

스마트 팩토리를 위한 로봇 모니터링 시스템 시뮬레이터 개발에 관한 연구

A Study on Development of Robot Monitoring System Simulator for Smart Factory

김희진^{1*}, 김상현^{1*}, 장기원^{1*}, 김두범², 동근한³, 한성현⁴

Hee-Jin Kim^{1*}, Sang-Hyun Kim^{1*}, Gi-Won Jang^{1*}, Du-Beum Kim²,
Guen-Han Dong³, Sung-Hyun Han⁴

〈Abstract〉

This study proposes a new approach to implement smart factory based on robot monitoring system for a small and medium sized enterprise. The simulator includes the forward kinematics and inverse kinematics analyzes of articulated robots. We also modeled the dynamics of the robot and made it possible to simulate it in the program. We studied the interface between the user and the monitoring programming system using the GUI environment of Windows OS, and it is configured to select the appropriate trajectory planning and control method. The reliability of simulator is illustrated by simulation test.

Keywords : Smart Factory, Monitoring System Simulator, Track Record, Manufacturing Robot, small and medium sized enterprise

1* 교신저자, 경남대학교 대학원
E-mail: spdhaos2@naver.com

2 육군정비창

3 (주)세광산업

4 정회원, 경남대학교 기계공학부

1* Corresponding author, Graduate School, Kyungnam
University

2 Maintenance Depot Co.,Ltd

3 Sekwang industry Co.,Ltd

4 Dept. of Mechanical Engineering, Kyungnam University

1. 서론

본 연구에서 제안되는 모니터링 프로그램은 편리한 사용자 인터페이스를 위하여 다양한 그래픽적 기능들을 보유하고 있다. 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. 그래픽을 이용한 시뮬레이터의 경우 가장 빈번히 사용되는 기능중의 하나가 시각위치 변경기능이므로 본 연구에서는 보다 편리하게 시각위치를 변경할 수 있는 기능을 갖추었다.

모니터링 프로그래밍 시스템 개발시 꼭 포함되어야 할 사항으로 다음의 4가지 요소들을 언급하였다. 주요 4가지 요소는 궤적계획 에뮬레이션 기능, 기구학 및 동역학적 에뮬레이션 기능, 사용자 인터페이스 기능, 그리고 3차원 모델링 기능으로 구성된다.

모니터링 프로그래밍 시스템은 공간상에서 매니플레이터가 취하는 경로를 정확히 에뮬레이트 하여야만 한다. 이 에뮬레이션은 로봇의 작업을 설정하는 부분과 충돌을 정확히 예측하는 부분에 필요하다. 이 문제에 대해 개발된 프로그램은 5개의 경로 계획 방식을 제공하여 선택하여 사용하도록 하고 있다. Windows OS에서 대화상자를 이용한 가상 교시상자를 구성하여 경로계획을 지원한다.[2]

3차원 시뮬레이션 되어진 세계를 유효하게 유지하기 위해서 필요하다. 대개 로봇의 정기구학과 역기구학을 프로그램에 포함시킴으로서 가능하게 된다. 로봇의 정, 역기구학을 개발된 프로그램에서는 수식을 풀어서 부합수를 따로 작성하였다. 여기에 기존 로봇의 세부사양 및 dimension, 그리고 무게 변화에 대처할 수 있도록 각종 파라미터 설정도 사용자가 할 수 있도록 하는 기능도 첨가시켰다.

동특성 해석에 있어서는 만약, 모니터링 프로그래밍 시스템이 제어기의 경로계획 알고리즘을 잘 에뮬레이트하고, 실제 로봇이 매우 느린 속도로 원하는 경로를 쫓아간다면, 프로그램에서 모의되는 동작의 동적인 특성은 무시할 수가 있다. 그러나, 아주 빠른 속도로 작업이 지시된 경우나 아주 높은 하중상태에서 경로를 추적하는데 발생하는 오류는 중요할 수 있다. 이러한 추적 오류를 시뮬레이트하기 위하여 정확한 동역학적 에뮬레이션이 필요하다.

이러한 문제해결을 위해 개발된 프로그램은 C++로 코딩하여 부합수화 하고, 구해진 동역학 방정식에는 로봇을 구동하는 서보계인 DC 모터에 대한 운동방정식도 함께 고려하여 보다 실제시스템에 가까운 모델링을 통해 수행한다.[3]

모니터링 프로그래밍 시스템은 사용자의 사용방법 습득 및 프로그램 데이터 관리가 쉽고 편해야 한다. 그 외에 로봇 시스템의 교시상자를 대체할 수 있는 컴퓨터상의 사용자 인터페이스 방식 개발이 필요하다.

모니터링 프로그래밍 시스템 Windows OS에서 기본적으로 제공되는 여러 가지 대화상자 및 마우스 기능, 단축키 기능 등을 사용하여 프로그램 관리 및 사용방법을 쉽게 할 수 있었다. 결국 교시상자는 대화상자 상에서 제공되는 여러 기능을 일부 변형 제작하여 최대한 쉽게 사용하게 하는데 집중했다.

컴퓨터 화면상에 제시되는 로봇 자체와, 공구, 작업셀 등이 모두 3차원 물체로 모델화되는 것을 요구한다. 또한, 이 3차원 모델화된 화면상의 물체들은 애니메이션 기능을 가져야 한다. 이 문제에 대해 개발된 프로그램에서는 실리콘 그래픽스사에서 제공하는 그래픽 라이브러리인 OpenGL를 이용하여 3차원 모델링을 구성하였다. 이 라이브러리의 사용으로 인하여 상당한 속도의 향상을 통하여 사용자의 편리성과 효율성을 증진한다.[4]

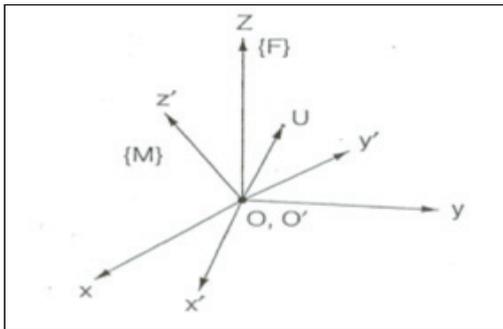
본 연구는 스마트 공장용 중소기업 보급형 제조로봇 실증라인 구축을 위한 목적으로 제조로봇 성능 평가 지원용으로 개발하는 것을 목적으로 하고 있다.

2. 제조용 로봇 운동학적 해석

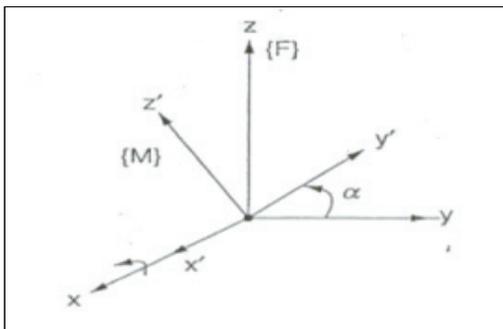
2.1 로봇 링크파라미터 해석

아래의 Fig. 2.1(a)에서 고정 좌표계 {F} 내의 점 U는 아래의 식과 같이 표현된다.

$$\vec{U} = U_1\vec{e}_1 + U_2\vec{e}_2 + U_3\vec{e}_3 \quad (2.1)$$



(a) Cartesian Coordinates {M} · {F}.



(b) Rotation coordinates(α) in x axes

Fig. 2.1 Translation coordinates of position and orientation

위의 좌표계 이동좌표계 {M}과 고정좌표계 {F}로부터 다음의 식이 정의된다.[5]

$$\vec{U}' = U_1'\vec{e}_1' + U_2'\vec{e}_2' + U_3'\vec{e}_3' \quad (2.2)$$

위의 식 (2.1)과 (2.2)식으로 부터 다음의 식을 유도할 수 있다.

$$U_1\vec{e}_1 + U_2\vec{e}_2 + U_3\vec{e}_3 = U_1'\vec{e}_1' + U_2'\vec{e}_2' + U_3'\vec{e}_3' \quad (2.3)$$

식(2.3)의 양변에 \vec{e}_1 에 대한 dot product(벡터 내적)을 구하면 U_1 은 다음식과 같이 정의된다.

$$U_1 = (\vec{e}_1' \cdot \vec{e}_1)U_1' + (\vec{e}_2' \cdot \vec{e}_1)U_2' + (\vec{e}_3' \cdot \vec{e}_1)U_3' \quad (2.4)$$

그리고 식(2.4)에서 \vec{e}_2 및 \vec{e}_3 에 대한 각각의 dot product(벡터내적)을 구하면 U_2, U_3 는 다음과 같이 정의된다.

$$U_2 = (\vec{e}_1' \cdot \vec{e}_2)U_1' + (\vec{e}_2' \cdot \vec{e}_2)U_2' + (\vec{e}_3' \cdot \vec{e}_2)U_3' \quad (2.5)$$

$$U_3 = (\vec{e}_1' \cdot \vec{e}_3)U_1' + (\vec{e}_2' \cdot \vec{e}_3)U_2' + (\vec{e}_3' \cdot \vec{e}_3)U_3' \quad (2.6)$$

식 (2.3), (2.5)그리고 (2.6)로부터, 다음의 식이 유도된다.

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = [R_M^F] \begin{bmatrix} U_1' \\ U_2' \\ U_3' \end{bmatrix} \quad (2.7)$$

여기서 R_M^F 는 이동좌표계 {M}으로부터 {F}까지의 회전행렬 변환 좌표계를 나타내고 다음의 식과 같이 주어진다.

$$R_M^F = \begin{bmatrix} \vec{e}_1 \cdot \vec{e}_1' & \vec{e}_1 \cdot \vec{e}_2' & \vec{e}_1 \cdot \vec{e}_3' \\ \vec{e}_2 \cdot \vec{e}_1' & \vec{e}_2 \cdot \vec{e}_2' & \vec{e}_2 \cdot \vec{e}_3' \\ \vec{e}_3 \cdot \vec{e}_1' & \vec{e}_3 \cdot \vec{e}_2' & \vec{e}_3 \cdot \vec{e}_3' \end{bmatrix} \quad (2.8)$$

Fig. 2.1(b)에 나타낸바와 같이 임의의 각(α)만큼 x축에 관하여 회전한 특별한 경우에 대한 회�행렬은 다음식과 같이 정의된다.

$$R_M^F(x; \alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & e_2 \cdot e_2' & e_2 \cdot e_3' \\ 0 & e_3 \cdot e_2' & e_3 \cdot e_3' \end{bmatrix} \quad (2.9)$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}$$

P^F 를 Fig. 2.1(a)에서 보여준 바와 같이 점 U 의 고정 좌표계 $\{F\}$ 에서의 벡터표현으로 정의하면, 같은점 P^F 는 이동좌표계 $\{M\}$ 내에서의 벡터 p^M 으로 표현될 수 있다. 식 (2.6)로부터 다음의 식이 유도된다.[6]

$$P^F = R_M^F P^M \quad (2.10)$$

여기서 이전의 유도된 것을 기반으로, if $d^F = \vec{0}$ 가 되고, 그리고 $P^F = R_M^F P^M$ 로 쓸 수 있다. 그러나 O 와 O' 가 일정한 거리만큼 떨어져 있으므로, $d^F \neq \vec{0}$ 가된다. 그러므로 P^F 는 다음 식과 같이 정의된다.

$$P^F = R_M^F P^M + d^F \quad (2.11)$$

여기서 R_M^F 는 이동좌표계 $\{M\}$ 과 고정좌표계 $\{F\}$ 로부터 회전 변환을 나타낸다. 그리고 d^F 는 이동좌표계의 원점 O' 과 고정 좌표계의 원점 O 사이의 고정좌표계를 나타내고 있다. 따라서 위의 Fig. 2.2로부터 다음과 같은 행렬식을 얻을 수 있다.[7]

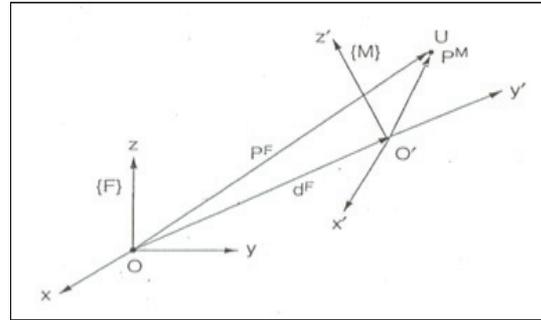


Fig. 2.2 The relation between fixed and moving coordinates.

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = R_M^F \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} + \begin{bmatrix} d_x \\ d_y \\ d_z \end{bmatrix} \quad (2.12)$$

회전 행렬과 변위벡터는 4x4 동차변환 행렬 A_M^F 의 행렬구성요소이다. 식 (2.13)은 4x4 동차변환행렬 A_M^F 을 나타낸다. 따라서 다음의 행렬식을 얻는다.[8]

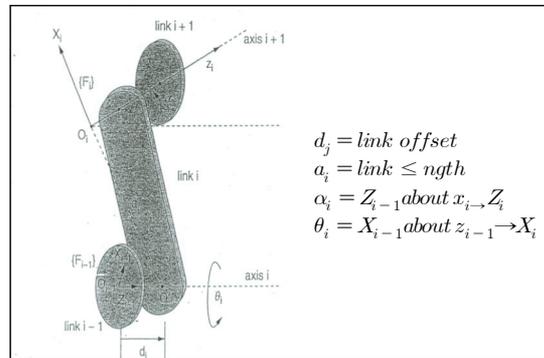


Fig. 2.3 Schematic of the adjacent axes with the assigned reference frames for determining the Denavit-hartenburg parameters.

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix}^F = \begin{pmatrix} R_M^F & d_3^F \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix}^M \Rightarrow A_M^F = \begin{pmatrix} R_M^F & d_3^F \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.13)$$

2.2 제조로봇의 기구학 해석

기준 좌표계로부터 $\{F_i\}$ 부터 기준 좌표계 $\{F_0\}$ 까지의 좌표변환을 수행하기 위해서는 회전행렬과 변위벡터를 유도해야한다.

$$R_i^{i-1} = U_i V_i \quad (2.14)$$

여기서 $U_i(\theta_i)$ 는 다음과 같이 주어진다.

$$U_i(\theta_i) = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.15)$$

따라서 $V_i(\alpha_i)$ 는 다음과 같이 주어진다.

$$V_i(\alpha_i) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha_i) & -\sin(\alpha_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix} \quad (2.16)$$

$\{F_i\}$ 에서 $\{F_{i-1}\}$ 의 좌표로부터 동차변환 행렬은 다음과 같이 주어진다.

$$A_i^{i-1} = A_i^{i-1} A_i^i \quad (2.17)$$

$$A_i^{i-1} = \begin{bmatrix} U_{i3 \times 3} & 0 \\ 0 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.18)$$

$$A_i^i = \begin{bmatrix} V_{i3 \times 3} & a_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.19)$$

따라서,

$$A_i^{i-1} = \begin{bmatrix} U_i & 0 \\ 0 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_i & a_i \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} U_i V_i & U_i s_i \\ 0 & 1 \end{bmatrix} \quad (2.20)$$

그리고 식 (2.21)을 사용하여 다양한 조건을 덧 붙이면 다음의 $U_i V_i$ 와 $U_i s_i$ 를 얻을 수 있다. [9]

$$U_i V_i = \begin{bmatrix} c\theta_i & -s\theta_i & 0 \\ s\theta_i & c\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i \\ 0 & s\alpha_i & +c\alpha_i \end{bmatrix} \\ = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i \\ 0 & s\alpha_i & +c\alpha_i \end{bmatrix}$$

$$U_i s_i = \begin{bmatrix} c\theta_i & -s\theta_i & 0 \\ s\theta_i & c\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_i \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ d_i \end{bmatrix} = \begin{bmatrix} c\theta_i a_i \\ s\theta_i a_i \\ d_i \end{bmatrix} \quad (2.21)$$

따라서, s_i 는 다음과 같이 표현된다.

$$s_i = \begin{bmatrix} a_i \\ 0 \\ d_i \end{bmatrix} \quad (2.22)$$

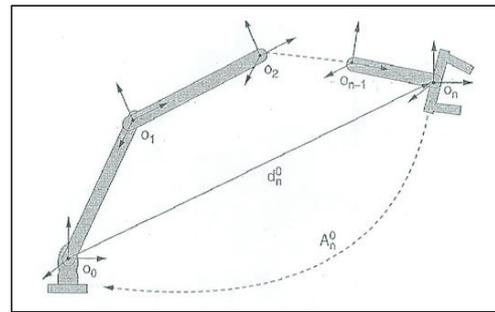


Fig. 2.4 Transformation from the end-effector frame to the base frame.

위의 Fig. 2.4로부터 n번째 좌표계로부터 기저좌 표계까지의 변환과정은 다음의 식과 같이 표현된다.

$$A_n^0 = A_1^0 A_2^1 \dots A_n^{n-1} = \begin{bmatrix} R_n^0 & d_n^0 \\ 0 & 1 \end{bmatrix} \quad (2.23)$$

식 (2.20)과 식 (2.23)을 사용하여 R_n^0 는 다음과 같이 표현된다.

$$R_n^0 = (U_1 V_1)(U_2 V_2) \cdots (U_n V_n) \quad (2.24)$$

그리고,

$$d_n^0 = U_{1s1} + U_1 V_1 U_{2s2} + U_1 V_1 U_2 V_2 U_{3s3} \cdots + U_1 V_1 U_2 V_2 \cdots U_{n-1} V_{n-1} U_{nsn} \quad (2.25)$$

본 연구는 제조공정 자동화를 위한 6관절 수직 다관절 로봇 모델을 적용하는 연구내용으로서 이에 대한 상세구조는 Fig. 2.5에서 나타내고 있다. 제안된 다관절 매니퓰레이터는 수직다관절형 로봇 매니퓰레이터로서 회전하는 6개의 조인트는 어깨, 팔꿈치, 손목형태로 사람팔과 유사하게 직렬로 배열되어 있다. 어깨 조인트(1, 2, 3)는 이 조인트 축들이 한 점에서 교차하기 때문에 스페리컬 조인트로 간주된다. 손목 조인트(4, 5, 6) 역시 같은 구조로 배열된다. 그리고 인접한 두 개의 조인트 축들은 수직으로 위치한다.[10]

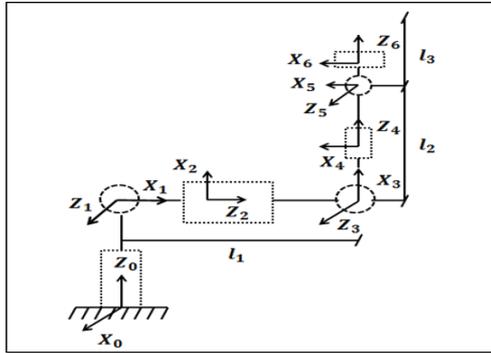


Fig.2.5 Robot coordinates frame with 6 D.O.F

운동학 관계를 설명하기 위해서 로봇 조인트의 좌표축들은 Fig. 2.5처럼 설정된다. Denavit-Hatenberg 법칙에 따라 n 자유도 로봇의 파라미터들은 Table 1에 나타난 것처럼 결정되었다. Table 1의 파라미터들은 Denavit-Hatenberg 표기법이 고유하지 않기 때문에 다른 방법들로 선택이 되었다.[11]

Table 1. Link Parameters

Joint i^{th}	a_i	d_i	α_i	θ_i
1	0	0	90	θ_1
2	0	0	90	θ_2
3	0	l_1	-90	θ_3
4	0	0	90	θ_4
5	0	l_2	-90	θ_5
6	0	0	90	θ_6

변환행렬의 일반적인 공식 다음과 같이 나타낼 수 있다.

$$T_j^i = \begin{bmatrix} c\theta_j & -s\theta_j \alpha_j & s\theta_j s\alpha_j & a_j c\theta_j \\ s\theta_j & c\theta_j \alpha_j & -c\theta_j s\alpha_j & a_j s\theta_j \\ 0 & s\alpha_j & c\alpha_j & d_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.26)$$

식 (2.26)에 Table.1의 Denavit-Hatenberg 파라미터들을 대입하기 위해 로봇 좌표들 사이의 변환행렬들은 다음 식에 따라 계산된다.

$$T_1^0 = \begin{bmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2^1 = \begin{bmatrix} c_2 & 0 & s_2 & 0 \\ s_2 & 0 & -c_2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_3^2 = \begin{bmatrix} c_3 & 0 & s_3 & 0 \\ s_3 & 0 & -c_3 & 0 \\ 0 & -1 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_4^3 = \begin{bmatrix} c_4 & 0 & s_4 & 0 \\ s_4 & 0 & -c_4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_5^4 = \begin{bmatrix} c_5 & 0 & -s_5 & 0 \\ s_5 & 0 & c_5 & 0 \\ 0 & -1 & 0 & l_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_6^5 = \begin{bmatrix} c_6 & 0 & s_6 & 0 \\ s_6 & 0 & -c_6 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.27)$$

여기서 T_j^i 는 좌표 $o_i x_i y_i z_i$ 에 대한 $o_j x_j y_j z_j$ 위치와 방향이 표현된 변환행렬이고 c_i, s_i 는 조인트 각도 i 의 코사인과 사인연산을 약어로 나타낸 것이다. 이때 로봇 엔드이펙터(끝점)의 자세는 변환 행렬의 계산에 의해 결정된다.

현재 로봇 상태 : $(\theta_1(t), \theta_2(t), \theta_3(t), \theta_4(t), \theta_5(t), \theta_6(t), \theta_7(t))$ 와 끝단(엔드이펙터) 상태는 아래와 같이 계산된다.

$$T_7^0(t) = T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 T_6^5 T_7^6 \quad (2.28)$$

$$= \begin{bmatrix} R_{Origin}^{Tip}(t) & P_{Origin}^{Tip}(t) \\ 0 & 1 \end{bmatrix}$$

여기서 R_{Origin}^{Tip} 은 회전행렬이고 P_{Origin}^{Tip} 는 글로벌 좌표계($O_0 X_0 Y_0$)에서의 엔드이펙터의 벡터 위치다. 역기구학 문제의 목적은 로봇 엔드이펙터의 다음 자세를 만족하는 새로운 조인트 $(\theta_1(t+1), \theta_2(t+1), \theta_3(t+1), \theta_4(t+1), \theta_5(t+1), \theta_6(t+1), \theta_7(t+1))$ 상태를 결정하는 것이다.[12]

$$T_7^0(t+1) = \begin{bmatrix} R_{Origin}^{Tip}(t+1) & P_{Origin}^{Tip}(t+1) \\ 0 & 1 \end{bmatrix} \quad (2.29)$$

여러개의 조인트가 한 점에서 교차하는 특정한 구조 때문에 셋 중에 첫 조인트와 마지막 조인트는 스페리컬 조인트로 간주한다.

$$J_{11} = l_2(s_4(c_1 s_3 - s_1 c_2 c_3) - s_1 c_4 s_2) - l_1 s_1 s_2$$

$$J_{12} = l_2(-s_4 c_1 s_2 c_3 + c_1 c_4 c_2) + l_1 c_1 c_2$$

$$J_{13} = l_2 s_4(s_1 c_3 - c_1 c_2 s_3)$$

$$J_{14} = l_2(c_4(s_1 s_3 + c_1 c_2 c_3) - s_4 c_1 s_2)$$

$$J_{21} = l_1 c_1 s_2 - l_2(s_4(-s_1 s_3 - c_1 c_2 c_3) - c_1 c_4 s_2)$$

$$J_{22} = l_1 s_1 c_2 - l_2(s_4 s_2 c_3 s_1 - c_4 s_1 c_2)$$

$$J_{23} = -l_2 s_4(c_1 c_3 + c_2 s_3 s_1)$$

$$J_{24} = -l_2(c_4(c_1 s_3 - c_2 c_3 s_1) + s_4 s_1 s_2) \quad (2.30)$$

$$J_{31} = 0$$

$$J_{32} = l_1 s_2 - l_2(-s_2 c_4 - c_3 c_2 s_4)$$

$$J_{33} = -l_2 s_3 s_2 s_4$$

$$J_{34} = -l_2(-c_2 s_4 - c_3 s_2 c_4)$$

네 번째 조인트의 각도 값은 앞에서 언급했던 과정과 같이 결정된다. 따라서 각속도는 쉽게 얻을 수 있다.[13]

$$\dot{\theta}_4 = \frac{\theta_4(t+1) - \theta_4(t)}{\Delta t} \quad (2.31)$$

식 (2.40)를 확장하면 선형 방정식 세트는 다음과 같이 설정된다.

$$\left(\frac{\partial}{\partial \theta_1} f_1\right) \dot{\theta}_1 + \left(\frac{\partial}{\partial \theta_2} f_1\right) \dot{\theta}_2 + \left(\frac{\partial}{\partial \theta_3} f_1\right) \dot{\theta}_3 = \partial Cx - \left(\frac{\partial}{\partial \theta_4} f_1\right) \dot{\theta}_4$$

$$\left(\frac{\partial}{\partial \theta_1} f_2\right) \dot{\theta}_1 + \left(\frac{\partial}{\partial \theta_2} f_2\right) \dot{\theta}_2 + \left(\frac{\partial}{\partial \theta_3} f_2\right) \dot{\theta}_3 = \partial Cy - \left(\frac{\partial}{\partial \theta_4} f_2\right) \dot{\theta}_4$$

$$\left(\frac{\partial}{\partial \theta_1} f_3\right) \dot{\theta}_1 + \left(\frac{\partial}{\partial \theta_2} f_3\right) \dot{\theta}_2 + \left(\frac{\partial}{\partial \theta_3} f_3\right) \dot{\theta}_3 = \partial Cz - \left(\frac{\partial}{\partial \theta_4} f_3\right) \dot{\theta}_4 \quad (2.32)$$

앞 절에서 언급한 것처럼 손목 위치는 손목 좌표로 독립적으로 얻을 수 있다. 게다가 행렬 (2.32)에서 determinant는 0이다(행렬이 3보다 작은 rank를 가진다). 그러므로 행렬의 세 식은 독립적이지 않고 유일해를 결정할 수 없다. 그러나 어깨 조인트 그룹 사이의 관계는 추정될 수 있다. $J_{31} = 0$ 이기 때문에 아래의 제약을 얻기 위해 (2.32)의 첫 번째와 마지막 식을 선택한다.[14]

$$\dot{\theta}_1 = A + B\dot{\theta}_3$$

$$\dot{\theta}_2 = C + D\dot{\theta}_3 \quad (2.33)$$

여기서 A, B, C, D는 다음과 같이 계산된다.

$$A = \left(\partial Cx - \left(\frac{\partial}{\partial \theta_4} f_1\right) \dot{\theta}_4 - \frac{J_{12} \left(\partial Cz - \left(\frac{\partial}{\partial \theta_4} f_4\right) \dot{\theta}_4 \right)}{J_{32}} \right) \frac{1}{J_{11}}$$

$$B = \frac{J_{11}J_{33} - J_{13}J_{32}}{J_{11}} \quad C = \frac{\partial Cz - \left(\frac{\partial}{\partial \theta_4} f_4\right) \dot{\theta}_4}{J_{32}}$$

$$D = -\frac{J_{33}}{J_{32}} \quad (2.34)$$

식 (2.42)로부터 어깨 조인트 각사이의 관계가 결정된다. 3개의 변수와 2개의 식이 있으므로 유일해를 계산하기 위해서는 코스트 함수라 불리는 하나의 식이 제안되어야 한다.

3. 모니터링 시뮬레이터의 구조

모니터링을 위한 시뮬레이터의 프로그래밍 시스템 개발시 꼭 포함되어야 할 사항으로 다음의 요소들을 언급하였다. 주요 기능으로는, 기구학 및 동역학적 에뮬레이션 기능, 사용자 인터페이스 기능, CAD 파일 기반 3차원 모델링 기능, 그리고 작업장 환경 모델링기능으로 구성된다.[15]

3.1 기구학 및 작업궤적 분석기능

3차원 시뮬레이션 되어진 세계를 유효하게 유지하기 위해서 필요하다. 대개 로봇의 정기구학과 역기구학을 프로그램에 포함시킴으로서 가능하게 된다. 로봇의 정, 역기구학을 개발된 프로그램에서는 수식을 풀어서 부함수를 따로 작성하였다. 여기에 기존 로봇의 세부사양 및 dimension, 그리고 무게 변화에 대처할 수 있도록 각종 파라미터 설정도 사용자가 할 수 있도록 하는 기능도 첨가시켰다.

동특성 해석에 있어서는 만약, 모니터링 프로그래밍 시스템이 제어기의 경로계획 알고리즘을 잘 분석되고, 실제 로봇이 매우 느린 속도로 원하는

경로를 쫓아간다면, 프로그램에서 모의되는 동작의 동적인 특성은 무시할 수가 있다. 그러나, 아주 빠른 속도로 작업이 지시된 경우나 중 하중상태에서 경로를 추적하는데 발생하는 오류는 중요할 수 있다. 이러한 추적 오류를 시뮬레이션하기 위하여 정확한 동역학적 에뮬레이션이 필요하며, 구해진 기구학 및 동역학 방정식에는 로봇을 구동하는 서보계인 DC/AC 모터에 대한 운동특성도 함께 고려하여 보다 실제시스템에 가까운 모델링을 통해 수행하였다.

3.2 사용자 인터페이스

모니터링 프로그래밍 시스템은 사용자의 사용방법 습득 및 프로그램 데이터 관리가 쉽고 편해야 한다. 그 외에 로봇 시스템의 교시상자를 대체할 수 있는 컴퓨터상의 사용자 인터페이스 방식 개발이 필요하다. 프로그래밍 시스템 Windows OS에서 기본적으로 제공되는 여러 가지 대화상자 및 마우스 기능, 단축키 기능 등을 사용하여 프로그램 관리 및 사용방법을 쉽게 할 수 있었다. 결국 교시상자는 대화상자 상에서 제공되는 여러 기능을 일부 변형 제작하여 최대한 쉽게 사용하게 하는데 집중했다.[16]

3.3 로봇 및 작업환경 모델링

컴퓨터 화면상에 제시되는 로봇 자체와, 공구, 작업셀 등이 모두 3D 물체로 모델화되는 것을 요구한다. 또한, 이 3D 모델화된 화면상의 물체들은 애니메이션 기능을 가져야 한다. 이 문제에 대해 개발된 프로그램에서는 실리콘 그래픽스사에서 제공하는 그래픽 라이브러리인 OpenGL를 이용하여 3D 모델링을 구성하였다. 이 라이브러리의 사용으로 인하여

상당한 속도의 향상이 가능했으며, 고정밀도 애니메이션시 기존의 GDI (Graphic Device Interface)에서 존재했던 약간의 깜박거림도 없앨 수 있었다.

3.4 Track record 및 데이터 정립

제조로봇 길증라인을 구축하는 많은 부분 중 가장 기본이 되는 요소로서 로봇시스템, 작업 공정라인, 제어시스템, 그리퍼 등의 자동화유닛 등의 데이터 검증 및 상태 분석을 위한 Track record가 필요하다. 스마트공장 실증라인을 구축하기 위해서는 이 요소가 모듈화 되어 데이터베이스로 저장된다. 새로운 작업 셀을 구축하는 경우 작업 셀 중에 사용자가 필요로 하는 모듈을 선택하여 원하는 위치에 배치함으로써 작업공정 효율을 극대화 시킬 수 있다. 작업장 내의 물체들을 모델링을 위해 3차원 박스 또는 위치, 속도 등의 데이터를 검증한다.

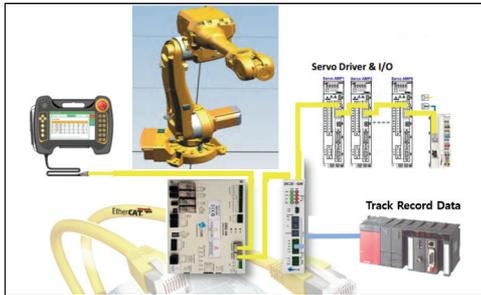


Fig. 3.1 The basic scheme of track record

3.5 3D 그래픽 구현

2차원 화면을 현실감 있는 3차원 모델로 표현하기 위해 좌표변환을 이용하면 원하는 스크린상의 3차원 이미지구현이 가능해진다. 물체를 화면에 3차원으로 그리기 위해서는 표현기준의 설정이 필요하다. 본 연구에서는 실제좌표계에서 물체의

꼭지점을 기준점으로 모델링하였고, 시뮬레이션을 하는 경우에는 관절 각도값을 변수로 하는 동차변환행렬을 이용하여 물체가 운동을 한 경우의 형상에 대한 최종 좌표값을 구한다.[17]

화면상의 3차원 표현을 위해서는 실제좌표계를 시각좌표계로 변화시켜야 한다. 즉 관찰자의 눈이 시각좌표계의 원점에 있다고 간주하여 관찰자의 입장에서 물체의 좌표값을 구하였다.

4. 제조로봇 모니터링시스템 시뮬레이터의 성능시험

4.1 모니터링 시뮬레이터의 성능

모니터링 프로그래밍 시스템은 크게 세팅 및 작업교시, 시뮬레이션, 성능 평가의 부분으로 크게 나눌 수 있다. 세팅 및 교시 부분에서는 파라미터 설정, 작업 교시, 작업 경로 및 궤적 생성 등이 수행되며, 시뮬레이션 부분에서는 주어진 다양한 제어 알고리즘에 의한 동역학 및 제어 시뮬레이션을 할 수 있고, 성능평가 부분에서는 수치 시뮬레이션 데이터에 의한 3차원 애니메이션과 제조로봇의 동작 및 운영상태의 모니터링 및 성능 검증이 수행된다.

4.1.1 파일 운영 및 메뉴 기능 시험

모니터링 프로그래밍 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. 4.1는 파일 메뉴에서 어떠한 모드가 있는지 나타내는 모습이다.

4.1.2 시뮬레이터 메뉴 기능 검증 및 분석

시뮬레이터 메뉴에서는 작업을 수행하기에 앞서 링크별 길이 및 질량, 작업 영역, 최대속도, 그리고 하중의 종류 등을 지정할 수 있다. 링크별 길이 및 질량지정 메뉴에서는 각 링크의 길이를 mm(혹은 cm)단위로 그리고, 질량을 g(Kg)단위로 지정할 수 있다. 작업영역지정 메뉴에서는 각 관절의 최대 운동범위를 지정할 수 있으며, 속도지정 메뉴에서는 관절별로 속도를 지정할 수 있다.

Fig. 4.1은 실제 로봇의 교시상자를 대체하는 모니터링 프로그래밍 시스템의 교시 대화상자와, 교시메뉴에서 설정 값을 입력하였을 때, 자동적으로 경로를 바꾸는 로봇의 동작상태를 보여주고 있다.

시뮬레이션 메뉴에서는 작업경로의 궤적에 따른 시뮬레이션을 통해 수행하고자 하는 작업동작 상태를 3차원과 2차원 그래픽으로 컴퓨터 화면상에서 미리 볼 수 있으며, 이에 이은 시뮬레이션 결과의 확인에 의해 성능 평가가 가능하므로 로봇동작 상태 및 성능을 미리 분석 점검할 수 있다.

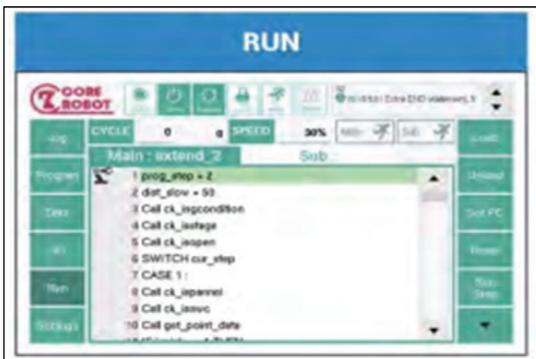


(b) execution menu



(c) Jog mode menu

Fig. 4.1 The Simulator scheme of monitoring system for manufacture robot



(a) 3D modeling

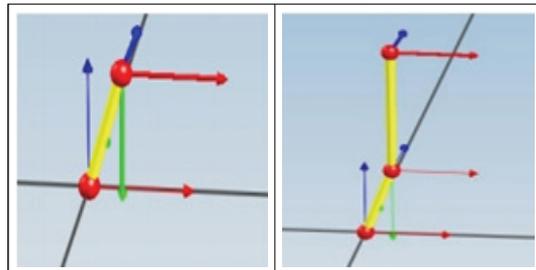
Id	X	Y	Z	RX	RY	RZ	Parent	Type	Joint#	DepId	Scale
1	0	0	0	0	0	0		Fixed	0	1	0
2	0	0	0	0	0	0	1	Revolu	1	1	0
3	175	0	495	-90	0	0	2	Revolu	2	1	0
4	0	-900	0	0	0	0	3	Revolu	3	1	0
5	960	-175	0	0	90	0	4	Revolu	4	1	0
6	0	0	0	0	90	0	5	Revolu	5	1	0
7	0	0	0	0	0	90	6	Revolu	6	1	0
8	0	0	135	0	0	90	7	Fixed	0	1	0

Fig. 4.2 Link parameters of robot manipulator



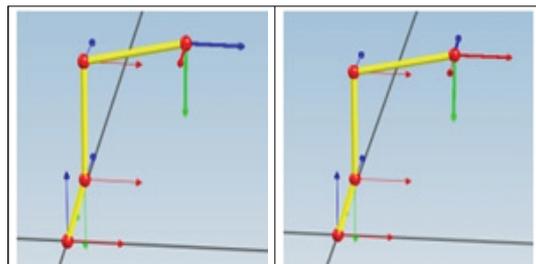
(a) Base

(b) Link 1



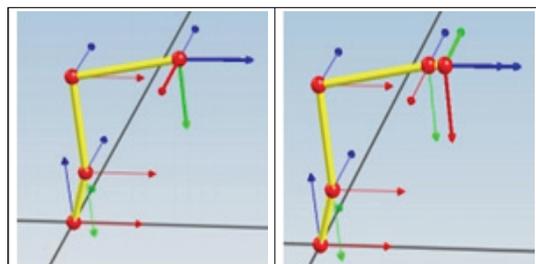
(c) Link 2

(d) Link 3



(e) Link 4

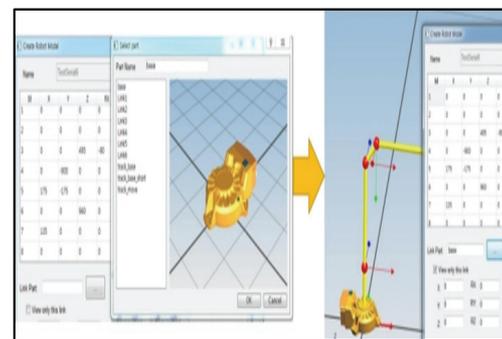
(f) Link 5



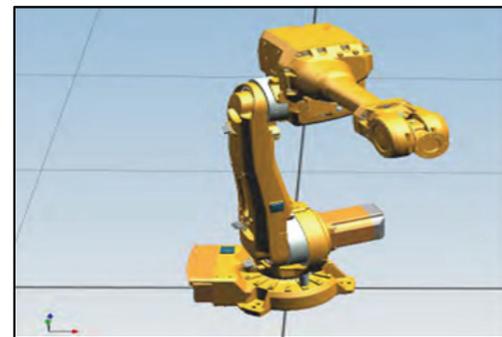
(g) Link 6

(h) Tool

Fig. 4.3 The model generation of robot link arm according to link parameter



(a) 3D model generation of base frame



(b) 3D model generation of robot with 6 D.O.F

Fig. 4.4 3D modeling generation of link arm according to CAD files in simulator

5. 맺음말

본 연구에서 제안된 중소기업 보급형 스마트공장 실증라인 적용을 위한 모니터링 프로그래밍 시스템은 크게 셋팅 및 교시, 시뮬레이션, Track Recorder, 기능 상태진단 분석기능으로 크게 나눌 수 있다. 셋팅 및 교시 부분에서는 파라미터 설정, 경로점 교시, 궤적 생성 등이 수행되었으며, 시뮬레이션 부분에서는 주어진 여러 알고리즘에 의한 동역학 및 시뮬레이션을 할 수 있고, 성능분석 부분에서는 수치 시뮬레이션 데이터에 의한 3차원 애니메이션과 제조로봇의 I/O신호 및

로봇운행상태 진단분석이 가능한 구조로 설계되었다.

기존의 작업장 도면을 시뮬레이터 프로그램에 이용하기 위해 CAD 파일을 입력하는 기능이 필요하다. 실제적으로 2차원 파일이 아닌 3차원 파일의 입력이 필요하다. 3차원 CAD 프로그램 중 많이 사용되는 ACAD의 파일과 3D 파일을 입력받는다. 입력받은 CAD 도면은 파일로 데이터베이스화 되어 사용자가 시스템 내에 적용하기 위해 위치, 속도 정보에 대해 수정을 할 수 있다.

스마트 팩토리를 위한 중소기업 보급형 제조로봇 실증라인에 적용하여 중소기업제조기술혁신에 효율적으로 적용될 수 있을 것으로 사료된다.

사 사

본 연구는 산업부지원 산업기술혁신사업(과제번호: 20005020)에 의해 수행되었습니다.

참고문헌

- [1] K.J. Kyriakopolos and G.N. Saridis. "Distance estimation and collision prediction on-line robotic motion planning", *Automatica*, Val. 28, No. 2, pp. 2, 389-394, (1992).
- [2] 이형묵, 양지훈, 이정환, 박성미, 박성준, "소프트 절환이 가능한 계통 연계형 CTTs 시스템 구성에 관한 연구", *한국산업융합학회 논문집* 제 21권 6호 pp. 361-368, (2018).
- [3] Han Sung Kim "Development of a 4-DOF industrial Robot System" *한국산업융합학회 논문집* 제 21권 1호, pp. 37-44, (2018).
- [4] Tae-seok Jin, "Simulation of Mobile Robot Navigation based on Multi-Sensor Data Fusion by Probabilistic Model" *한국산업융합학회 논문집* 제21권 4호, pp.167-174, (2018).
- [5] Johann Borenstein and Yoram Koren, "Error Eliminating Rapid Ultrasonic Firing For Mobile Robot Obstacle Avoidance," *IEEE Translation on Robotics and Automation*, Vol .11 No. 1, pp. 135-138, (1995).
- [6] 정영석, "회전자 위치센서 없는 동기전동기의 고속 운전 성능 개선" *한국산업융합학회 논문집* 제21권 6호, pp. 439-444, (2018).
- [7] L.A. Zadeh, "Fuzzy Sets," *Information&Control*, Vol 8, pp. 338-353, (1965).
- [8] S.M. Smith and D.J. Comer "Automated Calibration of a Fuzzy Logic Controller Using a Cell State Space Algorithm," *IEEE Control System*, pp. 18-28, (1991).
- [9] U Neis, K Nickel and a Tiehm, "Enhancement of anaerobic sludge".
- [10] U Neis, K Nickel and A Tiehm. "Enhancement of anaerobic sludge digestion by ultrasonic distegration," *Proc. of IEEE Intenational Conference on Robotics and Automation*, pp. 384-389, (2000).
- [11] Y. Kanayama, Y. Kimura, F. Miyazaki and T. Noguchi, "A Stable Tracking Control Method for an Autonomous Mobile Robot," *Proc. of IEEE Intenational Conference on Robotics and Automation*, pp. 384-389, (1990).
- [12] Woo-song Lee, Min-Seong Kim, Ho-Yong Bae, Yang-Keun Jung, Young-Hwa Jung, Gi-Soo Shin, In-Man Park, Sung-Hyun Han "A Study on Stable Motion Control of Humanoid Robot with 24 Joints Based on Voice Command" *한국산업융합학회 논문집* 제21권 1호, pp. 17-28, (2018).
- [13] 김한성, "위치/힘 동시제어를 위한 F/T측정 기능을 갖는 6축 순응기구 설계" *한국산업융합학회 논문집* 제 21권 2호, pp. 63-70, 2018
- [14] Won-Sik Choi, Soon-Goo Kwon, "Analysis of the Reduction Gear in Electric Agricultural Vehicle," *한국산업융합학회 논문집* 제21권 4호 pp. 159-166, (2018).
- [15] 이형묵, 양지훈, 이정환, 박성미, 박성준, "소

- 프트 절환이 가능한 계통 연계형 CTTS 시스템 구성에 관한 연구” 한국산업융합학회 논문집 제21권 6호, pp. 361-368, (2018).
- [16] 김두범, 김희진, 백영태, 배호영, 한성현, “스마트 팩토리를 위한 제조공정내에서 모바일 로봇의 장애물 회피 및 자율주행에 관한 연구” 한국산업융합학회 논문집 제21권 6호, pp. 379-388, (2018).
- [17] 백준영, 고석조, 김태훈, 운상민, 노치범, 차병수, 이민철, “운동 보초 및 운동 정버 모니터링이 가능한 오토 트레이닝 시스템 개발에 관한 연구” 한국산업융합학회 논문집 제20권 4호, pp. 333-338, (2017).

(접수: 2019.06.05. 수정: 2019.08.22. 게재확정: 2019.09.09.)