

소프트웨어 로봇 연구용 테스트베드

김인철(경기대학교)

요 약

소프트웨어 로봇들은 시뮬레이트된 센서와 실행기들을 가지고 가상 환경 안에서 자율적으로 동작하는 지능형 가상 에이전트로 볼 수 있다. 지능형 가상 에이전트 연구용 테스트베드들은 에이전트들의 성능 평가를 위해 표준화된 작업들을 구현할 수 있고, 환경이나 실험 에이전트들을 제어할 수 있는 도구들이다. 본 논문에서는 RoboCup 축구 시뮬레이션시스템, Gamebots 시스템, USARSim 시스템 등과 같은 지능형 가상 에이전트와 멀티 에이전트시스템 연구를 위한 대표적인 테스트베드들을 간략히 소개한다.

1. 서 론

소프트웨어 로봇(software robot) 즉, 소프트봇(softbot)은 소프트웨어 에이전트(software agent)를 지칭하는 포괄적인 표현이다. 물리로봇(physical robot)들이 실제 센서(sensor)들과 실행기(effector)들로 구성된 몸체를 가지고 물리법칙이 적용되는 실세계(real world)에서 동작하는데 반해, 이들 소프트웨어 로봇, 소프트웨어

에이전트들은 시뮬레이트된 센서와 실행기들을 가지고 가상 환경(virtual environment)에서 동작한다. 본 논문에서는 특히 이러한 소프트웨어 에이전트들 중에서도 인터넷 웹 로봇과 같은 정보 에이전트(information agent)들과는 달리, 가시화된 그래픽 몸체를 가지고 가상 세계에서 자율적으로 동작하는 지능형 가상 에이전트(intelligent virtual agent)들에 대해 관심을 가지며, 이들을 위한 대표적인 연구용 테스트베드(test-bed)들을 소개하고자 한다.

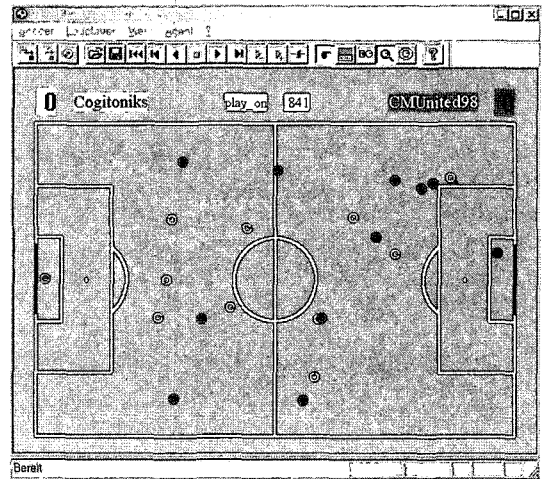
Hanks^[13] 등은 에이전트의 인지구조 연구와 관련하여 벤치마크(benchmark)와 테스트베드의 기능을 설명하였다. 그들은 벤치마크를 실제 응용영역에서 자주 등장하는 문제들을 대표할 수 있는 하나의 표준 작업(standard task)으로 정의하였다. 벤치마크의 잇점은 동일한 작업에 서로 다른 인지구조들을 적용하고 그 결과들을 측정함으로써 성능에 대한 비교 분석이 가능하다는 것이다. 이에 반해 테스트베드들은 이러한 표준 작업들이 구현될 수 있는 환경들로 정의하였다. 일반적으로 테스트베드들은 표준 작업들을 구현할 수 있는 기능 이외에도 데이터 수집을 위한 수단, 환경 인자(environmental parameter)들을

제어할 수 있는 기능, 시나리오 생성 기능 등을 부가적으로 제공한다. 또 테스트베드를 통해 사용자는 실험 에이전트들에 대한 세밀한 제어를 줄 수 있을 뿐 아니라 성능 평가기준들을 제시할 수도 있다. 본 논문에서는 현재 지능형 가상 에이전트와 멀티 에이전트시스템 연구 분야에 널리 이용되는 대표적인 테스트베드들인 RoboCup 축구시뮬레이션시스템, Gamebots 시스템, USARSim 시스템들의 구성과 특징들을 요약하여 소개한다.

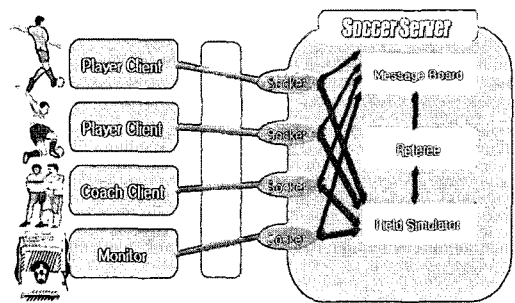
II. RoboCup 시뮬레이션시스템

RoboCup[4] 축구 시뮬레이션시스템은 매년 개최되는 국제 로보컵 축구 대회와 공식 경쟁부문을 위한 환경으로서, 또 지능형 멀티 에이전트 시스템 연구를 위한 테스트베드로서 대학과 연구소를 중심으로 폭넓은 사용자층을 확보하고 있으며 계속 진화 발전하고 있다. RoboCup 축구 시뮬레이션시스템은 그림 1과 같이 휴먼 또는 물리로봇들이 실제 경기장에서 팀을 이루어 벌이는 축구경기와 흡사하게 원격의 소프트웨어 에이전트들이 제어하는 가상 선수들이 하나의 가상 경기장에서 축구시합을 펼칠 수 있도록 해주는 시뮬레이션시스템이다.

그림 2는 네트워크 기반의 클라이언트-서버 분산시스템인 RoboCup 축구 시뮬레이션시스템의 개념적 구조를 나타내고 있다. RoboCup 축구 시뮬레이션시스템에서 각 클라이언트 프로그램은 UDP 소켓을 통해 서버에 연결되어 한 명의 가상 선수나 코치를 제어하는 자율 소프트웨어 에이전트이다. 서버 프로그램은 공과 선수를 포함한 경기장안의 모든 사물들의 움직임을 시뮬레이션하며, 그림 3과 같이 가상의 센서를



〈그림 1〉 RoboCup 축구 시뮬레이션 모니터 화면

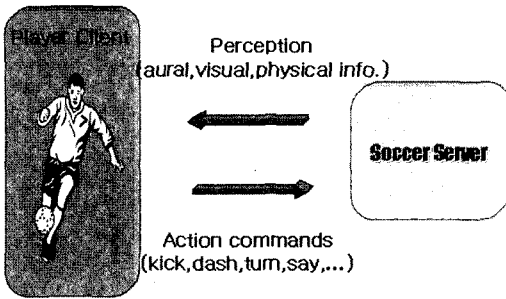


〈그림 2〉 RoboCup 축구 시뮬레이션시스템의 구조

통해 들어오는 센서정보(sensory information)를 각 클라이언트 에이전트에게 전달하고, 역으로 해당 에이전트가 결정한 행동 명령(action command)을 받아 시뮬레이션을 통해 가상 환경 위에 실행해주는 역할을 한다.

RoboCup 축구 시뮬레이션시스템에서는 클라이언트 프로그램으로 구현되는 각 팀의 선수나 코치 에이전트들 외에도 경기규칙에 따라 경기를 진행해나가는 별도의 가상 심판(referee)을 제공한다.

RoboCup 축구 서버는 약 0.1초 간격의 일정



〈그림 3〉 각 클라이언트와 서버간의 메시지 교환

한 시뮬레이션 주기(cycle)에 따라 선수 에이전트들의 행동 명령을 수집하여 실행하고 월드 상태를 변경하는 일을 계속한다. 서버가 각 에이전트에게 전달하는 센서정보들은 크게 그림 3과 같이 음성정보(aural information), 시각정보(visual information), 그리고 자신의 신체정보(physical information) 등으로 구성된다. 음성정보는(hear Time Direction Message) 형태의 메시지로, 시각정보는(see Time ObjInfo ObjInfo...) 형태의 메시지로, 신체정보는(sense_body Stamina, Effort, Recovery, Speed, View_Quality, View_Width) 형태의 메시지로 각각 전달된다. 특히 시각정보 메시지는 현재 에이전트의 시야에 들어오는 골대, 공, 선수, 경기장 경계표시 등 경기장안의 사물 각각에 대해(ObjName Distance Direction DistChng DirChng FaceDir) 형태의 상세 정보를 담고 있다. 반면에, 각 에이전트가 서버에 보내는 행동 명령들로는 Say(Message), Turn(Angle), Dash(Power), Kick(Power,Angle), Catch(Angle), Change_View(View_Quality, View_Width) 등을 허용하고 있다.

이와 같은 RoboCup 축구 시뮬레이션시스템은 협력해야하는 다수의 팀 동료(teammate)들

과 경쟁해야 하는 다수의 상대팀 선수(adversary)들이 함께 공존하는 분산 멀티 에이전트 환경을 제공한다. 그리고 이 환경에서 각 에이전트는 실제계와 같이 잡음(noise)이 존재하는 불완전한 센서들과 실행기를 가지고 있으며, 또 에이전트들은 공유 채널을 통한 매우 제한적인 통신만 가능하다. 그리고 빠른 변화가 계속되는 축구 게임의 특성상 각 에이전트들은 매순간 실시간 제약(real-time constraint)을 만족할 수 있도록 빠른 의사결정을 하여야 한다. 현재 RoboCup 축구 시뮬레이션시스템은 에이전트 제어구조(control architecture), 상대 모델링(opponent modeling), 멀티 에이전트 계획과 조정(multi-agent planning and coordination), 멀티 에이전트 학습(multi-agent learning) 등 다양한 지능형 에이전트 연구 분야에 효과적으로 이용되고 있다.

III. Gamebots 시스템

Gamebots^[11] 시스템은 지능형 에이전트 및 다중 에이전트 연구를 목적으로 USC의 ISI연구소와 CMU에 의해 공동으로 개발된 Unreal Tournament^[3] 게임엔진 기반의 지능형 에이전트 연구용 테스트베드이다. 그동안 에이전트 연구에 효과적으로 이용되어온 2차원 가상 시뮬레이션 환경인 RoboCup 축구 서버에 이어, 보다 복잡하고 현실적인 물리법칙이 적용되는 3차원 공간과 뛰어난 그래픽, 그리고 다양한 작업 목표와 문제들이 존재하는 새로운 연구 환경으로서 온라인 3차원 전략게임과 액션게임들을 고려하게 되었고, 이 중에서 상용 게임인 UT게임이 이러한 목적으로 선정되어 연구용으로 확장되었다. 그림 4는 대표적인 3차원 일인칭 전투게임인



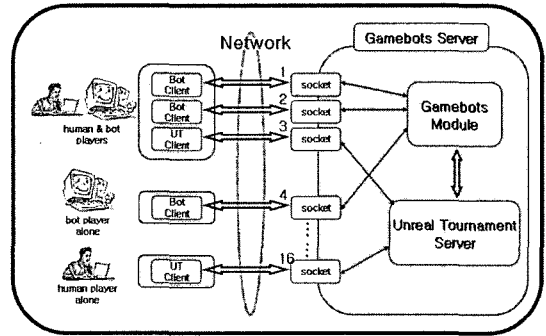
〈그림 4〉 Gamebots시스템의 게임 장면

UT게임의 한 장면을 보여준다.

Gamebots시스템의 서버는 그림 5와 같이 네트워크 소켓을 통해 하나의 NPC(Non-Player Character)를 제어하는 클라이언트 프로그램 - Gamebots 시스템에서는 보트 클라이언트(bot client)라고 부름- 들에게 휴먼 플레이어와 동일한 센서정보를 제공할 뿐만 아니라, 역으로 원격의 클라이언트 프로그램으로부터 행동 명령(action command)을 받아 수행해주는 역할을 한다.

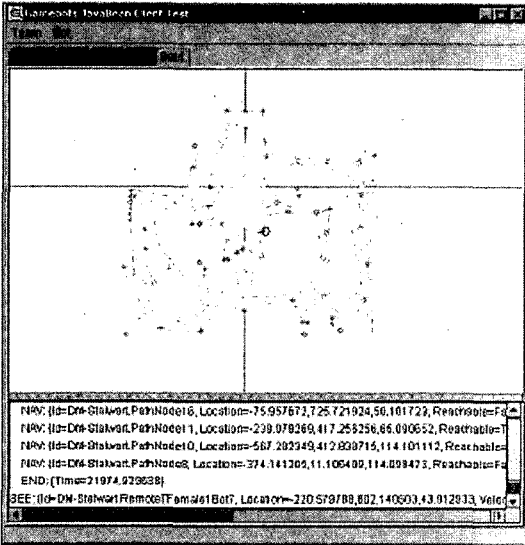
Gamebots 서버가 원격의 각 에이전트에게 전달하는 메시지들은 크게 초기화 메시지(initialization message), 동기 메시지(synchronous message), 비동기 메시지(asynchronous message)로 구성된다.

연결 초기 단계에는 초기화 메시지의 교환을 통해 서버 측에서는 보트 클라이언트에게 게임 타입과 게임의 승리조건 등을 알려주며, 보트 클라이언트는 자신이 제어하는 캐릭터의 이름, 소속 팀 등을 서버 측에게 알려준다. 일단 연결이 되고 나면, 서버는 동기 메시지 모드와 비 동기 메시지 모드를 옮겨 다니며 캐릭터들의 센서정



〈그림 5〉 Gamebots시스템의 구조

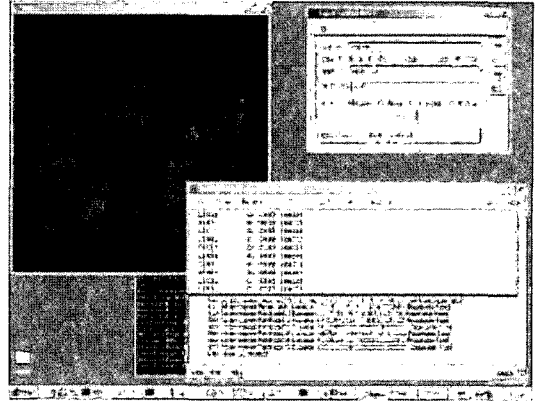
보(sensory information)를 해당 보트 클라이언트들에게 보내준다. 각 메시지는 메시지 유형(message type)과 다수의 속성-값(attribute-value) 쌍들로 구성된다. 동기 메시지는 약 0.1초 간격으로 계속해서 해당 캐릭터의 감각기관(눈, 귀)을 통해 들어오는 시각정보와 자신의 위치 및 신체정보, 그리고 게임 상황정보(팀별/개인별 득점)를 묶어 보트 클라이언트에게 전달하는 것이다. 동기 메시지는 하나의 BEG 메시지와 하나의 END 메시지를 경계로 삼아 상기의 정보들을 담은 하나의 메시지 블록을 만들어 보내진다. 이에 반해 비동기 메시지는 해당 캐릭터가 게임 중 어떤 특별한 사건에 직면했을 때만 보내지는 것으로서, 벽에 부딪히거나 아이템을 줍거나 무기를 바꾸거나 늪지대와 같은 새로운 지역으로 진입했을 때 관련 정보를 전달해준다. 한편, 원격의 보트 클라이언트로부터 Gamebots 서버에 전달되는 명령 메시지 유형에는 RUNTO, TURNT0, JUMP, SHOOT, STOP, CHANGWEAPON 그리고 에이전트 간의 대화를 위한 MESSAGE 등이 있다. 특히 원격의 보트 클라이언트들은 MESSAGE 명령에서 팀 전용 채널 또는 글로벌 채널을 지정함으로써 팀 동료들끼리만 메시지를 교환하거나 가상환경내



〈그림 6〉 Bot VizClient 도구

의 모든 캐릭터들과의 메시지 교환이 가능하다.

Gamebots 시스템은 가상 에이전트의 행위를 관찰하고 분석하기 위한 가시화 및 분석 도구를 제공한다. 먼저 Gamebots 시스템은 세 가지 종류의 가시화 도구를 제공한다. 첫 번째는 3차원 그래픽 환경을 제공하는 UT 게임 자체의 클라이언트 프로그램이다. 각 에이전트의 행위를 진단하고 실험하기 위해서는 해당 에이전트와 동일한 가상공간 안에서 함께 행동하는 것이 효과적이는데, UT 게임 클라이언트는 에이전트들의 행동을 3차원 환경에서 관찰할 수 있는 관찰자 모드(spectator mode)를 제공함으로써 그림 4와 같은 UT 게임 클라이언트 프로그램 자체가 가장 좋은 가시화 도구가 되는 것이다. 두 번째 가시화 도구는 Global VizClient 도구이다. 이 도구는 개별 에이전트의 행위가 아니라 가상 환경 안의 모든 에이전트들의 활동을 보여준다. 또한 팀이나 개개의 모든 객체의 점수, 위치를 보여준다. 주로 팀 행위나 전략 등을 분석할 때 사용하



〈그림 7〉 데이터 수집 및 분석 도구

는 도구이다. 세 번째 가시화 도구는 그림 6의 Bot VizClient 도구이다. Bot VizClient 도구는 개별 에이전트의 행위를 분석할 때 사용하는 도구이다. 이것은 개별 에이전트의 행위를 2차원 평면에 실시간으로 그려주고 텍스트 형태의 메시지로 각 에이전트의 경로를 기록해준다.

또한 Gamebots 시스템에서는 에이전트의 행위를 분석하고 평가하기 위한 목적으로, 그림 7과 같은 데이터 수집도구를 제공한다. 데이터 수집도구는 실행 중인 에이전트의 인지 정보와 행동 정보 등을 기록하였다가 재현해주며, 평균속도나 정확도 등 개별 에이전트나 각 팀에 대한 통계치를 구해주고, 그것을 통해 개별 에이전트나 각 팀의 행위를 분석, 평가할 수 있도록 해준다.

Gamebots 시스템이 제공하는 가상 환경 안에서 에이전트들이 자율적으로 다른 에이전트나 휴먼 플레이어들을 상대로 DeathMatch, Domination, CaptureTheFlag 같은 게임들을 성공적으로 진행해나가기 위해서는 매우 복잡하고 다양한 지능적 행위와 능력을 보여줄 수 있어야 한다. 예컨대 효과적인 탐색(exploring)을 통해 제한적인 시각정보로부터 자신이 놓여 있는

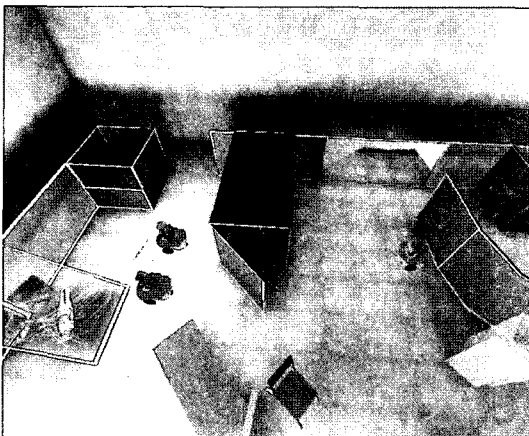
미지의 3차원 공간에 대한 지도를 작성(map making)하는 능력, 이를 바탕으로 원하는 목적지까지 효율적인 이동경로를 계획(path planning)하는 능력, 팀 구성원으로서 자신의 적절한 역할을 결정(role assignment)하는 능력, 전략적 계획수립(strategic planning) 능력 등이 요구된다. 따라서 많은 연구기관들에서 이와 관련된 지능형 에이전트 연구를 진행하고 있다. 그뿐만 아니라 군사용 가상현실시스템(virtual reality system)을 구현하거나, 쌍방향 드라마(interactive drama)를 제작하는 등 다양한 응용 분야에도 Gamebots시스템이 활발히 적용되고 있다.

IV. USARSim 시스템

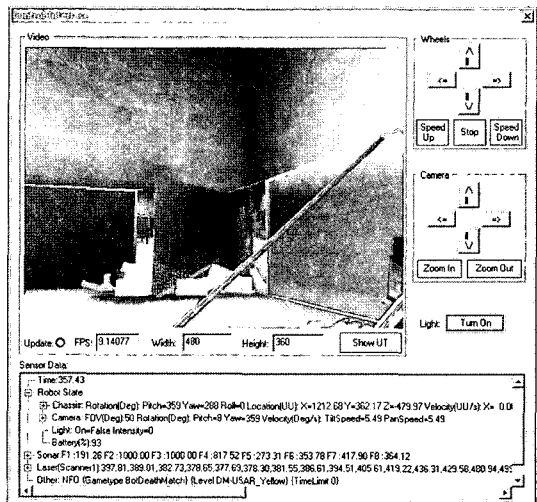
USARSim^[7, 12] 시스템은 긴급 상황에서의 휴먼-로봇 상호작용(Human-Robot Interaction, HRI)과 멀티로봇 조정(multirobot coordination) 연구를 위한 목적으로 개발된 도구로서, 도시 재난 발생시 탐색과 구조 활동(Urban Search and Rescue, USAR)을 펴는 로봇들을 시뮬레이션하

는 시스템이다. USARSim은 국가표준연구소(NIST)에 마련되어 있는 자율 이동 로봇용 USAR 표준 테스트 환경을 시뮬레이션을 통해 가상 환경으로 재현한다. NIST의 USAR 표준 테스트 환경은 Yellow, Orange, Red 등 세 가지 서로 다른 환경과 시나리오들로 구성되어 있는데, 그림 8은 그 중에서 Yellow를 USARSim시스템을 이용해 재현한 것이다. 이 화면은 도시재난이 발생한 한 가상 환경에서 다수의 로봇들이 희생자들을 찾기 위해 탐색하는 모습을 보여주고 있다.

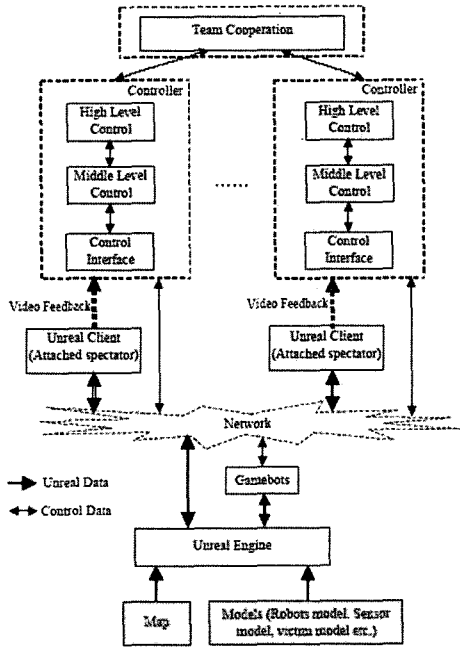
그림 9는 USARSim시스템의 한 클라이언트 프로그램으로서, 가상환경에서 작업 중인 로봇에 대한 그래픽 기반의 사용자 인터페이스를 보여주고 있다. 사용자는 이러한 사용자 인터페이스를 통해 카메라 영상 등 해당 로봇의 실시간 센서정보를 볼 수 있고, 우측의 동작버튼들을 이용하여 전후좌우 로봇의 이동 행위와 카메라 움직임을 직접 제어할 수 있다. USARSim 시스템의 서버는 앞서 소개한 상업용 3차원 게임엔진



〈그림 8〉 다수의 로봇들이 작업 중인 USARSim 시스템의 화면



〈그림 9〉 원격 로봇제어를 위한 사용자 인터페이스

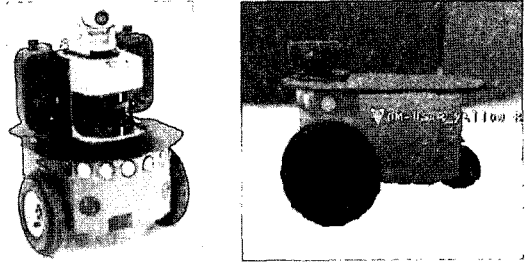


(그림 10) USARSim 시스템의 구조

인 Unreal Tournament와 Gamebots 서버 모듈을 이용하여 구현되었다. 그림 10은 USARSim 시스템의 전체 구조를 보여주고 있다.

USARSim 시스템은 현재 Pioneer사의 P2AT 로봇과 P2DX 로봇, CMU대학의 PER 로봇과 Corky 로봇, 그리고 전형적인 4바퀴 자동차 등 총 다섯 가지 종류의 로봇들에 대한 3차원 모델을 제공하고 있다. 그림 11은 좌측에 실제 P2DX 로봇과 USARSim 시스템에서 시뮬레이트된 P2DX 로봇의 모습을 보여주고 있다.

USARSim 시스템은 각 로봇에 장착된 레이저 및 초음파 센서들에 대한 시뮬레이션 모델을 제공하고 있다. 따라서 서버로부터 이들 센서에 감지되는 가상 데이터들이 일정한 형식의 메시지로 각 클라이언트 로봇 제어 프로그램(controller)에 전달된다. 또한 Gamebots 시스템의 경우와 같이, 클라이언트 로봇 제어 프로그램



(그림 11) 실제 P2DX 로봇과 시뮬레이트된 P2DX 로봇

이 생성한 행동 명령 메시지들이 서버에 전달되어 그 실행효과를 시뮬레이션하게 된다. 이 밖에도 로봇 제어 프로그램인 각 에이전트의 개발을 지원하기 위해 Python 프로그래밍 언어로 개발된 기존의 Pyro 패키지를 이용할 수 있는 기능을 제공하고 있다. 앞서 소개한 테스트베드들에 비해 최근에 개발된 USARSim 시스템은 올해부터 RoboCup 국제대회의 Robot Rescue 경쟁부문의 공식 지원시스템으로 채택되어 앞으로 이것을 활용한 연구가 활발히 진행될 것으로 기대된다.

V. 결론

본 논문에서는 소프트웨어 로봇 중 특히 지능형 가상 에이전트에 대한 연구용 테스트베드들을 소개하였다. 이러한 테스트베드들을 연구에 적극적으로 이용함으로써 실험 환경을 개발하는데 드는 노력을 줄이고 연구 결과들을 좀 더 객관적으로 평가할 수 있을 뿐 아니라 에이전트와 환경과의 상호작용에 대한 시각화를 통해 인공지능 및 지능형 에이전트 분야를 처음 접하는 연구자와 학생들에게 흥미와 이해를 증진시키는 데도 큰 도움을 줄 수 있을 것으로 본다.

참고문헌

- [1] J. Gerstmann, *Unreal Tournament : Action Game of the Year*, GameSpot, 1999.
- [2] H. kitano, M. Asada, Y. Kuniyoshi, I. Noda, E. Osawa, "RboCup: The Robot World Cup Initiative," in Proceedings of the First International Conference on Autonomous Agents, Marina del rey, CA, pp.340-347, 1997.
- [3] J.E. Laird, Michel van Lent, "Human-level AI's Killer Application : Interactive Computer Games", Proceedings of AAAI-2000, August 2000.
- [4] J. Wang, M. Lewis, S. Hughes, M. Koes, and S. Carpin, "Validating USARsim for use in HRI Research," in Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting, pp. 457-461, 2005.
- [5] L. Gasser, "MAS Infrastructure Definitions, Needs, and Prospects," in Proceedings of the Workshop on Scalable MAS Infrastructure, Barcelona, Spain, 2000.
- [6] R.B. Calder, J.E. et al., "ModSAF Behavior Simulation and Control," in Proceedings of the Second Conference on Computer Generated Forces and Behavioral Representation, STRICOM-DMSO, 1993.
- [7] R. Adobbati, et al, "Gamebots: A 3D Virtual World Test-Bed For Multiagent Research," in Proceedings of Agents-01, May, 2001.
- [8] S. Carpin, J. Wang, M. Lewis, A. Birk, and A. Jacoff, "High Fidelity Tools for Rescue Robotics: Results and Perspectives," in Proceedings of Robocup 2005 Symposium, 2005.
- [9] S. Hanks, M.E., Pollack, and P. Cohen, "Benchmarks, Test Beds, Controlled Experimentation, and the Design of Agent Architectures," *AI Magazine*, vol.14, pp.17-42, 1993.

저자소개



김인철

1985년 서울대학교 수학과 학사
 1987년 서울대학교 대학원 계산통계학과 석사
 1995년 서울대학교 대학원 계산통계학과 박사
 1989년-1996년 경남대학교 전산통계학과 전임강사, 조교수
 1996년-현재 경기대학교 정보과학부 전자계산학 전공 부교수, 교수
 2003년-2004년 미국 Michigan State University 방문교수
 주관심분야 인공지능, 지능로봇, 에이전트, 계획시스템, 기계학습