와 조립체 내에 내포된 관계를 표현하는데 한계를 보인다.

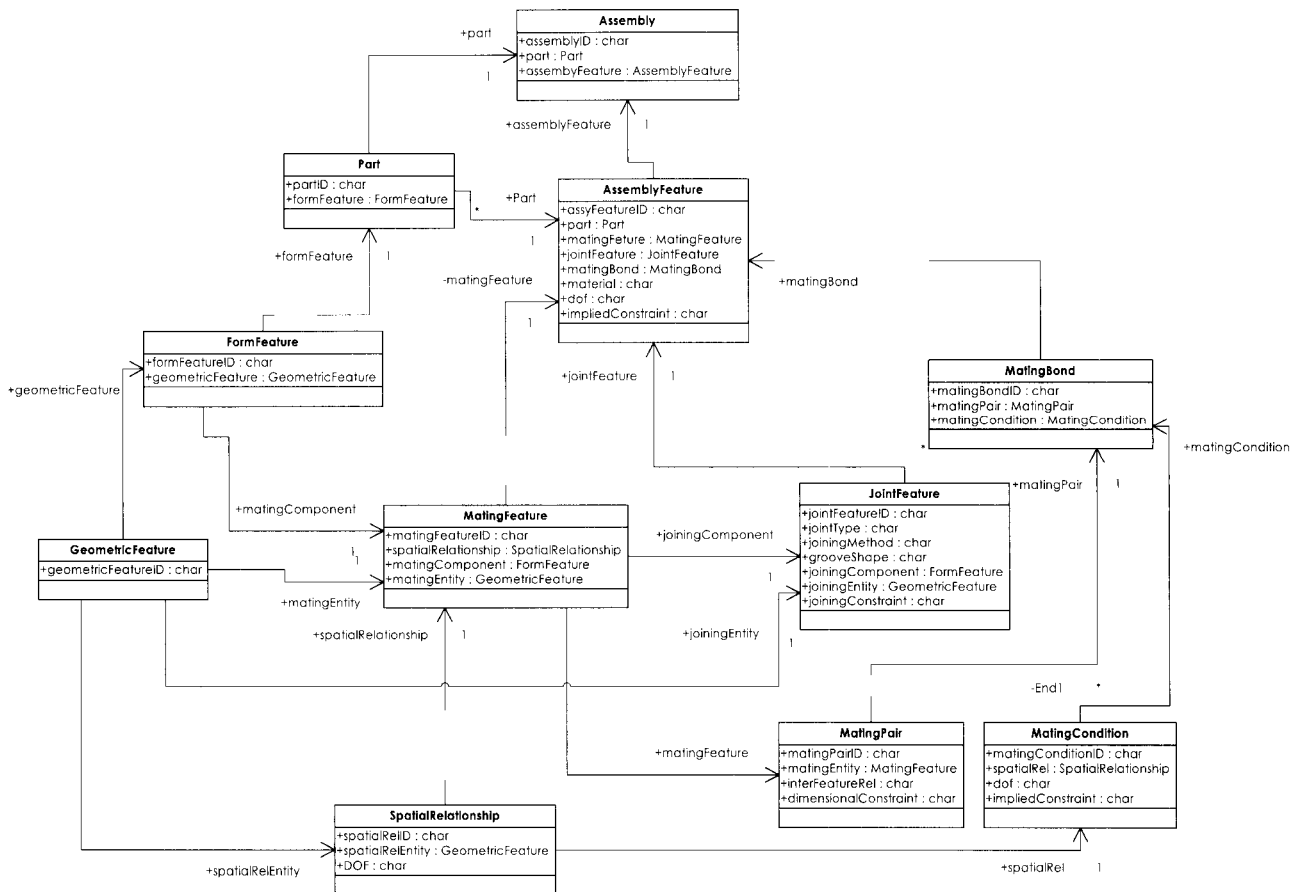
3. 조립체 관계 모델(Assembly Relationship Model)

ARM(Assembly Relationship Model)은 조립체의 공학적 관계를 표현하기 위해 [11]에서 제안된 조립체 관계 모델이

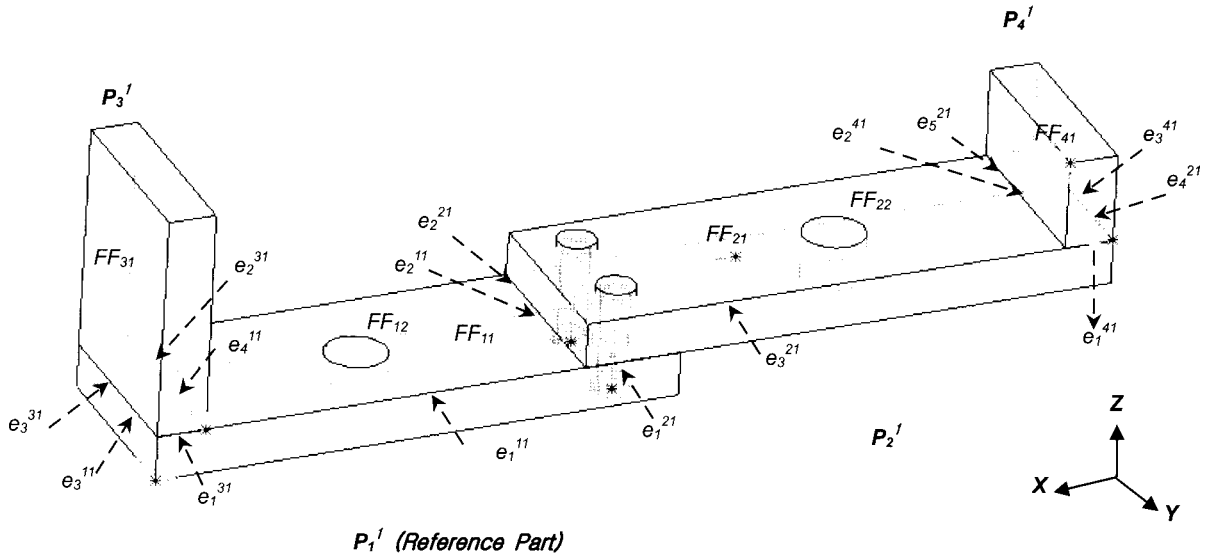
다. ARM(Assembly Relational Model)은 조립체의 정형화를 통해 생성된 XML 기반 메타 모델로서 각 설계자마다 다른 CAD툴의 데이터 형식과 독립적으로 협업적 환경에서 조립체의 정보를 교환하는데 사용된다. ARM내의 모든 기하학적 요소들은 대응되는 솔리드 모델에 연결되어 있고, 정형화의 결과로 CAD 파일내의 기하학적인 요소를 참조하는 XML 파일이 생성된다. ARM의 UML(Unified Modeling Language) 정적 구조는 (그림 1)과 같다. 조립체는 파트와 조립체 피쳐들과 'part_of' 관계이고, 파트는 폼피쳐와 'part_of' 관계이다. 조립체 피쳐는 접합피쳐, 결합 피쳐, 소재, 자유도 등을 속성으로 갖는다.

<표 1> 연결 조립체에 사용된 표기법 설명

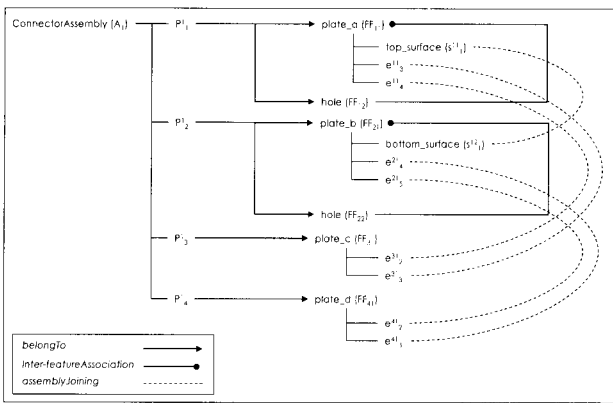
표기	설명
e^{jk}	i^{th} edge of FF_{jk} (FF_{jk} is Form Feature of Part (P_j) and P_j is a part of assembly (A_i))
$P_1^1, P_2^1, P_3^1, P_4^1$	plate_a, plate_b, plate_c, plate_d
FF_{11}	block (length, width, height) = block (L_{11}, W_{11}, H_{11}) = block (110, 40, 10)
FF_{21}	block (L_{21}, W_{21}, H_{21}) = block (110, 40, 10)
FF_{31}	block (L_{31}, W_{31}, H_{31}) = block (50, 40, 10)
FF_{41}	block (L_{41}, W_{41}, H_{41}) = block (20, 40, 10)
FF_{12}	hole (diameter, depth) = hole (DM_{12}, DT_{12}) = hole (12.81, 10)
FF_{22}	hole (DM_{22}, DT_{22}) = hole (12.81, 10)



(그림 1) ARM의 UML 표현



(그림 2) 연결 조립체



(그림 3) 연결조립체의 피쳐 계층과 조립관계

(그림 2)는 본 논문에서 ARM과 온톨로지 기반 조립체 모델링에 사용될 연결 조립체의 예이다. 이 연결 조립체는 두 개의 용접 조인트와 하나의 핀 조인트로 이루어져있다.

이 예에서 *plate_a*와 *plate_b*는 두개의 버튼 리벳을 이용하여 접합되어있다. *Plate_a*의 *top_surface*와 *plate_b*의 *bottom_surface*는 대치(*against*) 관계이다. 리벳은 설계자가 지정한 위치의 홀의 중심선을 따라 정렬된다. *Plate_a*와 *plate_c*는 용접 방법 중의 하나인 GMAW(Gas Metal Arc Welding) 방식으로 접합되어 있고, 접합 피쳐 요소들은 *plate_a*의 *top-surface*와 *plate_c*의 *bottom_surface*이다. (그림 2)의 연결 조립체에서 사용된 기호들은 <표 1>에 설명되어있다.

공간 관계는 부품들 사이의 상대적 위치를 기술하기 위해 제안되었다[1, 18]. 공간 관계에는 '*against*', '*coplanar*', '*fits*', '*parax*', '*lin*', '*rot*', '*fix*' 등이 포함된다. 이들 공간관계는 부품들의 배치 관계를 표현하거나, 기계적 부품 명세를 표현함으로써 일반적 조립체에 적용될 수 있다. 각 공간 관계를 통해 조립체의 위치를 추론할 수 있을 뿐만 아니라 설계자

의 의도를 파악할 수 있다. 즉, 공간 관계에 의해 접합 상태에서의 자유도가 제한되며, 자유도는 선형, 회전, 원형, 평면, 원통, 구형 등으로 분류되어 각 축과 함께 표기된다.

(그림 3)에서는 (그림 2)의 연결 조립체를 구성하는 피쳐들의 구성도를 보이고 있다. 연결 조립체는 네개의 파트들로 이루어져있고, 파트들은 품피쳐들로 구성되어있으며, 각 품피쳐들이 특정 관계를 가지고 연결되어있다. 예를 들어, 품피쳐 *FF11*와 품피쳐 *FF21*은 '*inter-featureAssociation*' 관계를 갖는다.

ARM은 조립체에 대한 정형화 방법을 제안하고 있지만 설계과정에서 나타나는 설계자의 설계의도를 정확히 표현할 수 없고, 상대 협업 설계자와의 정보 공유시 필요한 정보만을 선택적으로 전달할 수 없다는 단점이 있다. 따라서 본 논문에서는 이와 같은 단점을 해결하기 위해 시맨틱 웹에서 여러 사용자간의 공통된 관점을 유지하고 기계 관독성을 제공하는 온톨로지를 도입하여 조립체를 모델링한다.

4. 온톨로지 기반 조립체 모델링

본 장에서는 온톨로지 기반 조립체의 모델링을 제안한다. 본 논문에서 제안하는 프레임워크에서는 제품의 특성, 공정, 접합 등에 관한 정보가 온톨로지를 기반으로 구축되고, 각기 다른 e-Tool을 사용하고 있는 협업 설계자들이 의미적인 질의를 통해 조립체의 정보를 접근한다. 온톨로지 기반 조립체 모델은 기하학 모델들 사이의 연결과 공학적 관계를 포함한 다양한 조립체 지식을 표현하고, 추론 엔진에 의해 새로운 사실들을 유추하는데 이용된다.

4.1 조립체 표현을 위한 클래스와 속성의 구축

조립체 모델링을 위한 온톨로지 용어를 수집하기 위해 프로덕트(Product), 조립체(Assembly), 조립체 부품(Assembly

〈표 2〉 조립체 설계를 위한 용어 분석

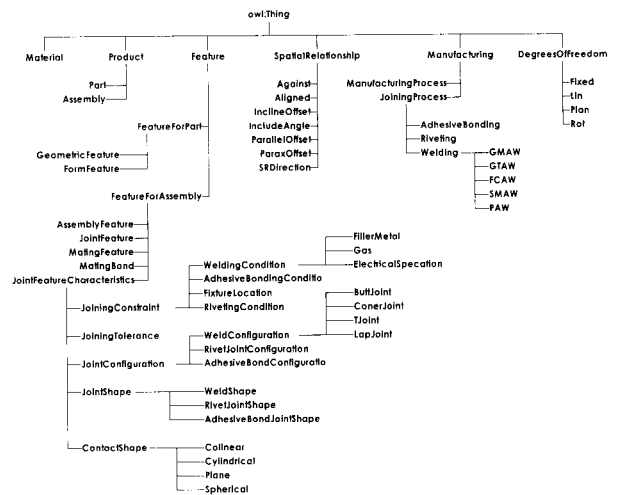
용어	기계 설계 도메인에서의 정의	기계 설계 도메인에서의 다른 가능한 용어	개념의 표현	개념의 표현 내용
프로덕트 (Product)	기능적 요구를 만족하기 위해 설계되고, 개발되고, 제조되고, 조합된 아이템	공학적 프로덕트, 기계적 프로덕트, 소비자 프로덕트, 산업적 프로덕트	드로잉(Drawing), 명세(description), 사양(specification)	선호도의 명세, 구조, 소재 등
조립체 (Assembly)	특정 기능을 위해, 특정 조건하에서 한 단위로 구성되기 위해 모여있는 부품의 그룹	프로덕트 분할 구조	드로잉, 명세, 사양	부품의 명세, 기하학적 위상, 공간관계 등
파트(Part)	개별적 제조 부품	부품, 조립체 부품	드로잉, 명세, 사양	기하학적 위상, 설계의도
조립체 피쳐 (Assembly Feature)	접합 피쳐들 사이의 공간관계 집합, 접합본드, 소재, 자유도, 제약사항등과 같은 접합관계와 조립 결합관계와 연관된 폼 피쳐와 이음매 피쳐들의 정보를 포함하는 조립체 정보 그룹.	없음	데이터구조	공간관계의 명세, 접합 피쳐, 접합본드, 소재 등.
폼피쳐 (Form Feature)	결합 관계 등의 명세와 함께 제공되는 공학적 기능적 표현, 예지, 정점 등과 같은 기하학 엔티티들의 집합.	없음	데이터구조	기하학 정보의 명세, 결합 정보, 공학적 기능
이음매 (Joint)	두 개 이상의 조립체 부품 사이의 접촉면과 접촉 방법	없음	드로잉, 명세, 사양	기하학 정보 명세, 결합조건

Component), 파트(Part), 서브조립체(sub-Assembly), 조립체 피쳐(Assembly feature), 피쳐(Feature), 폼피쳐(Form Feature), 조인트(Joint), 조인트피쳐(Joint Feature), 접합피쳐(Mating Feature) 등의 용어에 대해 설계 도메인에서의 정의, 일반적인 정의, 관련어, 표현 방법 등을 수집하였다. <표 2>에서는 이 중 일부 용어에 대해 수집된 내용을 보이고 있다. 예를 들어, 조립체 피쳐에 대해 살펴 보면, 기계 설계 도메인에서의 정의는 조립체에 대한 정보들로서 폼피쳐와 이음매 피쳐들의 접합관계와 결합 관계, 소재, 공간관계, 자유도, 기타 제약 사항 등을 포함하고 있다.

이상의 설계 용어들로부터 개념을 나타내는 클래스와 클래스의 특징을 나타내거나 클래스간의 관계를 나타내는 속성과 각 클래스에 속하게 될 인스턴스들이 추출된다. (그림 4)의 조립체 모델링을 위해 구축된 클래스 계층도에서 프로덕트의 하위클래스로 파트와 조립체가 정의되고, 피쳐의 하위클래스로 파트를 위한 피쳐와 조립체를 위한 피쳐 클래스가 정의된다.

<표 3>에서는 클래스의 특징을 나타내는 속성들과 클래스간의 관계를 나타내는 속성들의 도메인과 범위(range)를 나타내고 있는데 'belongTo' 속성은 도메인이 폼피쳐(FormFeature)이고, 범위는 파트(Part)임을 나타낸다.

본 논문에서는 OWL(Web Ontology Language)를 이용하여 조립체를 모델링하고, 온톨로지 구축 툴은 protégé-3.1을 사용한다. OWL은 W3C에서 권고안으로 발표된 차세대 웹 온톨로지 언어로 기존의 RDF/RDFS, DAML+OIL에 비해 클래스와 관계에 대하여 기술 할 수 있는 더 많은 어휘를 제공한다. 예를 들면, 클래스간의 관계, 관계차수, 동치성, 풍부한 속성 타입, 열거형 클래스 등을 기술할 수 있어 기존의 온톨로지 언어보다 의미적 표현력이 상대적으로 뛰어나다.



(그림 4) 조립체 설계를 위한 온톨로지 클래스

〈표 3〉 클래스 속성

Property	Domain	Range
belongTo	FormFeature	Part
belongToFormFeature	GeometricFeatures	FormFeature
hasMaterial	Part	Material
hasPart	Assembly	Part
referenceDirection	MatingFeature	SRDirection
hasMatingComponent	MatingFeature	FormFeature

4.2 조립체의 관계 표현

조립체내의 관계 중 폼피쳐(Form Feature)와 폼피쳐 사이의 관계와, 폼피쳐와 파트(part) 사이의 관계는 OWL(Web Ontology

Language)과 SWRL(Semantic Web Rule Language) 규칙으로 모델링 되는데, 이때 각 관계에서 나타나는 제약 사항들을 고려하여 온톨로지를 구축한다. SWRL은 시맨틱 웹을 위한 규칙 명세 언어로 제안되었고, OWL의 하부 언어인 OWL DL 및 OWL Lite와 RuleML의 하부 언어인 Unary/Binary Datalog RuleML을 통합한 언어이다. SWRL은 유사 혼 규칙(Horn-like rules)을 포함시켜 OWL의 공리(axiom)를 확장함으로써 유사 혼 규칙을 OWL 지식 베이스와 통합시킬 수 있도록 하였다.

본 논문에서 조립체의 관계는 파트와 폼피처에서 발생하는 'belongTo' 관계와 두 폼피처 사이에서 발생하는 'inter-featureAssociation'과 'assembly/joining' 관계를 OWL과 SWRL 규칙으로 모델링 한다. 세 관계를 모델링하기 위해 본 논문에서는 [11]에서 제안한 정형화된 관계의 표현과 이에 대응되는 OWL과 SWRL 규칙을 보인다.

세 관계를 정의하기 위해 먼저 A 는 조립체의 집합, P 는 파트의 집합, FF 는 폼피처의 집합이라 하고, 함수 $comp()$ 는 각 객체의 구성 요소를 구하는 함수로 정의한다. 예를 들어, $comp(A_i)$ 는 $A_i \in A$ 를 구성하는 파트들 $P_i \in P$ 를 구한다.

- 1) 'belongTo' 관계는 파트와 폼피처 사이에서 다음과 같이 정의한다.

$P_j^i \in comp(A_i), j = 1, 2, \dots, m$ 이고,
 $FF_{jk} \in comp(P_j^i)$ 일 때 파트 P_j^i 와 폼피처 FF_{jk} 는 'belongTo' 관계를 갖고 다음과 같이 표현한다.

$$B_{jk}^{(i)}: FF_{jk} \rightarrow P_j^i, k = 1, 2, \dots, n$$

'belongTo' 관계를 OWL 온톨로지로 표현하기 위해서는 다음과 같은 두가지 제약 사항을 고려해야 한다.

- CI-1: 모든 폼피처는 반드시 파트에 포함되어야 한다.
 CI-2: 한 폼피처는 두개의 파트에 포함될 수 없다.

첫 번째 제약 사항인 '모든 폼피처는 반드시 파트에 포함되어야 한다'는 제약은 존재 한정자(existential quantifier)를 이용하여 $\exists FormFeature \text{ belongTo } Part$ 로 표현한다. protégé-3.1로 구축할 경우 폼피처 클래스의 'asserted condition'을 명시함으로써 표현된다. 두 번째 제약사항인 '폼피처는 하나의 파트에만 포함되어야 한다'는 제약은 해당 속성을 통해서 하나의 인스턴스만 가질 수 있도록 제한하며, protégé-3.1에서는 'belongTo'의 속성으로 함수적(Functional) 속성을 지정함으로써 표현된다.

- 2) 'inter-featureAssociation' 관계는 폼피처와 폼피처 사이에서 다음과 같이 정의 된다.

파트 $P_j^i \in comp(A_i), j = 1, 2, \dots, m$ 와 $FF_{jp}, FF_{jq} \in comp(P_j^i)$ 가 'belongTo' 관계($FF_{jp}, FF_{jq} \rightarrow P_j^i$)이고, FF_{jp} 와 FF_{jq} 에 대해 차원제약(DC_r)과 폼피처 사이의 관계제약(RC_{pq})이 만족될 때 FF_{jp} 와 FF_{jq} 는 'inter-featureAssociation' 관계를 갖고 다음과 같이 표기 된다.

$$I_{pq}^{(j)}: FF_{jp} \Leftrightarrow FF_{jq}, p = 1, 2, \dots, n, q = 1, 2, \dots, l$$

관계제약(Relational Constraints), RC_{pq} 은 한 폼피처가 다른 폼피처를 포함한다면 '0'으로 정의하고, 다른 폼피처에 포함 된다면 '1', 두 폼피처 사이의 포함 관계가 없다면 '2'로 정의한다. 두 폼피처 사이의 거리는 차원제약(dimensional constraint)이다. 'inter-featureAssociation' 관계는 다음과 같은 제약사항을 내포하고 있다.

- C2-1: 폼피처들은 서로 다른 폼피처이어야 한다.
 C2-2: 폼피처들이 같은 파트에 포함되어있어야 한다.
 C2-3: 관계제약이 만족되어야 한다.

'inter-featureAssociation' 관계를 표현하기 위한 SWRL 규칙은 다음과 같다. 즉, 'inter-featureAssociation' 규칙은 두 폼피처가 서로 다르면서, 하나의 파트에 포함되고 세가지 관계제약 중 하나를 만족하는 두 폼피처 사이에 정의된다.

- $FormFeature(?x) \wedge FormFeature(?y) \wedge Part(?z) \wedge differentFrom(?x, ?y) \wedge belongTo(?x, ?z) \wedge belongTo(?y, ?z) \wedge relationalConstraint_0(?x, ?y \rightarrow inter-featureAssociation(?x, ?y)$
- $FormFeature(?x) \wedge FormFeature(?y) \wedge Part(?z) \wedge differentFrom(?x, ?y) \wedge belongTo(?x, ?z) \wedge belongTo(?y, ?z) \wedge relationalConstraint_1(?x, ?y) \rightarrow inter-featureAssociation(?y, ?x)$
- $FormFeature(?x) \wedge FormFeature(?y) \wedge Part(?z) \wedge differentFrom(?x, ?y) \wedge belongTo(?x, ?z) \wedge belongTo(?y, ?z) \wedge relationalConstraint_2(?x, ?y) \rightarrow inter-featureAssociation(?x, ?y)$

- 3) 'assembly/joining' 관계는 서로 다른 파트에 포함된 폼피처들 사이의 관계를 표현하며 다음과 같이 정의된다.

파트 $P_g^i, P_h^i \in comp(A_i), g = 1, 2, \dots, m_1, h = 1, 2, \dots, m_2$ 이고, 두 폼피처 $FF_{gp} \in comp(P_g^i), p = 1, 2, \dots, l_1; FF_{hq} \in comp(P_h^i), q = 1, 2, \dots, l_2$ 이며, FF_{gp}, FF_{hq} 가 결합 관계를 가질 때, 한 폼피처 FF_{gp} 와 다른 폼피처 FF_{hq} 는 'assembly/joining' 관계를 갖고 다음과 같이 표기한다.

$$\delta_{pq}^{(gh)}: FF_{gp} \otimes FF_{hq}$$

위 정의에 내포된 'assembly/joining' 관계의 제약사항은 다음과 같다.

- C3-1: 두 폼피처는 반드시 서로 다른 파트에 포함되어야 한다.
 C3-2: 두 폼피처는 반드시 결합 쌍(joining pair)이어야 한다.

위의 제약 사항을 포함한 assembly/joining 관계는 SWRL 규칙으로 다음과 같이 정의한다.

- $FormFeature(?x) \wedge FormFeature(?y) \wedge Part(?z) \wedge Part(?a) \wedge belongTo(?x, ?z) \wedge belongTo(?y, ?a) \wedge differentFrom(?z, ?a) \wedge isJointPair(?x, ?y) \rightarrow assemblyJoiningRelationship(?x, ?y)$

SWRL로 명세된 규칙의 실행은 ETRI에서 개발된 RETE 기반 추론 엔진[30]을 이용하였으며, 'assembly/joining' 규칙 수행 결과로 생성된 새로운 사실들의 예는 다음과 같다.

- fact ns_0:Fact1029 is ns_2:assemblyJoiningRelationship (ns_2:FormFeature_11, ns_2: FormFeature_21);
- fact ns_0:Fact1027 is ns_2:assemblyJoiningRelationship (ns_2:FormFeature_21, ns_2: FormFeature_11);
- fact ns_0:Fact1031 is ns_2:assemblyJoiningRelationship (ns_2:FormFeature_11, ns_2: FormFeature_31);
- fact ns_0:Fact1026 is ns_2:assemblyJoiningRelationship (ns_2:FormFeature_31, ns_2: FormFeature_11);
- fact ns_0:Fact1030 is ns_2:assemblyJoiningRelationship (ns_2:FormFeature_41, ns_2: FormFeature_21);
- fact ns_0:Fact1028 is ns_2:assemblyJoiningRelationship (ns_2:FormFeature_21, ns_2: FormFeature_41);

위의 추론 결과에서 중복을 제외하면 다음의 세 사실들이 'assembly/joining' 관계로 새롭게 추론된다.

- assemblyJoiningRelationship (FormFeature_11, FormFeature_21)
- assemblyJoiningRelationship (FormFeature_31, FormFeature_11)
- assemblyJoiningRelationship (FormFeature_21, FormFeature_41)

4.3 규칙을 이용한 자유도 추론

조립체 설계 시 두 파트의 공간 관계에 따라 자유도가 유추될 수 있는데, 본 논문에서는 SWRL 규칙을 수행하여 자유도를 추론한다. 본 절에서는 공간 관계와 자유도가 유추될 수 있는 경우들과 이를 SWRL 규칙으로 표현하고 수행한 결과를 살펴본다.

z축을 따라 평면(planar) 표면상에서 'against' 공간 관계를 갖는 두 파트는 평면과 회전자유도를 갖는다. 즉, 참조파트의 좌표계의 z축을 따라 평면(plane) 자유도와 자신의 좌표축의 z축을 따라 회전 자유도를 갖는다. 다음은 이러한 자유도를 추론할 수 있는 SWRL 규칙이다.

- $MatingFeature(?x) \wedge FormFeature(?y) \wedge Part(?z) \wedge belongTo(?y, ?z) \wedge hasMatingComponent(?x, ?y) \wedge hasSpatialRelationship(?x, against) \wedge hasContactShape(?x, Plane_Z) \wedge ReferenceDirection(?x, Z-Direction) \rightarrow hasDOF(?z, Plan_Z)$
- $MatingFeature(?x) \wedge FormFeature(?y) \wedge Part(?z) \wedge belongTo(?y, ?z) \wedge hasMatingComponent(?x, ?y) \wedge hasSpatialRelationship(?x, Against) \wedge hasContactShape(?x, Plane_Z) \wedge ReferenceDirection(?x, Z-Direction) \rightarrow hasOwnDOF(?z, Rot_Z)$

위의 규칙들로부터 추론 엔진을 통해 생성된 새로운 사실들의 예는 다음과 같다.

- fact ns_0:Fact151387 is ns_3:hasDOF

(ns_3:Part_12, ns_3:Plan_Z);

- fact ns_0:Fact151429 is ns_3:hasOwnDOF (ns_3:Part_14, ns_3:Rot_Z);
- fact ns_0:Fact151385 is ns_3:hasDOF (ns_3:Part_13, ns_3:Plan_Z);
- fact ns_0:Fact151430 is ns_3:hasOwnDOF (ns_3:Part_12, ns_3:Rot_Z);
- fact ns_0:Fact151386 is ns_3:hasDOF (ns_3:Part_14, ns_3:Plan_Z);
- fact ns_0:Fact151428 is ns_3:hasOwnDOF (ns_3:Part_13, ns_3:Rot_Z);

두 파트가 x축을 따라 동일 직선상에서 정렬되어 있다면, 선형과 회전 자유도를 갖는다. 즉, 참조파트의 좌표계의 x축을 따라서 선형 자유도를 갖고, 자신의 좌표축의 x축을 따라 회전 자유도를 갖는다. 이와 같은 자유도는 다음과 같은 SWRL 규칙으로 표현된다.

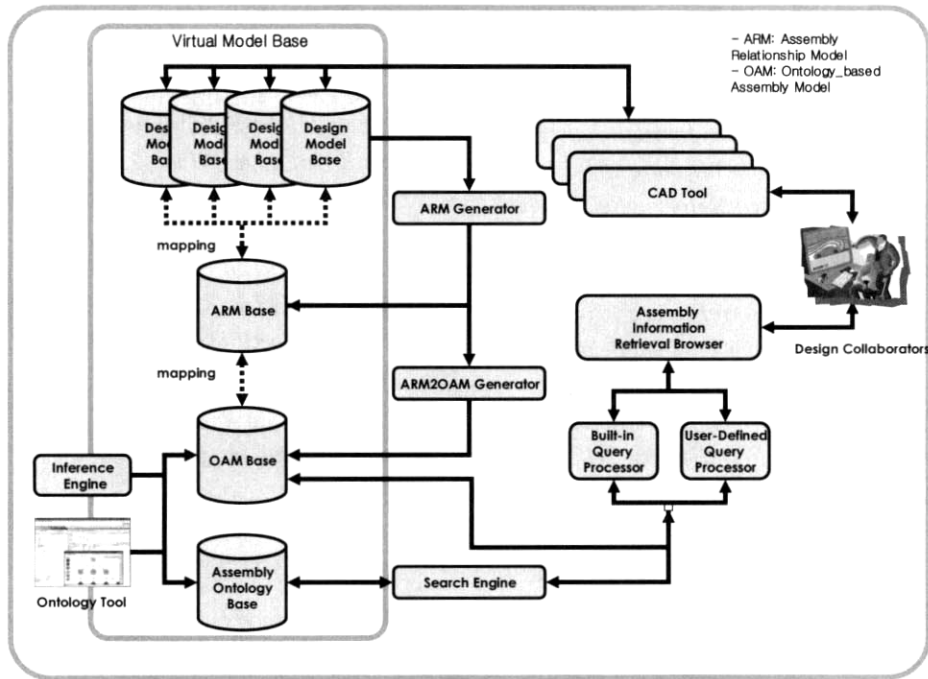
- $MatingFeature(?x) \wedge FormFeature(?y) \wedge Part(?z) \wedge belongTo(?y, ?z) \wedge hasMatingComponent(?x, ?y) \wedge hasSpatialRelationship(?x, Aligned) \wedge hasContactShape(?x, Collinear) \wedge ReferenceDirection(?x, X-Direction) \rightarrow hasDOF(?z, \wedge Lin_X)$
- $MatingFeature(?x) \wedge FormFeature(?y) \wedge Part(?z) \wedge belongTo(?y, ?z) \wedge hasMatingComponent(?x, ?y) \wedge hasSpatialRelationship(?x, Aligned) \wedge hasContactShape(?x, Colinear) \wedge ReferenceDirection(?x, X-Direction) \rightarrow hasOwnDOF(?z, Rot_X)$

회전 자유도와 두 선형 자유도를 갖는 파트는 고정된 설계 자유도를 갖는다. 이를 위한 SWRL 규칙은 다음과 같다.

- $Part(?x) \wedge hasDOF(?x, Rot_Z) \wedge hasDOF(?x, Lin_X) \wedge hasDOF(?x, Lin_Y) \wedge hasDOF(?x, Fixed_in) \rightarrow hasDesignedDOF(?x, Fixed_in)$

결합 방법에 의해 자유도가 유추될 수 있다. 예를 들어, 용접(welding)에 의해 결합된 파트는 용접에 의해 유추되는 고정된 자유도를 갖는다. 또한 만약 파트가 하나 이상의 리벳에 의해 조인된다면 이 파트는 고정된 자유도를 갖고 이것을 결합 방법인 리벳으로부터 유추할 수 있다. 다음은 이를 위한 SWRL 규칙이다.

- $JointFeature(?y) \wedge FormFeature(?z) \wedge Part(?b) \wedge isJoiningMethod(?y, Welding) \wedge hasJoiningComponents(?y, ?z) \wedge belongTo(?z, ?b) \rightarrow hasJoiningInferredDOF(?b, fixed_In)$
- $JointFeature(?y) \wedge FormFeature(?z) \wedge Part(?b) \wedge isJoiningMethod(?y, Riveting) \wedge hasJoiningComponents(?y, ?z) \wedge belongTo(?z, ?b) \wedge RivetingCondition(?c) \wedge hasJoiningConstraint(?y, ?c) \wedge NumberOfRivet(?c, ?d) \wedge swrlb:greaterThanOrEqual(?d, 2) \rightarrow hasJoiningInferredDOF(?b, fixed_In)$



(그림 5) 조립체 정보 공유 프레임워크

5. 조립체 정보 공유 프레임워크

본 논문에서는 조립체 검색 시스템을 자바와 SWT(standard widget toolkit)을 이용하여 개발하였다. 조립체 검색 시스템은 협업적 설계 환경에서 필요한 정보만을 필터링하여 전달함으로써 제품에 대한 전체 정보로부터 수동적으로 필요한 정보를 찾아야 하는 기존 시스템들의 단점을 해결한다. 예를 들어, (그림 2)의 연결 조립체를 개발하는데, 설계자 A가 설계자 B가 설계한 부품에 연결될 다른 부품을 설계하는 상황에서, 설계자 A가 전체 제품에서 자신이 설계한 부품의 성능을 측정해보고자 한다면, 협업 설계자 B가 설계한 전체 제품에 대한 정보를 필요로 하지 않는다. 그러나 기존의 협업 설계에서는 전형적으로 전체 설계에 대한 정보를 전달함으로써 불필요한 정보까지 공유하게 되어 수동적으로 필요한 정보를 검색해야 하는 단점이 있다.

(그림 5)에서는 본 논문에서 제안하는 조립체 정보 공유를 위한 프레임워크를 보이고 있다. 먼저, 각 CAD 툴을 사용하여 생성된 조립체 모델은 ARM 생성기를 통해 ARM으로 변환되고, CAD 모델과 ARM 정보는 매핑된다. 온톨로지 구축 툴에 의해 조립체 온톨로지가 생성되고, ARM2OAM 생성기에 의해 온톨로지 기반 조립체 모델(Ontology-based Assembly Model, OAM)이 생성되고, ARM 정보와 매핑된다. 추론 엔진은 조립체 온톨로지내의 규칙을 수행하여 새롭게 생성된 사실들을 OAM 베이스에 추가한다. 조립체 정보 검색 시스템의 인터페이스를 통해 주어진 시스템 정의 질의와 사용자 정의 질의는 질의 처리기에 의해 해석되고, 온톨로지를 참조하여 확장된 후 검색엔진으로 전달된다.

조립체 검색 시스템은 온톨로지를 이용하여 설계자들 사이의 상이한 용어에 대한 공통된 관점을 제공하고, 설계자

의 의도가 파악된 설계 정보를 제공함으로써 협업 설계자들이 필요한 정보를 용이하게 검색할 수 있다. 즉, 조립체 설계 정보를 키워드 검색을 이용하여 선택적으로 검색하고자 할 경우, 온톨로지를 이용하여 여러 협업 설계자들의 원격에 위치하고 있는 부품에 대한 정보를 공통된 관점에서 접근할 수 있다. 본 프레임워크에서 조립체 온톨로지는 ARM으로 매핑되고, ARM은 서로 다른 CAD 형식의 관련된 솔리드 모델에 명시적으로 연결되어 있다.

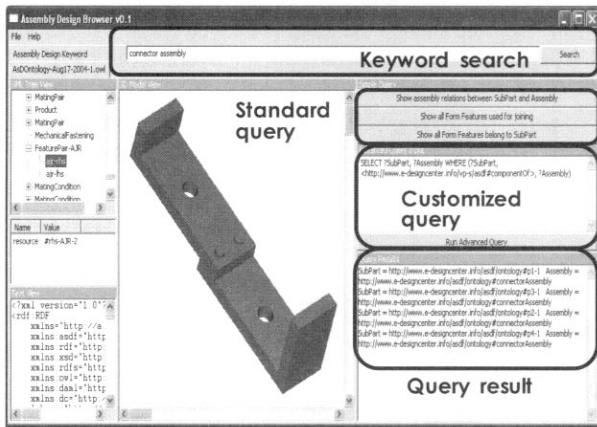
협업 설계 도메인에서 설계 정보를 검색하기 위해 빈번히 질의되는 내용은 다음 <표 4>와 같다. 설계자들의 질의는

<표 4> 설계 정보 질의 예

카테고리	요구되는 정보	질의 예
부품	기하학 정보 (Geometric Information)	각 부품의 차원은 무엇인가?
	소재(Material)	각 부품의 소재는 무엇인가?
	무게(Weight)	각 부품의 무게는 얼마인가?
	허용한계(Tolerance)	홀의 허용한계는 얼마인가?
조립체	조립방법 (Assembly method/operation)	조립방법은 어떤 종류인가?
	조립관계(Assembly relations)	부품들 사이에 어떤 관계가 있는가?
	작동환경 (Working environment)	주변환경에 따른 제약 사항이 있는가? (boundary condition)
	예상동작(Intended function)	임펠러(impeller) 회전수는 얼마인가?
	오류가능(Potential failure)	가장 취약한 연결부위는 어디인가?
	조립체 설계 이유 (Assembly design rationale)	왜 연결이 필요한가?

〈표 5〉 표준 질의 패턴

질의 1:	조립체 X에 포함되는 파트를 보여라.
질의 2:	부품 X와 Y의 조립관계를 보여라
질의 3:	부품 X와 관련 있는 'inter-feature' 관계들을 보여라
질의 4:	부품 X의 자유도나 기하학적인 제약들을 보여라
질의 5:	부품 X와 직접적으로 연결된 부품들을 보여라
질의 6:	부품 X와 직/간접적으로 연결되었거나 관련 있는 부품들을 보여라



(그림 6) 조립체 검색 시스템 인터페이스

크게 부품에 관련된 내용과 조립체에 관련된 내용으로 나누어진다. 부품에 관련된 질의는 기하학적인 정보, 소재, 무게, 오차 허용도에 관한 내용이고, 조립체에 관한 질의는 조립 방법, 관계, 작동환경, 의도된 기능, 오류가능성, 조립체 설계 의도 등에 관한 내용이다. 이상의 질의들은 <표 5>과 같이 전형적인 질의 패턴으로 요약된다.

전통적으로, <표 5>와 같은 질의에 대한 정보는 각 설계 데이터에 해당하는 CAD 툴을 통해서 구하거나 CAD 파일을 STEP이나 IGES 등으로 변환하여 구한다. 따라서 각 해당 CAD 툴이 필요하고, 또한 중간 형태로의 변환 과정 중에 결합 정보와 같은 중요한 조립체 정보가 손실 되는 문제가 발생한다. 그러나 본 논문에서 제안하는 조립체 검색 시스템을 통해 사용자는 특정 CAD 형식을 알 필요가 없고, 결합 정보가 ARM으로 명확히 모델링 되어 있기 때문에 정보의 손실이 적다.

본 논문에서는 제안하는 조립체 정보 검색 시스템의 인터페이스는 (그림 6)과 같다. 사용자는 인터페이스에서 시스템 정의 질의와 사용자 정의 질의를 통해 조립체를 검색하여, 조립체의 설계 정보를 접근한다. 사용자 정의 질의는 RDF를 위한 질의어인 RDQL[31]의 구문을 사용한다.

온톨로지로 모델링된 조립체 요소는 CAD 형식과 응용 툴에 독립적인 ARM에 의해 해당 솔리드 모델에 연결되어 있다. 따라서 협업적 조립체 개발 환경에서 특정 부분을 위해 필요한 정보 집합이 유지되며, 조립체 온톨로지는 선택

적 조립체 정보 공유에 사용된다. 일반적으로 기업내에서의 시스템은 보안과 권한의 계층을 이루고 있기 때문에 협업적 제품 개발에서 정보의 공유는 이러한 보안계층을 고려하여 정보를 전달하여야 한다. 보안 문제는 본 논문의 범위를 벗어나므로 다루지 않지만 조립체 정보 공유를 위해서는 해결되어야 하는 문제이다.

6. 결론

본 논문에서는 조립체 온톨로지를 이용한 협업적 제품 개발 프레임워크를 제안하였다. 조립체의 설계 정보는 온톨로지를 이용하여 기계 가독적인 형식으로 표현되었고, 조립체 정보 검색 시스템을 이용하여 질의되고 추론된다. 품피처와 파트들 사이에 존재하는 복잡한 관계들을 표현하는데 OWL과 SWRL을 사용하였고, 관계와 자유도 등이 온톨로지로 표현되어 협업 설계 동안 표준적인 방법을 통해 해석되고 유지되기 때문에 시스템 일관성과 설계 비용을 최소화 하는데 도움을 준다.

조립체 온톨로지의 구축은 공학적 설계의 고차원적인 시맨틱 모델링에 기여한다. 조립체 설계에 사용된 개념들이 온톨로지를 사용하여 정의됨으로써 조립체를 설명하는 표준 용어가 생성되고, 고차원적인 시맨틱은 조립체 모델의 정보를 추론하는 능력을 컴퓨터가 가질 수 있도록 한다. 이러한 능력은 이질적인 환경과 툴들을 사용하는 협업적 조립체 모델 개발에서 중요한 역할을 한다.

수작업으로 수행되는 조립체 온톨로지의 구축은 사용자에 대한 프로파일 학습 기법이나 키워드의 빈도를 가중치로 계산하여 중요 속성을 선택하는 것과 같은 전처리 과정을 수행하여 반 자동화를 시도해볼 수 있다. 조립체 온톨로지의 반자동 구축에 관한 부분은 향후 연구과제로 남긴다.

참고 문헌

- [1] AP. Ambler, RJ. Popplestone, "Inferring the positions of bodies from specified spatial relationships," Artificial Intelligence 1975, 6(2).
- [2] American Welding Society, Vision for Welding Industry, 2003, <http://www.aws.org/research/vision.pdf>.
- [3] R. Beeson, "Pipeline welding goes mechanized," Welding Journal 1999; 78(11):47-50.
- [4] Cycorp, Inc. 2004; <http://www.cyc.com/cyc>.
- [5] 2005 Engineous International Symposium & Workshop, Novi, MI. USA, October, 10-12, 2005.
- [6] Z. Fu, A. De Pennington, S. Saia, "A graph grammar approach to feature representation and transformation," Int. J. Computer Integrated Manufacturing 1993, 6(1&2):137-51.
- [7] T. Graf, H. Stauer, "Laser-hybrid welding drives VW improvements," Welding Journal 2003, 82(1):42-8.
- [8] M. Gruninger, RD. Sriram, J. Cheng, K. Law, "Process specification language for project information exchange," Int. J.

IT in Architecture, Engineering and Construction 2003, 1:307-28.

[9] I. Horváth, JPW. Pulles, AP. Bremer, JSM. Vergeest, "Towards an ontology-based definition of design features," SIAM Workshop on Mathematical Foundations for Features in Computer Aided Design, Engineering, and Manufacturing, 1998.

[10] K.Y. Kim, DG. Manley, BO. Nnaji, "Lean virtual product model sharing for collaborative virtual prototyping," IIE Annual Conference (IERC), May, 15-19, 2004.

[11] K.Y. Kim, Y. Wang, OS. Muogboh, BO. Nnaji, "Design formalism for collaborative assembly design environment," Computer Aided Design on Distributed CAD 2004; 36(9):849-71.

[12] Y. Kitamura, T. Sano, K. Namba, R. Mizoguchi, "A functional concept ontology and its application to automatic identification of functional structures," Advanced Engineering Informatics 16(2):145-63, 2002.

[13] Y. Kitamura, R. Mizoguchi, "Ontology-based description of functional design knowledge and its use in a functional way server," Expert Systems with Application 24(2):153-66.

[14] Y. Kitamura, R. Mizoguchi, "Ontology-based systematization of functional knowledge," In press for J. Engineering Design 2004.

[15] Y. Kitamura, M. Kashiwase, F. Masayoshi, R. Mizoguchi, "Deployment of an ontological framework of function design knowledge," In press for Advanced Engineering Informatics 2004.

[16] R. Kurland, "NX systems engineering powers the product lifecycle," TechniCom Inc technical article, 2003.

[17] J. Lin, MS. Fox, T. Bilgic, "A requirement ontology for engineering design," Concurrent Engineering Research and Applications, 4(3):279-91, 1996.

[18] HC. Liu, BO. Nnaji, "Design with spatial relationships," J Manufacturing Systems, 10(6), 1991.

[19] TL. Liu, "A coordinated constraint-based modeling and design advisory system for mechanical components and assemblies," Ph. D. dissertation, University of Massachusetts Amherst, 1997.

[20] D. Lutters, AH. Streppel, HJJ. Kals, "The role of information structures in design and engineering processes," 3rd Workshop on Product Structuring, 1997.

[21] RW. Messler, "Joining of Advanced Materials," Boston: Butterworth-Heinemann, 1993.

[22] Pegasus, "NSF I/UCRC for e-Design: Strategic Planning Meeting," Pittsburgh, PA, USA, Dec., 9 to 10, 2003, www.e-designcenter.info.

[23] L. Rémondini, JC. Léon, P. Trompette, "High-level operations dedicated to the integration of mechanical analysis within a design process," Engineering with Computers. 1998, 14:81-92.

[24] C. Schlenoff, R. Ivester, D. Libes, P. Denno, S. Szykman, "An analysis of existing ontological systems for applications in manufacturing and healthcare," NISTIR 6301 National Institute of Standards and Technology 1999.

[25] World Wide Web Consortium, "World wide web - summary," 1992, http://www.w3.org/Summary.html.

[26] World Wide Web Consortium, "OWL web ontology language guide," 2004, http://www.w3c.org/TR/owl-guide.

[27] World Wide Web Consortium, "SWRL: A Semantic Web Rule Language," 2004,

http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/.

[28] DE. Whitney, R. Mantripragada, JE. Adams, SJ. Rhee, "Toward a theory for design of kinematically constrained mechanical assemblies," Int J Robotics Research, 18(12):1235-48, 1999.

[29] DE. Whitney, "A design procedure applicable to different classes of assemblies," Proceedings of the ASME 2001 Design Technical Conferences, September, 9-12, 2001.

[30] Minsu Jang., Joo-chan Sohn., "Bossam: an extended rule engine for the web," Proceedings of RuleML 2004 (LNCS Vol. 3323), Nov., 8, 2004.

[31] Jena Tutorial, "A Programmer's Introduction to RDQL," http://jena.sourceforge.net/tutorial/RDQL/

[32] S. Szykman, RD. Sriram, WC. Regli, "The role of knowledge in next-generation product development systems," J. Computing and Information Sc. in Engr., 2001, 1:3-11.

[33] D. Deneux, "Introduction to assembly features: an illustrated synthesis methodology," J. Intelligent Manufacturing, 10:29-39, 1999.

[34] W. van Holland, WF. Bronsvort, "Assembly features in modeling and planning," Robotics and Computer Integrated Manufacturing, 16:277-94, 2000.

양 형 정



e-mail : hjyang@chonnam.ac.kr
 1991년 전북대학교 전산통계학과(학사)
 1993년 전북대학교 전산통계학과(석사)
 1998년 전북대학교 전산통계학과(박사)
 2003년~2005년 카네기멜런 대학교 연구원
 2005년~현재 전남대학교 전자컴퓨터공학부
 전임강사

관심분야 : e-Design, 데이터 마이닝, e-Learning

김 경 윤



e-mail : kykim@eng.wayne.edu
 1996년 전북대학교 산업공학과(학사)
 1998년 전북대학교 산업공학과(석사)
 2003년 피츠버그대학교 산업공학(박사)
 2003년~2005년 피츠버그대학교 연구교수
 2005년~현재 Wayne State University 조교수

관심분야 : 협업적 설계, CAD/CAM, Telerehabilitation

김 수 형



e-mail : shkim@chonnam.ac.kr
 1986년 서울대학교 컴퓨터공학과(학사)
 1988년 한국과학기술원 전산학과(공학석사)
 1993년 한국과학기술원 전산학과(공학박사)
 1993년~1996년 삼성전자 멀티미디어
 연구소 선임연구원

1997년~현재 전남대학교 전자컴퓨터정보통신공학부 부교수
 관심분야 : 인공지능, 패턴인식, 유비쿼터스컴퓨팅