



한국정보과학회  
KOREAN INSTITUTE OF INFORMATION SCIENTISTS AND ENGINEERS

제35권 제2호

2008 가을

학술발표논문집(B)

Proceedings of  
The 35<sup>th</sup> KIISE Fall Conference



한국정보과학회

KOREAN INSTITUTE OF INFORMATION SCIENTISTS AND ENGINEERS

2008년 10월 24일 ~ 25일 · 중앙대학교

# 목 차 (B)

## 소프트웨어공학

1. VIS 정형 검증 자동화를 위한 소프트웨어 VIS Analyzer 개발	정세훈 · 유준범 · 차성덕	1
2. 지능형 시뮬레이션 로봇을 위한 멀티센서 기반 지능 컴포넌트 개발 연구	홍성용	6
3. 모바일 환경을 위한 경량의 웹 서비스 프레임워크	김연석 · 이경호	11
4. 신뢰와 평판 기반의 웹 서비스 선택	하미수 · 이경호	17
5. Embedded System을 위한 Component and Connection 기반의 Component Model : SESA	이종인 · 양현곤 · 김준길	22
6. 커뮤니티 컴퓨팅의 역할 할당 방법	김희수 · 이정태 · 김만구	27
7. ECA 규칙과 확장된 UML을 통한 서비스 지향 적응형 시스템의 모의 실험 환경 모델링	삼재근 · 이병정 · 우치수	33
8. UML 다이어그램기반 Z명세서 작성 방법론	조병일 · 윤현상 · 이은석	38
9. e-엔지니어링 프레임워크에서 그리드 서비스 접근을 위한 그리드 서비스 게이트웨이 설계	김동욱 · 국승학 · 김현수 · 이재경 · 박성환	43
10. Alloy를 이용한 CC3.1 기반의 EAL7등급 FSP 정형 명세 및 분석	신지훈 · 최진영	49
11. 보호관찰서비스 : 유비쿼터스 컴퓨팅 환경에서의 통합 서비스 시스템과 그 운용 모델	이건수 · 김만구	55
12. 유비쿼터스 컴퓨팅 환경에서의 서비스 모델과 그 모델을 사용한 과금 방법	이건수 · 김만구	59
13. 프레임워크 기반 웹 어플리케이션을 위한 JUnit 코드 생성에 대한 고찰	송화정 · 최병주 · 정상인	63
14. 계층형 큐잉 페트리 넷을 이용한 가용성 분석 방법론	전금재 · 윤현상 · 이은석	69
15. Embedded System에서의 BUS 와 Connection을 기반한 Component Architecture	양현곤 · 이종인 · 김준길	75
16. 아키텍처 진화를 고려한 아키텍처 설계 방법	최유희 · 최희석 · 김제명	80
17. 임베디드 시스템 리엔지니어링에 관한 연구	김민경 · 윤희용	85
18. 정보보호 수준 개선을 위한 위협수준 예측모델	황산명 · 김상수 · 채현철	90
19. GeneSys : 임베디드 시스템에서의 틀을 이용한 컴포넌트 개발 방법론	김준길 · 이종인 · 양현곤	94
20. 임베디드 시스템을 위한 범용 인터페이스 모델	김준길 · 이종인 · 양현곤	100
21. 개발지표에 근거한 임베디드 소프트웨어의 정량적 프로젝트 관리 사례연구	박명오 · 이은석	105
22. A Web-based Networked Display User Interface for Smart Meeting Spaces	VinayRamachandra · ChanghyeokBae · JongWonKim	111
23. SOA 기반의 교육정보 검색 API 개발에 관한 연구	이재원 · 최은만	116
24. UML 표기법의 유용성 평가에 대한 연구: Systematic Review	권태희 · 임좌상	121

## ■ 인간과 컴퓨터 상호작용

1. 동적 임계값과 단순화된 칼만필터알고리즘을 이용한 효과적인 차선검출 .....	김세훈 · 김계영 · 최형일	132
2. 감정 상태에 따른 반응형 서비스 .....	김주석 · 고 건 · 이용찬 · 배유석	138
3. 인간 로봇 상호작용을 위한 Disparity 정보를 이용한 동작 인식 .....	김순기 · 김대진	142
4. 확률분포 통합 형태를 이용한 음성과 제스처의 멀티모달 융합 .....	이지근 · 한문성 · 김진태	147
5. 검색 결과를 보장하는 개체 유형별 자동완성 .....	정한민 · 이미경 · 상원경	152
6. VisualMessage:영상 인식을 통한 자동 Morse code 시스템 .....	박춘석 · 윤준영 · 이세인 · 한택돈	156
7. 양방향 데이터 방송 서비스에서 비디오 핫스팟 생성을 위한 반자동 객체 추출 방법 .....	박태진 · 임순범 · 최윤철	160
8. 음악 장르 구분을 위한 대표 샘플 추출 방법 .....	김현주 · 유만준 · 이인권	165
9. 사용자 정의에 의한 웹 위젯 구현 .....	박상우 · 박지호 · 한상건 · 공기석 · 이상호	169
10. 큐빅 스플라인 곡선을 이용한 Web3D 투어 가이드 설계 및 구현 .....	송득성 · 김학근	174
11. 사용자 경험에 기반한 웹 서비스 설계 .....	신승철 · 심보경 · 박종찬 · 배희정	178
12. 사용자 접근성과 도메인 확장성을 고려한 대화 인터페이스 기반 멀티 에이전트 시스템 .....	임성수 · 최봉환 · 조상배	182
13. 시간 정보를 이용한 사진 파일 관리 인터페이스 .....	김예진 · 최윤철	188
14. 손 인식을 이용한 컴퓨터 제어 .....	복창순 · 손연미 · 방영철 · 나보균	192
15. BIFS를 이용한 모바일 인터랙티브 콘텐츠 저작도구 제작 .....	이승훈 · 김상욱	196

## ■ 전산교육 시스템

1. 인문사회계열 대학생을 위한 수요지향적 IT 교육과정 연구 .....	정경화 · 강수용 · 조성호	200
2. m-Learning 환경에서의 학습자 정보 모형 연구 .....	이병일 · 장종성 · 손진곤	206

## ■ 컴퓨터 그래픽스

1. 풍경사진의 수채화 변환을 위한 비사실적렌더링 알고리즘 개발에 관한 연구 .....	조정찬 · 김종찬	212
2. 히스토그램 최적화 방법에 따른 이미지 스타일라이제이션 .....	김나리 · 윤종철 · 이인권	217
3. 동적 할당 노드를 가지고 있는 계층적 구조의 3차원 그래픽스 셰이더 .....	하상원 · 정우남 · 한택돈	221
4. 건축물 내에서의 군중 피난 시뮬레이션 시스템 개발 .....	조준성 · 박종승 · 이동호	225
5. 상호작용이 가능한 증강공간 편집도구 개발 .....	김동희 · 박종승 · 김정규	230
6. 점진적 해상도 기법을 이용한 정지영상 압축 방법 .....	김정율 · 위영철	236

# 자기 재구성 로봇 동향 연구

서진원<sup>○</sup> 김영철

홍익대학교 컴퓨터정보통신공학과

{suh.bob}@selab.hongik.ac.kr

## Survey of Self-Reconfigurable Robotics

Jin won Seo<sup>○</sup> R. Young Chul Kim

Dept. of CIC, Hongik university, Jochiwon ,Korea

### 요 약

본 논문에서는 미리 정의된 동적 S/W 아키텍처를 임베디드 시스템에 적용하기 위해 현존하는 자기 재구성 로봇에 대해 분석한다. 자기 재구성이란 외부 환경이나 이벤트에 의해 자기 스스로 형태를 동적으로 변형하여 새로운 환경이나 태스크에 맞게 처리 할 수 있도록 하는 아키텍처이다. 적용하는 방법으로는 외형적으로 변형을 일으키거나 무리나 때를 구성하여 태스크를 수행한다. 최종적으로 자기 재구성 로봇을 소프트웨어 아키텍처의 동적 재구성으로서 로봇의 특징인 모듈 구성 형태, 이동방법, 연결 메커니즘, 자원 관리 별로 분류, 분석하였다.

### 1. 서 론

현재 소프트웨어 아키텍처 관점으로 시스템을 설계하기 위해 기업에서나 연구 분야에서 많은 시도를 하고 있다. 기능적인 요구사항만을 고려하여 설계하였을 때 시스템이 완성도면에서 비기능적 요구사항의 만족도가 떨어지거나 이해관계자들과의 의사소통이 원활하게 되지 못하기 때문이다. 소프트웨어 아키텍처는 비기능적 요구사항을 접근하는 최선의 방법이다. 비기능적 요구사항은 서비스의 질과 시스템의 다른 관점을 상충하고 있고 또한 여러 도메인에 따라 다양함으로서 동적으로 재구성도 필요하다[1,5].

자기 재구성 로봇은 자기 자신의 형태를 변형하거나 연결성을 재구성하여 새로운 환경에 적응하는 로봇으로서 현재 많은 이슈가 되고 있는 분야이다. 가장 흥미로운 부분은 주어진 환경에서 새로운 환경으로 변화를 하였을 때 다양한 방법에 의해 동적으로 재구성하여 비기능적 요구사항인 성능이나 가용성 등을 만족할 수가 있다.

본 논문에서 자기 재구성 로봇의 동적 구조를 소프트웨어 아키텍처 관점으로 분석하여 연구 중인 동적 구조는 여러 가지 자기 재구성 로봇들을 모듈 구성 형태, 자유도, 이동방법, 연결 메커니즘, 자원 관리 등에 따라 분류를 할 수 있다. 동적 재구성이 요구하는 시스템이기에 소프트웨어 동적 구조를 적용한 로봇의 특징을 분석하고 향후 전망에 대해 예측한다. 논문의 구성은 2장에서는 소프트웨어 아키텍처와 자기재구성 로봇에 대한 관하여 언급하고 3장에서는 자기 재구성 로봇의 특성에 따른 로봇들을 분석 및 비교하였다. 마지막으로 4장에서는 결론 및 향후연구에 대해 서술하였다.

### 2. 관련 연구

#### 2.1 소프트웨어 아키텍처

소프트웨어 아키텍처란 개발하려고 하는 소프트웨어 시스템에 대한 핵심적인 기능적, 비기능적 요구 사항을 담고 있는 추상화 된 구조이다. 활용 방안에는 소프트웨어가 구현되기 전에 아키텍처가 제공하는 다양한 모형으로부터 여러 품질 특성을 추론하고 이를 바탕으로 품질을 향상시킬 수 있다[2,9].

소프트웨어 아키텍처를 접근하려고 하는 이유는 여러 가지가 있다. 그 중에 하나인 일반성으로서 특별한 도메인만 접근을 하려고 시도하지 않고 여러 도메인을 사용하기 때문이다. 두 번째로는 소프트웨어 아키텍처의 설계 단계를 추상화 단계로서 동적인 변화를 적절하게 추상화시키기 위해서이다. 세 번째로는 확장성으로서는 계층적 구조와 숨은 기술이 있기 때문이다. 넷째로는 여러 가지 ADL(Architecture Description Language: 아키텍처를 분명하게 표현하기 위해 정의한 언어)과 표기법, 정형적인 아키텍처 기반으로 추론과 분석으로 현존하는 시스템을 개발 할 수 있다. 마지막으로 많은 ADL과 아키텍처 접근으로 구성과 배치, 재구성이 가능하여 통합 접근이 가능하다.

본 논문에서 접근하는 초점으로서 소프트웨어 아키텍처의 추상화 설계 단계로서 적절한 동적인 변화를 하는 로봇에 대한 특징을 파악하는 것이다

#### 2.2 Modular Self-Reconfigurable Robotics

자기 재구성 모듈 로봇은 80년대 처음 나타나기 시작했다. 스탠포드 대학 Mark Yim에 의해 환경에 따라 이동을 완전 자동으로 하는 아이디어를 90년대 초반에 보급이 되었다. 그 시대에는 동종 로봇에 초점으로 단일 모듈 타입의 많은 복제품이 구성되었다[6].

자기 재구성 로봇은 많은 독립적인 모듈로 구성된 로봇이다. 각각 모듈은 액추에이터와 센서, 프로세서, 전원, 메모리, 모듈간의 연결/분리 통신방법이 있다. 이 모듈을 가지고 자기 자신 스스로 연결과 분리를 할 수 있고 새로운 환경에 적응하고 그들의 형태를 변형시킬 수 있는 로봇이다.

자기 재구성 로봇의 장점으로는 다양성과 견고함, 낮은 비용을 언급할 수가 있다. 다양성으로서는 기존 시스템보다는 환경이나 외부 요소에 더 적응을 할 수 있고 그 능력은 로봇 전체의 모듈들이 분해하거나 재조합하는 형태로서 새로운 태스크에 알맞게 구성한다. 견고함으로서는 로봇부분이 호환성이 있는 기계로서 하나의 모듈이 파손되면 다른 모듈로 대체할 수 있다. 낮은 비용으로서는 로봇하나의 전체 시스템을 구성하는 것보다는 각각의 모듈 타입으로서 경제적으로 부분을 생산하여 비용을 절감할 수 있다. 그러나 아직 이 세 가지 장점을 완벽하게 구현되지는 못하였다. 그리고 부가하자면 좀 더 다양한 자유도(Degree Of Freedom)를 만들도록 노력해야 한다. 그래서 결과적으로 다양한 태스크를 수행하기 위해서는 형태가 고정적인 로봇들과는 성능 면에서는 모듈 로봇이 가지고 있는 장점에 의해 분명한 차이가 있다[6,7].

그렇지만 자기 재구성 로봇의 장점만 있는 게 아니다. 모듈 로봇의 구성하기 어려움 면으로서 하드웨어 측면에서 바라보면 첫째, 하드웨어 의존도가 매우 높다. 하드웨어가 지원이 되지 않는다면 현재로서는 재구성하는 방법이 어렵다. 둘째, 자원관리의 문제점으로 각 모듈간의 자원을 공유로서 해결을 하려고 하지만 아직은 미숙하다. 셋째, 연결부의 결합과 분리가 쉬워야 하는데 아직 최적의 설계조건을 제안하지 못하였다. 넷째, 시스템의 기계적인 면과 전기적인 면의 내구력과 견고성이 떨어진다. 다섯째, 모듈 안에 있는 모터의 힘도 아직은 많이 부족하다. 그래서 이것을 극복하기 위해 모듈끼리의 협업을 통한 물체 이동을 연구 중이다[8].

### 3. Self-Reconfigurable Robotics 분류 및 분석

#### 3.1 자기 재구성 로봇의 분석

로봇 재구성 아키텍처로서의 분류는 크게 동종의 모듈로 구성되었거나 이종의 모듈로 구성된 로봇으로서 구분된다. 그러나 아직까지는 이종 모듈 시스템은 거의 개발이 되고 있지 않음과 동종 모듈 시스템으로 구성되어 있다. 동종 모듈 시스템에서 구성한 기하학적인 유닛 변화 분류와 재구성되거나 이동되어지는 방법에 대한 분류로 나누어진다. 또한 기하학적 유닛의 변화란 변형하는 형태의 모양, 즉 바둑판 모양과 비슷한 격자형이나 긴 사슬처럼 늘어진 체인형, 작고 움직이는 모빌형, 앞의 형태들의 장점이나 여러 가지를 2가지 이상을 통합한 하이브리드형으로 나누어 질 수 있고 재구성되거나 이동되어지는 방법에 대한 분류로서는 재구성하는 동안 타겟의 위치로 직접 조작용 하거나 움직이는 유닛에 의해 의지하는 재구성인 결정적 구조와 확률적 프로세스를 사용하여 유닛의 움직임을 의지하는 재구성인

확률적 구조로 분류가 되어 있다[10,11].

본 논문에서는 자기 재구성 로봇의 특징들 중 모듈 구성 형태, 모듈이 자기 재구성을 하여 이동하는 방법, 모듈들이 전자석이나 반영구자석, 전기, 수동, 등의 여러 가지 조합 방법인 연결 메커니즘, 보조전원 또는 외부전원을 사용하는 유무와 서로 모듈 간에 자원을 공유할 수 있는 자원 관리 등의 특성을 조사하였다.

#### 3.1.1 Metamorphic

격자형 구조로서 Chirikjian[Chirikjian, 1994;Pamecha and Chirikjian 1996]이 Meramorphic robotics 시스템을 증명하였다[3]. 기본적인 형태는 평면 6각형 모양으로서 다양한 기하학 구조를 가지고 있으며 서로 조합할 수 있는 3자유도 가지고 있다. 그림 1은 Metamorphic의 결합 형태별 사진이다. 주어진 구조 안에서의 구성 변화는 연결 각도의 변화에 의해 이루어지는 동적 자기 재구성의 능력도 가지고 있다.



그림 1. Metamorphic

#### 3.1.2 Fracta

Murata는 동종의 분산 시스템인 입체적 구조[Murata et al 1998]와 평면구조[Tomita et al 1999]를 고려했다[3]. 입체적으로 설계한 Fracta는 3개의 대칭축으로 12자유도를 가지고 있다. 그림 2는 Fracta의 실제 모습이다. 자기 재구성은 자동 연결 메커니즘과 판의 회전 방법에 의해 수행된다. 결점으로는 너무나 크고 무거운 것이다. 연결 메커니즘은 6개의 센서와 암호기를 이용한다. 그러나 입체적인 자기 재구성을 이룰 수 있는 시스템의 몇 개중 하나이다.



그림 2. Fracta

#### 3.1.3 Molecule

Molecule는 가장 기본적인 이종의 구조를 이루고 있는 시스템이다. 로봇의 형태는 그림 3과 같이 단단한 연결 묶음으로 되어 원자 연결이라고 부르고 두 개의 큐빅 모듈로 되어 있다. 각 각의 원자는 5개 Molecule 연결지점과 2개 자유도를 가지고 있다. 한 가지 자유도는 Molecule 연결묶음과 원자가 회전하는 것이 관련이 있고 다른 자유도는 내부 Molecule 커넥터 중 하나가

원자가 회전하는 것과 연관이 있다. 3가지의 형태로 나누어 대칭을 이루기 때문에 Molecule는 12가지 다른 변화를 만들어 질 수 있다.

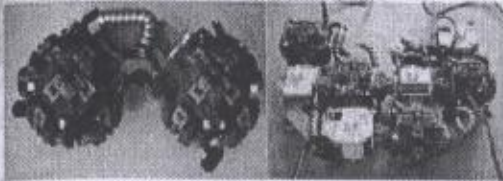


그림 3. Molecule

### 3.1.4 Telecube

Telecube는 그림 4와 같이 큐브 형태로서 각 면에 확장할 수 있는 팔이 있다. 팔은 몸길이의 반절만큼 독립적인 확장을 할 수 있다. 팔은 완전히 확장을 하거나 쏙 집어넣는 상태로 제한된 행위를 한다. 팔의 확장과 수축은 모듈의 행위를 형성하는데 제공된다. 각 팔의 끝 부분인 빗장은 각 각의 모듈을 연결하기 위해 두 개 모듈을 정렬하여 질 수 있다. 팔의 행위는 빗장을 걸고 풀고 하는 것으로서 모듈 로봇의 연결 구조를 의미한다. 이 행위는 임의의 재구성이라고 할 수 있기에 충분하다 [4].



그림 4. Telecube

### 3.1.5 I-Cube

I-Cube는 두 부분으로 나누어지는 시스템으로서 독립적인 제어 메커니즘의 모음이면서 소극적인 연결 요소가 있다. 그림 5와 같이 링크부분이 다른 큐브로부터 연결과 분리할 수 있는 능력이 있다. 모든 링크와 큐브는 전원과 정보는 붙어 있는 모듈에게 흘러 보내질 수 있다. 자기 재구성을 위한 행위는 링크에 의해 제공되고 큐브는 연산, 센싱, 자원을 제공하는 것을 사용할 수 있고 3자유도를 가지고 있다.



그림 5. I-Cube

### 3.1.6 Pneumatic

Pneumatic 시스템은 압축된 공기에 의해서 움직인다. 즉, 유동적인 바람을 사용하여 시스템을 이동시킨다. 압축된 공기를 Pneumatic 액추에이터로 눌렀을 때 로봇은 뒤틀린 관절 요소 주위로 회전을 한다. 관절의 묶임과 풀

어 주는 요소는 역시 압축된 공기에 의해 작동된다. 그림 6과 같이 두 개 모듈간의 결합을 수행 한다[12]

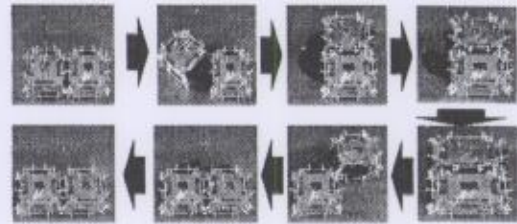


그림 6. Pneumatic

### 3.1.7 Atron

Atron은 격자형 시스템으로 그림 7와 같이 모듈은 회전축이 x,y,z축에 병렬로 놓여 있고 연결된 두 개의 모듈은 수직의 회전축을 가지고 있다. 모듈의 기본적인 행위는 적도 둘레로 90도 회전을 한다. 반구는 하나 또는 두 개의 다른 모듈이 단단하게 붙어 있고 다른 반구는 구조의 주요부분에 단단하게 붙어 있다. 자기재구성 시스템으로 이웃 모듈을 연결과 분리를 할 수가 있고 각각의 모듈끼리는 통신을 할 수가 있다[13].



그림 7. Atron

### 3.1.8 Miche

Miche는 MIT에서 Rus의 사람들에 의해 개발된 시스템으로서 그림 8과 같이 격자형 시스템으로서 임의대로 형태를 구성할 수 있다. 각 모듈은 자발적으로 연결할 수 있고 이웃 모듈과 통신을 할 수가 있다. 위치 교환할 수 있는 자석을 통해 연결 메커니즘이 제공된다. 모듈들은 이웃의 존재를 적외선 센서로 탐지하여 '면 대 면(face to face)' 통신을 한다. 조립된 구조가 있을 때 모듈은 컴퓨터 인터페이스와 분산 처리를 사용하여 가상으로 조각된 시스템을 형성한다. 모듈 모음의 그룹은 정보를 전송하고 저장하는 작은 알고리즘을 사용하여 최종 형태를 결정한다. 모듈의 구조는 중력에 의해 떨어 뜨려 형태를 구성한다[11].

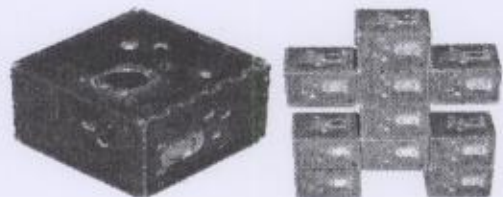


그림 8. Miche

### 3.1.9 CONRO

CONRO는 그림 9과 같이 동종의 모듈로 구성이 되었고 모듈 이웃끼리 연결과 분리가 가능하다. 모듈끼리 조

합하여 뱀처럼 이동가능하고 4축 보행도 가능하며 6축 보행도 가능하다. 다양한 센서가 장착되었고 카메라와 무선 랜을 장착하여 메인 시스템과 영상 전송이 가능하다. 모듈의 자유도는 없으면 시스템 자유도는 2DOF 이다.



그림 9. CONRO

3.1.10 Molecube

Molecube는 그림 10과 같이 하나의 모듈은 정육면체 모양 또는 큐브로 회전축이 큐브의 가장 긴 대각선에 자리 잡고 있다. 모듈 하나씩을 서로 각각 수작업으로 붙여서 자동차나 6축 로봇, 볼건을 잡는 로봇 등 여러 가지 모양들로 구성할 수가 있다. 오픈 소스기반으로 쉬운 모듈 확 장성과 낮은 제조비용의 이점이 있어 모듈러 로봇에 기여할 것이라고 생각된다. 모듈 자유도는 1DOF이고 시스템 자유도는 3DOF이다.



그림 10. Molecube

3.1.11 Swarm-bot

Swarm-bot은 다른 로봇으로부터 연결하거나 끊는 능력을 할 수 있는 자발적인 이동 로봇이다. 그림 11처럼 Track & wheel의 조합으로서 물리적으로 연결되어 무리를 지어가며 혼자서 물체를 이동시킬 수 없는 일들을 협업하여 목적 지역까지 이동시키는 로봇이다 [8,14].



그림 11. Swarm-bot

3.2 자기 재구성 로봇의 비교분석

자기 재구성 로봇의 특징 중 각 모듈 로봇 간에 자기 재구성을 하기 위한 연결 방법으로서 면 대 면, 점 대 점 등 여러 가지 방법들이 있는 연결 메커니즘과 모듈 로봇에서는 구성하기 위해서는 모듈들이 서로 같은 모양이나 성질을 가지고 있는 모듈을 동종과 서로 모양이나 성질이 다른 모듈을 이종, 변형할 수 있는 공간인 2차원 변형/3차원 변형, 모듈간의 통신 가능 여부 및 방법, 모듈 로봇의 자원 관리에 대해 표 1에 비교분석하였다.

표 1에서 현재까지 존재하는 자기 재구성 로봇은 대부분이 같은 형태나 성질의 모듈을 다수 구성하고 있는 동종시스템이고 Molecube, I-Cube의 로봇은 모양이나 성질이 다른 성질을 지닌 이종시스템의 모듈형태로 구성되어 있다. 그리고 변형 차수로는 초창기에 자기 재구성 로봇의 몇몇이 2차원 변형을 선택했으나 Molecube 이후부터 3차원적인 입체적 변형을 하였다. 연결 메커니즘으로서는 면 대 면, 점 대 점으로서 연결 부위에는 물리적인 고리나 집게이고 자석류는 전자석, 전기적, 반영구자석이고 그 외에 압축된 공기를 통하여 연결되어 있다. 각 모듈간의 통신으로서는 모듈끼리 연결시에 통신이 가능하거나 그렇지 못한 방식이 있고 또한 유/무선 방식으로도 나누었다. 그리고 마지막으로 자원 관리 부분에서는 모듈마다 소형전전지를 개별전원으로 가지고 있거나 보다 큰 전원을 필요시에는 외부전원을 사용하였다. 또한, 개별전원은 시스템 작동시간이 중요하기 때문에 자원공유 기능이 매우 중요하다. 모듈끼리는 자원을 공유하거나 그렇지 못한 로봇이 있다.

표 1. 존재하는 모듈 로봇의 특징 비교분석

시스템	동종/이종	차수	연결 메커니즘	모듈간의 통신	자원관리
Metamorphic	동종	2D	face to face(고리)	연결시 커뮤니케이션 가능	개별 전원
Fracta	동종	2D->3D	point to point(전자석)	연결시 커뮤니케이션 가능	외부 전원
Molecube	이종	3D	point to point(집게)	x	외부 전원
PolyBot	동종	3D	face to face(전기적)	연결시 커뮤니케이션 가능	개별 전원(연결시 공유)
TeleCube	동종	3D	face to face(반영구자석)	연결시 커뮤니케이션 가능	개별 전원(연결시 공유)
I-Cube	이종	3D	face to face(물리적)	연결시 커뮤니케이션 가능	개별 전원(연결시 공유)
Pneumatic	동종	3D	face to face(압축된 공기)	-	외부전원
Atron	동종	3D	point to point(물리적)	연결시 커뮤니케이션 가능	개별 전원(연결시 공유)
Swarm-bot	동종	3D	point to point(물리적)	모듈간 무선 통신가능	개별 전원
Superbot	동종	3D	face to face(수동)	연결시 커뮤니케이션 가능	개별 전원(연결시 공유)
Molecube	동종	3D	pin-and socket(전기적)	모듈간 무선 통신가능	개별 전원
Miche	동종	3D	face to face(반영구자석)	연결시 커뮤니케이션 가능	개별 전원(연결시 공유)

4. 결론

본 논문에서는 현존하는 자기 재구성 로봇에 대한 동향과 개발 기술, 연결 메커니즘을 분석하였다. 자기 재구성이란 외부 환경이나 이벤트에 의해 자기 스스로 형태를 동적으로 변형하여 새로운 환경이나 태스크에 맞게 처리할 수 있도록 하는 아키텍처이다. 자기 재구성 모듈 로봇의 분류를 하였고 각 형태와 연결 메커니즘, 이동 방법, 자원관리, 자유도 등 분석하여 언급하였다. 상황에 따라 모듈끼리 협업을 하여 불체를 목적지까지 이동을 하여 태스크를 처리하거나 모듈간의 자원을 공유한다. 아직 모듈 로봇의 제어 컨트롤러와 최적의 연결 메커니즘, 하드웨어의 견고함, 자원 유지, 모터의 힘 등 하드웨어적인 면이나 소프트웨어적인 면을 해결해야 할 문제들이 많이 있다.

향후 연구로서는 불확실한 환경이나 예기치 못한 이벤트에 적용하여 새로운 태스크를 수행할 수 있도록 작게는 자기 재구성 로봇, 크게는 임베디드 시스템에 소프트웨어 아키텍처 관점으로 최적인 동적 변형의 아키텍처를 미리 정의(pre-defined)하여 적용하려고 진행 중이다.

참고문헌

[1] Jesper Andersson, "Issues in Dynamic Software Architectures", 25th February 2000  
 [2] Mary Shaw, David Garlan "Software Architecture, Perspectives on an Emerging Discipline", Prentice Hall, pp.43-51  
 [3] Pakpong Jantapremjit David Austin, "design of a Modular Self-Reconfigurable Robot", Austalian Conference on Robotics and Automation Sydney, 14-15 Nov 2001.  
 [4] Jeremy Kubica, Eleanor Rieffel, "Creating a Smarter Membrane: Automatic Code Generation for Modular Self-Reconfigurable Robots", Proceedings of the 2002 IEEE International Conference on Robotics & Automation Washington, DC, May 2002

[5] Bernd Bruegge, Allen H.Dutoit "International edition Object-Oriented Software Engineering : Using UML, Patterns, and Java", pp.121-170  
 [6] Matthew D. Hancher, Gregory S. Hornby, "A Modular Robotic System with Applications to Space Exploration", 2nd IEEE International Conference on Space Mission Challenges for Information Technology , 2006.  
 [7] Kasper Støy, "Emergent Control of Self-Reconfigurable Robots", January 6, 2004  
 [8] <http://www.swarm-bots.org/>  
 [9] 서진원 "소프트웨어 아키텍처 기반의 임베디드 시스템 개발", 모바일학회 추계학술대회 pp.240-244 2006  
 [10] Zack Butler, Keith Kotay, Daniela Rus and Kohji Tomita, "Generic Decentralized Control for Lattice-Based Self-Reconfigurable Robots".  
 [11] Mark Yim, Wei-Min Shen, Behnam Salemi, Daniela, Mark Moll, Hod Lipson, Eric Klavins, Gregory s. Chirkjian, "Modular Self-Reconfigurable Robot Systems\_Challenges and Opportunities for the Future", IEEE Robotics & Automation Magazine, 2007.  
 [12] N. Inou, H. Kobayashi and M. Koseki, "Development of Pncumatic Cellular Robots Forming a Mechanical Structure", Seventh International Conference on Control, Automation, Robotics And Vision (ICARCV'02), Dec 2002, Singapore  
 [13] Morten Winkler Jørgensen, Esben Hallundbæk Østergaard, Henrik Hautop Lund, "Modular ATRON: Modules for a self-reconfigurable robot", September 28 - October 2, 2004, Sendai, Japan  
 [14] ELIO TUCI, MARCO DORIGO, "Cooperation Through Self-assembly in Multi-Robot Systems", ACM Transactions on Autonomous and Adaptive Systems, Vol. 1, No. 2, 115--150, December 2006.