

Jini 기반의 이동 로봇 제어 시스템에 관한 연구

이동일*, 한동창*, 김중무**, 박주현*
 *영남대학교 전기공학과, **한국전기연구소

Design of a Mobile Robot Control System based on Jini

Dong Il Lee*, Dong Chang Han*, Jong Moo Kim**, Ju Hyun Park*

*Dept. of Electrical Engineering, Yeungnam University, **Korea Electrical Research Institute(KERI)

Abstract - 본 논문은 유연하고 탄력적인 네트워크 구조를 제공하는 Jini를 기반으로 하는 이동 로봇 제어 시스템의 설계와 구현 방법을 제안한다. 기존의 tele-operation 시스템의 단방향 제어를 개선하여, 제안한 시스템에서는 양방향 제어가 가능하고 클라이언트 시스템과 로봇이 상호 연동할 수 있는 구조를 가진다. 기존의 tele-operation 시스템에 Jini 기술을 도입하여 네트워크상에서 자원 이동이 간편하고 분산 컴퓨팅이 가능한 시스템을 제안하고자 한다. 본 논문은 구체적으로 복수의 작업장 사이에서 자율 이동이 가능하고 새로운 작업장의 네트워크에 자동으로 적용할 수 있는 이동 로봇 시스템과 이러한 로봇 시스템과 연결될 수 네트워크 시스템을 제안한다.

로봇 제어 시스템에 Jini 기술을 도입하면 고가의 로봇 시스템을 여러 작업장에서 공유하여 사용할 수 있으므로 로봇의 활용도를 높이고 시설투자 비용을 줄일 수 있다. 그림 1은 Jini를 기반으로 이동 로봇을 공동으로 사용하고 있는 작업환경을 나타낸다.

1. 서 론

근래에 나타난 급격한 변화 중의 하나로 각 분야에 네트워크화를 꼽을 수 있다. 인터넷이 보급되면서 시작된 이러한 변화가 현재에는 가전제품을 네트워크로 연결하려는 시도로 이어지고 있다[1]. 로봇 분야에도 WWW을 이용한 이동로봇의 원격제어와 같은 연구가 진행되고 있다[2]. 네트워크를 이용하면 이중 장치 및 소프트웨어 간의 데이터 통신, 분산처리, 상호 작업 연동 등이 가능하다. 이것을 가능하게 하는 제어 미들웨어 기술로 HAVI[3], UPnP[4], Lonworks[5], Jini[6,7,8] 등이 있다.

Jini는 1998년에 썬 마이크로시스템즈사에서 발표한 분산 환경의 네트워크 자원공유를 위한 플랫폼이다. Jini를 이용하면 네트워크 상의 모든 종류의 장치와 소프트웨어 자원의 통합체를 구성하여 서비스와 자원을 공유할 수 있다. 그리고 사용자의 위치 변화에 관계없이 네트워크상의 자원에 대한 접근 및 네트워크의 개성이 용이하고 갱신 및 변경 작업이 단순해진다[9].

현재 공장 자동화를 위해 사용되고 있는 대부분의 산업용 로봇들은 고정된 환경에서 미리 설정된 작업들만 수행하도록 되어 있다. 산업용 로봇 이외에 인간의 상상으로 가능한 모든 종류의 로봇을 '서비스 로봇'이라 통칭한다. 서비스 로봇에 대한 연구는 로봇연구의 초기단계부터 이루어져 왔으나 경제성의 논리와 개발과정의 어려움 때문에 그리 활발하지 못하였다. 근래에 로봇 제조기술이 발전되고 각 분야에서 로봇의 필요성이 구체적으로 대두되면서 다양한 방면으로 로봇의 개발이 시도되고 있다. 예를 들면 방사능 지역과 같은 위험한 환경에서의 작업, 장애인들의 기능을 보조해 주는 기구와 같이 인간을 보조해 주는 작업, 우주공간이나 심해에서의 작업로봇과 같이 인간이 할 수 없는 일을 대신하는 기계로서의 작업 등이 있다. 특히 원자력 발전소의 작업로봇에 대한 연구가 활발하고, 일본과 같은 선진국에서는 이러한 로봇이 작업현장의 상당한 부분에서 일을 맡아서 하고 있다[10]. 그러나 국내의 경우 아직 연구가 초보단계이며 각 분야의 특수성과 막대한 개발비에 비해 활용도가 낮기 때문에 이러한 로봇의 개발에 부담이 많다. 이동 로

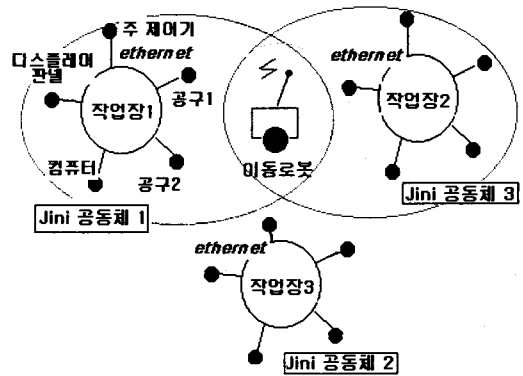


그림 1. 이동 로봇의 작업환경

2. Jini 기반의 이동 로봇 제어 시스템

2.1 Jini의 소개

Jini는 Java 기술을 기반으로 이기종 운영체제와 이기종 네트워크 환경에서 공유될 수 있는 하드웨어 또는 소프트웨어 자원을 연결해주기 위한 미들웨어 기술이다. 예를 들어 디지털 카메라가 네트워크에 연결되었다면, 네트워크에서 프린터를 찾아 카메라에 저장되어 있던 사진을 프린트할 수 있을 것이다. 그리고 사용자는 자유로이 디지털 카메라나 프린터를 다른 곳으로 옮길 수도 있다. 이와 같이 네트워크에서 다른 객체에서 어떤 유용한 기능을 제공하는 객체들을 '서비스(service)'라고 하며, 이 서비스가 모인 집합을 '연합(federation)' 혹은 '공동체(community)'라고 한다. 또 이들 다른 서비스를 이용하는 사용자나 소프트웨어를 '클라이언트(client)'라고 하는데, 서비스는 그 자체로서 다른 서비스를 사용하는 클라이언트일 수도 있다[12].

2.2 Jini 기반의 이동 로봇 제어 시스템 개발

그림 2는 Jini 공동체에서 네트워크 객체들의 소프트웨어 구조 및 계층도를 보여준다. 그림 1의 이동 로봇과 각 Jini 공동체의 네트워크 객체들은 그림 2의 A, B와 같은 시스템 구조를 가진다. 네트워크상의 서비스를 검색하기 위해서 Jini 공동체마다 Lookup Service가 하나 이상이 존재하여야 한다. 이동로봇은 이동이 가능하므로 그림 1의 작업장 1,2,3을 옮겨다니면 각 작업장의 Jini 공동체에 포함되기도 하고 사라지기도 한다. 이동

로봇은 세 공동체 중의 하나에 포함되어 고정되어 있는 작업장의 네트워크 객체와 서로 연동이 가능해진다. 이러한 잇점은 고가이지만 활용도가 적은 로봇 시스템을 반드시 설치하여야 하는 경우에 적합하다. 다음 절들에서 이러한 경우에 적용할 수 있는 Jini 기반의 이동 로봇 제어 시스템의 설계 내용에 대해 소개한다.

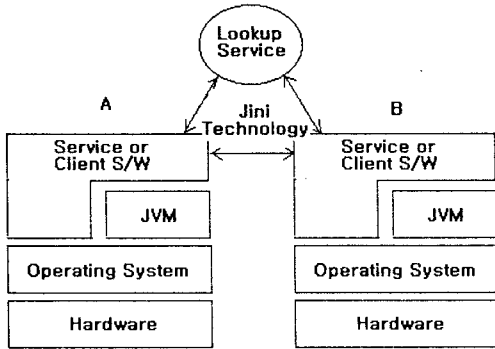


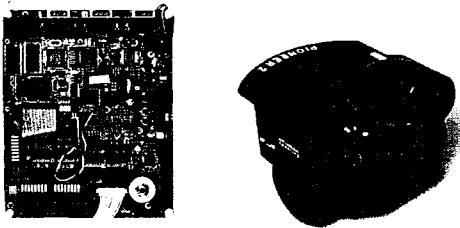
그림 2. 네트워크 객체의 소프트웨어 구조 및 계층도

2.2.1 로봇의 하드웨어 시스템

(1) 컨트롤러 구성요소

항목	수량	비고
CerfBoard	1	Intel StrongARM SA1110
		10Base-T Lan
		USB slave port
		32MB DRAM
		Compact flash socket
Break-out board	1	Flash memory(28F128J3A)
		JTAG port
		UCB 1200
		GPIO output port
		Power connector
LCD and Touch screen	1	Audio out
		Microphone in
		Three serial port
LCD and Touch screen	1	5.7" VGA LCD (320x240)

표 1. 컨트롤러 구성요소



(a) 컨트롤러 (b) 차체
그림 3. 로봇의 컨트롤러와 차체 외관

(2) 로봇차체 구성요소

- 2 DC Motors and Position Encoders
- Sonar Sensors : 전방향 14개, sensitivity

ranges from 10cm to more five meters

- Batteries and Power : 12V DC, 7A/hour
- Motion Controller : 20MHz Siemens 88C166
- Serial Port : RS232-compatible
- Radio Modem

2.2.2 로봇의 소프트웨어 시스템

(1) Infrastructure

- JVM(Java Virtual Machine) : 자바 가상 머신
- Java RMI : 컴포넌트 간의 기본적인 통신 방법
- Lookup Service : 공동체의 구성 요소들이 제공하는 서비스 발견 및 서비스 저장소의 발견

(2) Jini 프로그래밍

로봇이 제공할 수 있는 모든 기능들을 Jini 서비스 형태로 제공할 수 있다. 로봇 시스템에서 제공할 수 있는 서비스로는 센서를 통한 장애물 검색, 공구 조작을 통한 작업, 물체 이송 등이 될 수 있다. Jini 서비스 제공하기 위해서는 기존의 로봇 제어 프로그램들을 Jini 명세에 맞추어 수정해야한다. 서비스를 이용하는 클라이언트 프로그램은 단순히 공개된 인터페이스를 사용하여 프로그램하면 된다. Jini 프로그래밍 내용은 그림 3과 같다.

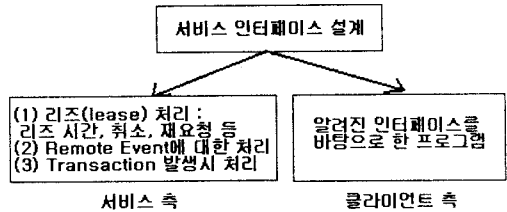


그림 4. Jini 프로그래밍 내용

그림 4에서 볼 수 있듯이 Jini 프로그래밍은 잘 알려진 인터페이스를 정의하고, 서비스 측에서는 이 인터페이스의 처리 내용을 구현하고 클라이언트 측에서는 알려진 인터페이스 코드를 프로그램에 추가하여 사용하는 것이다. Jini 프로그래밍으로 네트워크상의 모든 자원을 서비스화 할 수 있다. Jini 공동체는 이러한 서비스 기반의 네트워크이기 때문에 하드웨어의 기종이나 운영체제의 종류가 달라도 서로 연동이 가능하다.

3. 실험 및 결과

본 연구실에서는 제안된 Jini기반의 이동 로봇 제어 시스템을 실제 구현하기 위해서 다음과 같은 모의실험을 하였다. 그림 3의 (a)에 보여진 임베디드 컨트롤러와 PC들을 Jini 공동체로 연합하고, 임베디드 컨트롤러가 그림 5의 (b)에 나타난 것과 같은 문자를 클라이언트에서 출력하는 서비스를 제공하도록 하였다. 그림 5는 Jini 서비스를 테스트하기 위해 실행한 프로그램들이다. 실험을 위해 임베디드 컨트롤러와 두 대의 PC 1.2를 이용하였으며 다음과 같은 내용을 구현하였다.

- 1) 임베디드 컨트롤러 보드에 리눅스(Linux) 포팅 (Porting)
- 2) 포팅된 리눅스를 기반으로 하는 JVM(Java Virtual Machine) 포팅[13]
- 3) 문자를 출력하는 Jini 서비스 구현
- 4) PC 1에서 Lookup Service 실행
- 5) PC 2를 클라이언트로 하고 Jini 서비스를 이용하는 프로그램 작성

```

discovered a lookup service?
Set serviceID to ff1ca61d-a16e-457e-8935-a4fb8d5f129
Renewing lease.
Renewing lease.
Renewing lease.

```

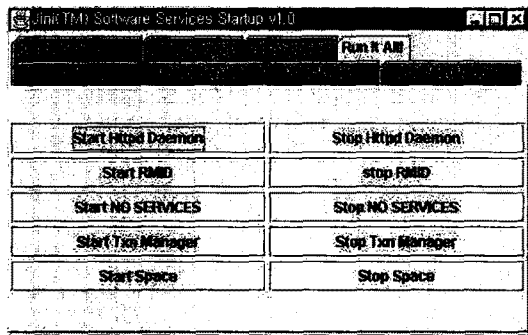
(a) 서비스 측 실행화면

```

Got matching service.
Its message is : Hello, Jini!
Renewing lease.
Renewing lease.
Renewing lease.

```

(b) 클라이언트 측 실행화면



(c) 록업 서비스 프로그램

그림 5. Jini 서비스 테스트 프로그램

4. 결 론

본 논문을 통하여 이동 로봇 제어 시스템에 Jini 기술을 도입하여 고가의 로봇 시스템을 여러 작업장에 공유할 수 있는 방법을 제안하였다. 제안된 방법은 자원의 공유를 통해서 그 이용도를 증가시키므로 생산라인에서 시설투자비를 절감할 수 있는 효과를 낼 수 있다. 결론적으로 앞으로는 Jini와 같은 분산 네트워킹 기술이 널리 보급되어 자원 분배와 결합을 통한 다양한 시스템의 개발이 가능함을 보았다.

향후 본 연구실에서 현재 개발중인 실제 이동로봇에 본 논문에서 제안한 Jini 기반의 이동 로봇 제어 시스템을 적용할 예정이다.

[참 고 문 헌]

- [1] 인터넷 정보가전 산업 협의회, <http://iipc.kait.or.kr/>.
- [2] 박창욱 외 4인, "WWW을 이용한 이동로봇의 원격제어", 대한전기학회 하계 학술대회 논문집, D권, pp2738-2740, 2000.
- [3] <http://www.havi.org/>.
- [4] <http://www.upnp.org/>.
- [5] <http://www.echelon.com/>.
- [6] Ken Arnold, "The Jini Specification", Addison-Wesley, 1999.
- [7] Scott Oaks, Henry Wong, "Jini in a Nutshell", O'Reilly, 2000.
- [8] W. Keith Edwards, "Core Jini", Prentice-Hall,

1999.

[9] 인터넷 정보가전 산업 협의회, "인터넷 정보가전 기술개발 기획연구", pp7, 2000.

[10] 이법희, 구관모, 우희곤, "원전 무인감시용 원격조종 이동로봇 시스템 개발사례", 제어·자동화·시스템공학 학회지, 제2권 제3호, pp71-80, 1996.

[11] Michael Deleo, "A Demonstration of Jini Technology and the K Virtual Machine", <http://java.sun.com>, 1999.

[12] 김도형, "동적이며 견고한 네트워크를 위한 지니", 월간 마이크로 소프트웨어, 11월호, pp220-228, 1999.

[13] <http://www.kaffe.org/>.