

## 산업용 로봇의 3차원 오프라인 시뮬레이터 개발

김홍래\*(경남대 대학원), 신행봉(경남대 대학원), 한성현(경남대 기계자동화공학부)

### Development of 3D Off-line Simulator for Industrial Robots

H. R. Kim(KNU), H. B. Shin(KNU), S. H. Han(Division of Mechanical and Automation Eng., KNU)

#### ABSTRACT

We propose a unmaned integrating control system based-on Windows XP version Off-Line Programming System which can simulate a Robot model in 3D Graphics space in this paper. The robot with 4 and 6 axes modeled SM5 and AM1 respectively were adopted as an objective model. Forward kinematics, inverse kinematics and robot dynamics modeling were included in the developed off-line program. The interface between users and the off-line programming system in the Windows XP's graphic user interface environment was also studied. The developing language is Microsoft Visual C++. Graphic libraries, OpenGL, by silicon Graphics, Inc. were utilized for 3D Graphics.

**Key Words** : Unmaned integrating control system(무인통합제어시스템), Industrial robot(산업용 로봇), Off-line programming system(오프라인 프로그래밍 시스템), Auto-teaching(자동교시), Visual C++(비주얼 C++), 3D Graphics(3D 그래픽스)

#### 1. 서론

오늘날 산업계는 생활향상에 따른 노동력의 고임금화와 소비시장의 다품종 소량생산 요구가 서로 상반되어서 상당한 진통을 겪고 있으며, 작업의 빈번한 변화에도 적응할 수 있는 숙련된 기능공은 이전의 국내 사정과 달리 상당한 임금을 요구하고 있다.

이로 인하여 근간 각 기업을 중심으로 생산성 향상을 위한 로봇의 도입을 통한 공장 자동화가 산업 전반에 걸쳐 진행되어 오고있다. 또한, 복잡한 작업의 경우에는 교시에 많은 시간이 소요되고, 시스템의 일부분의 변경에 따른 성능 평가과정에서도 많은 시간과 비용이 소요되는 문제점이 있다.

대부분의 로봇에 의한 자동화 교시작업이나, 변경된 시스템의 성능실험은 온라인 방식으로 수행되는 실정이며, 현재 로봇을 통한 공장 자동화는 빈번한 작업환경 변화에 경쟁력 있게 대처하기보다는 오히려 시간과 인력을 더욱 소모하는 문제점으로 대두되고 있다.

앞서 거론된 바 있는 온라인 방식으로 교시, 성능 실험을 하기 위해서는 별도의 개발라인이 요구되며, 작업경험이 풍부한 전문가에 의하여 교시, 성능 실험이 이루어져야 한다.

프로그래밍을 위해서는 전체 시스템이 중단되어야 하며, 로봇의 오동작에 의한 안전사고의 위험성도 상당부분 내재되어 있다.

이러한 문제점은 실제 작업과 유사한 환경을 가진 시뮬레이터인 Off-Line Programming System(OLP 시스템)을 이용함으로써 해결될 수 있다. OLP 시스템을 이용하면 로봇의 가동 중에도 동적 시뮬레이션이 가능하며, 작업교시, 궤적계획, 제어 알고리즘 등의 개발 및 성능평가를 소프트웨어로 수행할 수 있다. 따라서, 오프-라인(Off-Line)방식으로 교시와 성능실험을 수행하면 생산라인의 작업중단도 방지할 수 있고, 동시에 작업에의 적용과 성능평가도 용이하게 된다.

본 연구에서는 산업용 로봇인 삼성 SM5와 AM1에 대한 3차원 그래픽 시뮬레이터인 OLP 시스템을 현재 일반 PC의 OS인 Windows XP 버전으로 개발하였다.

#### 2. OLP 시스템이 갖추어야 할 조건

본 연구에서 개발되어진 프로그램은 편리한 사용자 인터페이스를 위하여 다양한 그래픽적 기능들을

보유한다. 이러한 기능들을 위해서는 3차원적 데이터베이스의 구축 및 그래픽적 알고리즘의 구축이 필요하게 된다. 특히, 본 시뮬레이터는 PC에서 운용될 수 있도록 구축되고 있으므로 여기에 따른 문제점들을 해결해야만 한다.

온선 및 온면제거 기술이나 음영처리기술은 PC에서 이용되기가 상당히 까다로우며, 시뮬레이터에 적용하기 위해서 기존의 알고리즘을 수정·보완하여 적용하였다.

3차원적 그래픽을 위한 알고리즘의 개발로 시뮬레이션 결과를 3차원 영상으로 볼 수 있게 된다. 이를 위해서 실제 좌표계에서 시각 좌표계로의 변환 및 원근 투영 기술이 포함되었다.

그래픽을 이용한 시뮬레이터의 경우 가장 빈번히 사용되는 기능중의 하나가 시각위치 변경기능이므로 본 연구에서는 보다 편리하게 시각위치를 변경할 수 있는 기능을 갖추었다.

오프라인 프로그램내의 로봇과 다른 물체들에 대해 온선, 온면을 제거하고 음영 처리한 결과는 와이어 프레임으로 물체를 표현한 그림들에 비해 상당히 현실감을 높일 수 있음을 알 수 있으나, 처리속도의 저하로 인하여 온선, 온면 제거와 음영처리된 상태로 실시간 처리는 힘든 상황이므로 사용자가 한 화면을 지정하여 음영 처리된 화면을 볼 수 있도록 프로그램이 구성되어 있다.

온선 및 온면의 제거를 위하여 스캔 라인 z-버퍼 방식을 채택하였는데, 이 방식은 메모리 사용을 극단적으로 줄여 개인용 컴퓨터에서 실현이 가능하지만 실행속도의 저하를 감수해야만 한다. 음영처리 알고리즘으로는 Phong 음영처리 방식을 사용하였다.

OLP 시스템 개발시 꼭 포함되어야 할 사항으로 R. Bolles와 B. Roth는 다음의 5가지 요소들을 언급하였다.

- 사용자 인터페이스
- 궤적계획 예시레이션
- 기구학적 예시레이션
- 동특성 예시레이션
- 3차원 모델링

본 논문에서는 위의 5가지 요소에 집중하여 프로그램을 개발하였다.

### 3. 시스템의 구성 및 기능

#### 3.1 시스템 구성

Fig. 1은 개발된 OLP 시스템의 전체 구성을 나타낸다. OLP 시스템은 크게 Setting 및 Teaching, 시뮬레이션, 성능 평가의 3부분으로 크게 나뉠 수 있다. Setting 및 Teaching 부분에서는 Parameter 설정, 경로

점 교시, 궤적 생성 등이 수행되며, 시뮬레이션 부분에서는 주어진 여러 제어 알고리즘에 의한 동역학 및 제어 시뮬레이션을 할 수 있고, 성능평가 부분에서는 수치 시뮬레이션 데이터에 의한 3차원 애니메이션과 제어알고리즘의 추종 성능 검증이 수행된다.

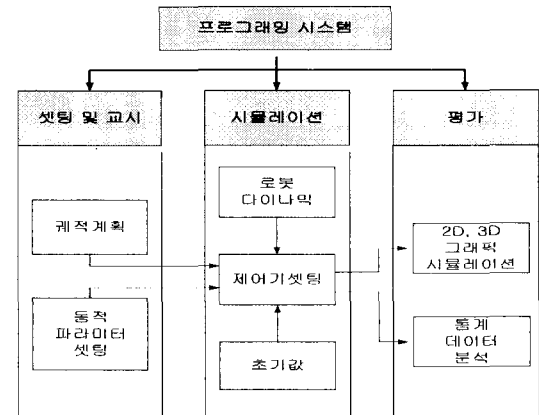


Fig. 1 The Structure of Off-Line Programming System

#### 3.2 시스템 초기 및 work range 화면

프로그램을 시작시키면 풀다운 방식의 메뉴를 기본 틀로 하는 초기창이 뜨며, 이 초기창의 클라이언트 구역에는 SM5 로봇이 그려지도록 구성되어 있다. 이때, SM5 로봇 및 AM1 로봇은 모든 Joint 값이 0인 상태로 기본자세를 취하게 된다. Fig. 2는 SM5 로봇의 work range를 나타낸다.

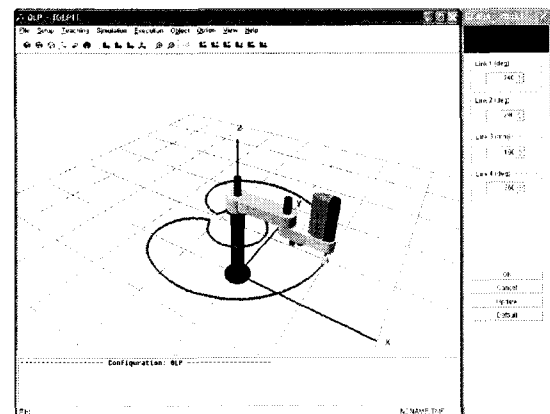


Fig. 2. The work range of OLP System(SM5)

#### 3.3 시스템 메뉴 구성

개발된 OLP 시스템의 풀다운 메뉴 구성은 다음의 표 1과 같으며, 이 메뉴중 실행 메뉴(Execution)는 DSP 보드를 이용한 OLP 시스템과 실제 스카라 로봇의 연결을 위한 것으로서 아직 기능을 연구 중에 있다.

Table 1. The Menu Structure of OLP System

File	Setup	Teaching	Simulation
Select Robot...	Specification	New	Controller Type
New Ctrl+N	Work Range	Open	Run
Open Ctrl+O	Max. Velocity	Trajectory	Evaluation
Save Ctrl+S	Load		
Save As		Object	Execution
Convert TMF	View	Create Point	On-line
Exit	Tool Bar	Create Line	Verification
	Status Bar	Create Triangle	Comparison
	Tool Bar1	Create 2D Box	
Help	Tool Bar2	Create 3D Box	
Kinematics	Output Dockbar	Create Cube	
About OLPS		Create Cone	
		Create Cylinder	
		Create Torus	
		Delete Object	
		Delete All	

### 3.3.1 파일 운영 및 메뉴

OLP 시스템은 교시점에 관한 정보, 표준 궤적 데이터, 시뮬레이션 궤적 데이터 등 모든 정보를 데이터 파일로 관리한다. 각종 정보가 저장되어 있는 파일들은 개발시 선정된 몇 가지 고유한 확장자를 가지게 되며, 작업이 수행되기 전에 파일메뉴에서 새롭게 만들거나 기존의 작성된 파일을 열어야 한다. Fig. 3은 파일 메뉴에서 Open 명령으로 확장자 inf 파일을 열려하는 모습이다.

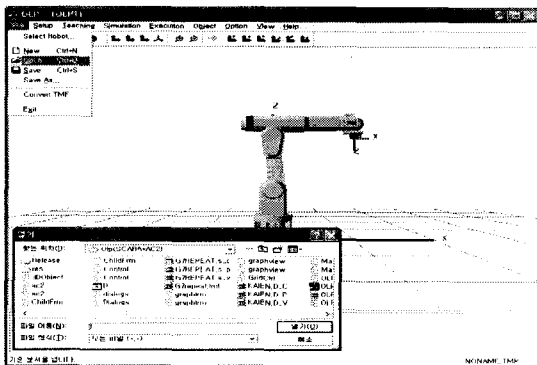


Fig. 3. Execute Open Command in FILE menu

### 3.3.2 Setup 메뉴

Setup 메뉴에서는 작업을 수행하기에 앞서 링크별 길이 및 질량, 작업 영역, 최대속도, 그리고 하중의 종류 등을 지정할 수 있다.

작업영역지정 메뉴(Work Range)에서는 각 관절의 최대 운동범위를 지정할 수 있다. 최대속도지정 메뉴(Max. Velocity)에서는 관절별로 최대 속도의 한

계를 지정할 수 있으며 이 모든 조건들은 궤적 계획시에 제한 조건으로 이용된다. 각 메뉴 선택 시는 해당하는 정보의 수정을 위하여 대화상자가 뜨게 되며, 이 상태에서 키보드나 마우스를 이용하여 원하는 수치들을 기입한다. Fig. 4는 이 대화상자들에서 데이터를 설정하는 모습이다.

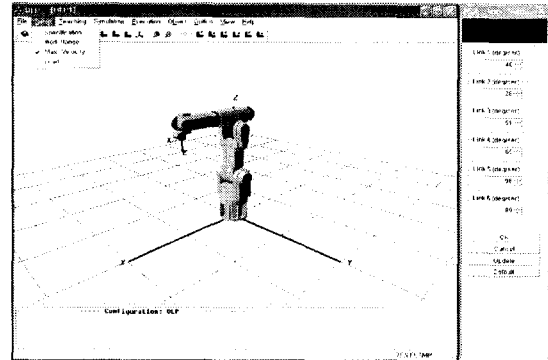


Fig. 4. Max. Velocity box of Setup Menu

### 3.3.3 교시 메뉴

교시는 교시 메뉴(Teaching)에서 대화 상자를 통해 교시 정보를 저장할 교시 파일을 만들고, 여러 교시정보를 입력한 후, 교시 파일을 저장함으로써 이루어진다. 입력하여야 할 교시 정보로는 관절 좌표계나 절대 좌표계에서의 로봇 경유점, 통과 시간 그리고 경유점 사이의 궤적 계획법이 있다.

### 3.3.4 궤적 계획

궤적 계획법으로는 Cubic Spline법, LFPB (Linear Function with Parabolic Blend)법, B-Spline법, 직선 보간법, 원호 보간법등이 준비되어 있어 이 중에서 한 가지를 선택한다. Fig. 5는 실제 로봇의 교시상자를 대체하는 OLP 시스템의 교시 대화상자와, 교시 대화상자에서 설정 값을 변경하였을 때, 자동적으로 자세를 바꾸는 로봇의 모습을 보여주고 있다.

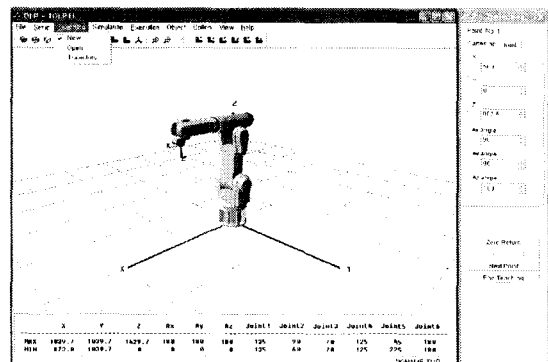


Fig. 5. Input Via\_Point information in Teach Dialog Box

### 3.3.5 동적 시뮬레이션 및 성능 예상

시뮬레이션 메뉴(Simulation)에서는 궤적계획에 이은 동역학 및 제어 시뮬레이션을 통해 수행하고자 하는 작업을 3차원과 2차원 그래픽으로 컴퓨터 화면상에서 미리 볼 수 있고 이에 이은 시뮬레이션 궤적 확인에 의해 제어 추종 성능 평가가 가능하므로 로봇에 적용된 제어 알고리즘 교사의 적합성을 판단할 수 있다.

시뮬레이션에 적용될 알고리즘으로는 기존 산업용 제어에 널리 이용되는 비례-미분(PD)제어, 로봇의 동역학 해석에 기초한 계산 토크(computed torque) 제어, 그리고 가변 구조 제어 이론을 근간으로 하는 슬라이딩 모드(sliding mode)제어가 준비되어 있는데 이 중 하나를 선택할 수 있다. Fig. 6은 주어진 제어 방식과 궤적에 따라 로봇이 동적 시뮬레이션을 하고 있는 장면이다.

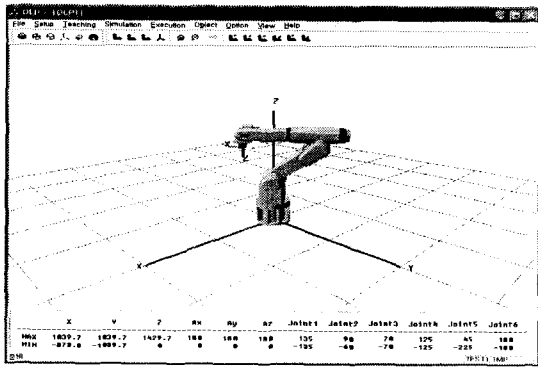


Fig. 6. Simulation in 3D Graphic space

### 3.3.6 그 외의 기능

그 외의 메뉴에서는 원하는 자세로 로봇 이동, 시 각위치 변경, 와이어 프레임 모델로 처리 등이 가능하다. Fig. 7은 성능평가를 할 수 있도록 엔코더와 OLP 데이터 간의 값을 비교하여 나타낸 장면이다.

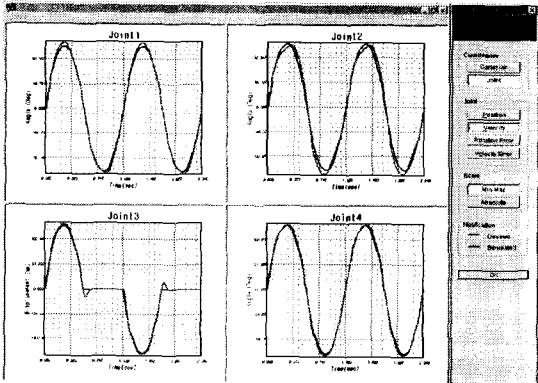


Fig. 7. Comparison of the value between encoder and OLP data

## 4. 프로그램 개발 환경

개발된 오프라인 프로그래밍 시스템은 Microsoft Visual C++ 버전 6.0과 OpenGL을 이용하여 코딩되었다. Visual C++은 Window 프로그램을 쉽고 구조적으로 제작할 수 있는 MFC와 각종 Wizard를 제공하므로, 개발된 프로그램은 향후 다른 여러 사용자들에 의해서 간단히 확장되거나 이식될 수 있다. 각종 Visual Editor들은 원하는 대화상자나, 메뉴 바를 간단히 제작할 수 있도록 한다. OpenGL은 새로운 산업용 3D 그래픽 표준으로 등장하고 있는 Library이다. Windows 상에서 빠른 3D 그래픽 능력과 간단한 프로그래밍 명령어로 프로그램 개발에 유용히 활용된다.

## 5. 결론

본 연구에서는 PC의 기본 OS인 Windows XP 환경 하에서 운용되는 4축과 6축 산업용 로봇의 통합 제어를 위한 Off-line programming system 및 3D graphic simulation tool을 개발하였다. 제안된 방법에는 기구학, 동역학, 3차원 그래픽, 궤적계획, 제어 알고리즘 등의 기능이 부가되고, 그래픽 사용자 인터페이스(GUI)에 기초한 메뉴 바 및 대화상자들로 구성되는 오프라인 프로그래밍 시스템이 개발되었으며 이를 통해 로봇의 작동과정과 구성요소의 특성을 화면상에서 평가할 수 있도록 하였다. 기존의 그래픽 공간상의 시뮬레이션과 실제 로봇의 동작과 연결시킬 수 있는 기능은 현재 연구수행 중에 있다.

## 참고문헌

1. 박민준, 손권, 안두성, "PC에서 운용되는 스카라형 로봇의 오프-라인 프로그래밍 시스템", 대한 기계학회 논문집, 제19권, pp 568-579, 1995.
2. 신기태, 이민철, 손권, 이만형, "슬라이딩 모드 제어를 이용한 SCARA 로봇의 동적 시뮬레이터 개발", '94 기계학회 춘계학술대회논문집, pp. 577~582., 1994.
3. Craig, J. J., Introduction to Robotics Mechanics and Control, Second edition Addison-Wesley, New York, 1989.
4. G. Wittenberg, "Developments in Offline Programming : An Overview", Industrial Robot, Vol.22, No.3, pp 21-23, 1995.
5. David J, and Kruglinski, Inside Visual C++ 4, Microsoft Press, 1996.
6. Ron Fosner, OpenGL Programming for Windows 95 and Windows NT, Addison-Wesley Developers Press, 1997.