

산업용 FANUC Robot의 kinematics와 inverse kinematics에 대한 연구

박 형준, 한 덕수, 이 쾌희
서강대학교 이공대학 전자공학과

A study on Kinematics and Inverse Kinematics of Industrial FANUC Robot

Hyoung Jun Park, Duk Soo Han, Kwae Hi Lee
Dept. of EE, Sogang Univ.

Abstract

This paper deal with the solution of kinematics and inverse kinematics of industrial FANUC robot by the bisection method with IBM PC 386. The inverse kinematics of FANUC robot cannot be solved by the algebraical method, because arm matrix T_6 is very complex and 6-joint angles are associated with the position and the approach of end-effector. Instead we found other 5-joint angle by an algebraical method after finding θ_4 value by a bisection method.

I. 서 론

현대 산업 사회의 고도화는 여러 분야에서의 자동화에 힘입어 더욱 가속화되고 있고 자동화의 중추적 역할을 하는 것은 다름아닌 robot이다. Robot는 고도의 정밀화 작업뿐만 아니라 물품의 검사, 고장의 발견 및 수리 등 인간의 일을 대신하게 되었고 나아가서 우주에서의 작업, 심해의 지질 조사 등 그 이용 범위가 인간이 할 수 없는 작업까지 확대되어 가고 있다. 이와 같이 robot의 사용 용도와 분야가 사회 전반에 걸쳐 광범위해짐에 따라 인간이 교시기구(Teach pendant)를 이용하여 robot를 제어하는 단순한 play-back 방식에서 벗어나 스스로 판단하고 제어할 수 있는 지능을 가진 robot가 필요하게 되었고 이러한 지능적이고 복잡한 작업의 실시간 처리가 요구되었다.

실제 robot의 운용은 robot 각 joint angle이 주어졌을 때 end-effector의 위치와 접근 방향을 구해내는 kinematics와 end-effector의 위치와 접근 방향이 주어졌을 때 robot의 각 joint angle을 구하는 inverse-kinematics를 이용한다. 따라서 robot로 하여금 실시간 작업을 하게 하기 위해서는 kinematics와 inverse-kinematics의 처리 시간을 줄여야만 한다.

교육용 robot는 그 구조가 비교적 간단하여 기하학적 방법을 이용하여 각 joint의 대수적인 유일한 해를 구할 수 있으나 고도의 정밀 작업에 필요한 정밀도가 산업용 robot에 비해 다소 떨어진다. 일반적인 산업용 robot는 교육용 robot에 비해서 더 큰 힘을 낼 수 있고 정밀하게 위치를 제어할 수 있는 반면에 구조적으로 매우 복잡하다는 단점이 있다. 이러한 단점 때문에 교육용 robot와 같이 기하학적 방법으로는 대수적인 유일한 해를 구하기 어렵다.

본 논문은 일본 FANUC사의 산업용 robot인 FANUC Arc Mate의 실시간 운용을 위해 inverse-kinematics의 해를 기존의 반복법을 이용하여 구하는 방식과 달리 이분법을 이용하여 구하는 방식을 제안하여 그 처리 시간을 단축하였다.

II. 본 론

1. FANUC ROBOT ARC Mate 의 구조

본 논문에서는 산업용 robot인 FANUC ROBOT ARC Mate (이하 FANUC robot)를 사용하여 실험을 하였다. FANUC robot은

6개의 관절로 구성되어 있는 자유도(degree of freedom)가 6인 다관절 robot로서 일반적인 교육용 robot에 비하여 구조적으로 매우 복잡하고 큰 힘을 가지며 0.01°까지의 joint angle과 0.01mm까지의 위치를 제어할 수 있다 [1]. 그림 1에서 FANUC robot의 기하학적 구조를 보여주고 있다. FANUC robot은 작업 환경내에서 robot에 직접 작업을 시키면서 program하는 on-line 방식이며 teach pendant를 사용하여 작업을 지시한 후 이를 반복 작업시키는 play-back 방식의 robot이다.

산업용 FANUC robot을 이용하여 지능적인 작업을 수행하려면 먼저 end-effector의 위치와 접근 방향을 알아내는 kinematics를 구하는 일과 가고자 하는 위치와 접근 방향으로 end-effector를 이동시킬 각 joint-angle을 구하는 inverse kinematics의 문제를 해결해야 한다.

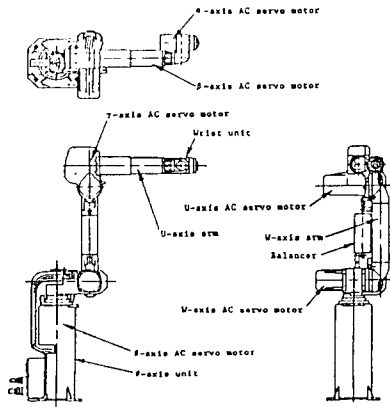


그림 1. FANUC robot의 기하학적인 구조

2. Kinematics

Robot arm의 각 joint angle이 주어졌을 때 end-effector의 위치와 접근방향을 구해내는 것을 kinematics라 하며 여기에는 여러가지 방법이 있으나[2] 본 논문에서는 가장 일반적인 Denavit-Hartenberg 방법에 의한 kinematics를 사용하였다 [3]. FANUC robot의 각 6개의 link 좌표계는 그림 2와 같이 결정하였고, 각 관절의 동작 범위는 그림 3과 같고 link 좌표변수(link coordinate parameter)는 $\theta_i, \alpha_i, d_i, a_i$ 를 사용하였으며, 좌표계 사이의 변환은 homogeneous transformation인 matrix를 사용하였다. θ 와 α 는 각도로 나타내고, d, a 는 길이로 나타낸다. 6개의 관절에는 각각의 좌표계를 갖고 있으며 각각의 좌표계는 $XYZ_0 \sim XYZ_5$ 로 표시하고 임의의 관절을 i 로 표시하여 link 좌표변수를 정의하면

표 1과 같다.

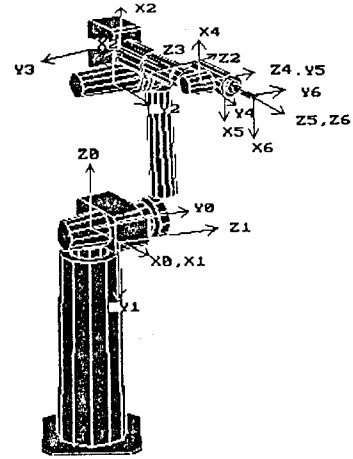


그림 2. FANUC robot의 축 지정.

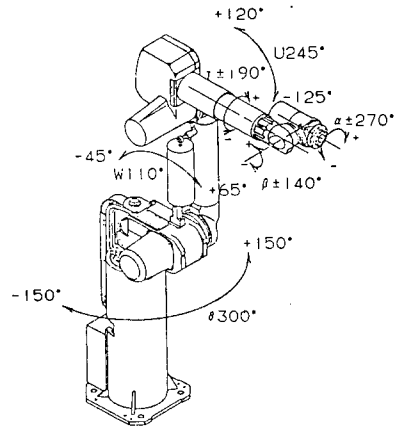


그림 3. FANUC robot의 동작 범위.

표 1. FANUC robot의 link 좌표 변수 [1] 단위 [cm]

joint i	θ_i	α_i	d_i	a_i	동작 범위
1	0°	-90°	20	0	-150° ⇔ 150°
2	-90°	0°	60	0	-45° ⇔ 65°
3	0°	-90°	13	3	-125° ⇔ 120°
4	0°	90°	0	55	-190° ⇔ 190°
5	180°	90°	0	10	-140° ⇔ 140°
6	0°	0°	0	25	-270° ⇔ 270°

θ_i : Z_i 축에 대해 X_{i-1} 축이 X_i 축으로 회전하는 데 필요한 각도

α : X_i 축에 대해 Z_{i-1} 축이 Z_i 축으로 회전하는 데 필

요한 각도

- d : X_i 축 방향으로 Z_{i-1} 축과 X_i 축의 교점에서 i 축의 원점까지의 거리
- a : Z_{i-1} 축 방향으로 $i-1$ 축의 원점에서 Z_{i-1} 축과 X_i 축의 교점까지의 거리

각 좌표계는 기준좌표(base coordinate)에 대한 관절 좌표의 표시이며 Z 축을 다음 link의 방향으로 지정하였다. $n-1$ 번째 link의 좌표계에서 n 번째 link의 좌표계로의 homogeneous transformation 관계를 A_n matrix로 표현하며 이들간의 관계는 4×4 행렬로 나타낸다. 일반적인 식으로 표시하면 식(1)과 같다.

$$A_n = \begin{bmatrix} c\theta & -s\theta c\alpha & s\theta s\alpha & ac\theta \\ s\theta & c\theta c\alpha & -c\theta s\alpha & as\theta \\ 0 & s\alpha & c\alpha & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

여기서 $s\theta = \sin\theta$ $c\theta = \cos\theta$
 $s\alpha = \sin\alpha$ $c\alpha = \cos\alpha$

$n-1$ 번째 좌표계에 대한 end-effector 좌표계의 관계는 U_n 으로 나타낸다 [4].

$$U_n = A_n * A_{n-1} * \dots * A_6 \quad (2)$$

Robot arm의 기준 좌표계에 대한 end-effector의 변환 관계는 U_1 이며 이는 arm matrix T_6 로 표현된다.

$$T_6 = U_1 = A_1 * A_2 * A_3 * A_4 * A_5 * A_6 \quad (3)$$

FANUC robot의 6개 link 사이의 homogeneous transformation matrix는 식(4) ~ 식(9)와 같으며 원하는 FANUC robot arm matrix T_6 는 식(10) ~ 식(14)의 과정을 통하여 식(15)의 형태로 얻어진다 [5].

$$A_1 = \begin{bmatrix} c1 & 0 & -s1 & 20c1 \\ s1 & 0 & c1 & 20s1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4) \quad A_2 = \begin{bmatrix} c2 & -s2 & 0 & 60c2 \\ s2 & c2 & 0 & 60s2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

$$A_3 = \begin{bmatrix} c3 & 0 & -s3 & 13c3 \\ s3 & 0 & c3 & 13s3 \\ 0 & -1 & 0 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6) \quad A_4 = \begin{bmatrix} c4 & 0 & s4 & 0 \\ s4 & 0 & -c4 & 0 \\ 0 & 1 & 0 & 55 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

$$A_5 = \begin{bmatrix} c5 & 0 & s5 & 0 \\ s5 & 0 & -c5 & 0 \\ 0 & 1 & 0 & 10 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8) \quad A_6 = \begin{bmatrix} c6 & -s6 & 0 & 0 \\ s6 & c6 & 0 & 0 \\ 0 & 0 & 1 & 25 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

$$U_6 = A_6 \quad (10)$$

$$U_5 = A_5 * A_6 \quad (11)$$

$$U_4 = A_4 * A_5 * A_6 \quad (12)$$

$$U_3 = A_3 * A_4 * A_5 * A_6 \quad (13)$$

$$U_2 = A_2 * A_3 * A_4 * A_5 * A_6 \quad (14)$$

$$U_1 = A_1 * A_2 * A_3 * A_4 * A_5 * A_6 \quad (15)$$

$$= T_6 = \begin{bmatrix} N_x & O_x & A_x & P_x \\ N_y & O_y & A_y & P_y \\ N_z & O_z & A_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

T_6 의 요소들의 물리적 의미는 N_{xyz} , O_{xyz} , A_{xyz} , P_{xyz} 는 각각 end-effector의 수직, 방향, 접근 그리고, 위치의 벡터 성분이다. 식(15)로부터 FANUC robot의 kinematics를 구해낼 수 있고 그것을 식(16)~식(27)에 나타내었다.

$$N_x = c1c23(c4c5c6+s4s6)-c1s23s5s6+s1(s4c5c6-c4c6) \quad (16)$$

$$N_y = s1c23(c4c5c6+s4s6)-s1s23c5c6-c1(s4c5c6-c4s6) \quad (17)$$

$$N_z = -s23(c4c5c6+s4s6)-c23s5c6 \quad (18)$$

$$O_x = -c1c23(c4c5s6-s4c6)-s1(s4c5s6+c4c6)+c1s23s5s6 \quad (19)$$

$$O_y = -s1c23(c4c5s6-s4c6)+c1(s4c5s6+c4c6)+s1s23s5s6 \quad (20)$$

$$O_z = s23(c4c5s6-s4c6)+c23s5s6 \quad (21)$$

$$A_x = c1(c23c4s5+s23c5)+c23s5s6 \quad (22)$$

$$A_y = s1(c23c4s5+s23c5)-c1s4s5 \quad (23)$$

$$A_z = c23c5-s23c4s5 \quad (24)$$

$$P_x = 25A_x+c1[c23(10s4+13)-55s23+60c2+20]-s1(10c4+3) \quad (25)$$

$$P_y = 25A_y+s1[c23(10s4+13)-55s23+60c2+20]+c1(10c4+3) \quad (26)$$

$$P_z = 25A_z-s23(10s4+13)-55c23-60s2 \quad (27)$$

3. Inverse kinematics

Inverse kinematics는 작업 대상의 위치와 접근 방향이 주어졌을 때 end-effector가 주어진 접근 방향과 위치로 움직일 수 있는 각 joint angle의 값을 구하는 것을 말하며, 일반적으로는 기하학적 방법, 반복 계산법과 대수적 방법을 사

용한다.

기하학적 방법은 robot arm의 각 link 사이의 기하학적 구조에서 임의의 위치와 접근 방향에 대한 각 joint angle을 계산해 내는 방법으로 구조가 간단한 교육용 robot에 주로 이용되며, 대수적 방법은 arm matrix T_6 와 이를 구하는데 사용된 homogeneous transformation matrix A_n 의 역행렬을 사용하여 식(10) ~ 식(14)의 식들과 다음의 식들을 비교하여 수학적으로 해를 구해내는 방법이다. 대수적인 방법은 구해진 해가 정확한 반면에 많은 식들 중에서 주어진 robot arm의 형태에 맞는 해를 찾아내기 위한 식들을 선택하는 것이 매우 어려우며 반복 계산법은 시간이 너무 많이 걸린다는 단점이 있다 [6].

$$A_1^{-1} * T_6 = U_2 \quad (28)$$

$$A_2^{-1} * A_1^{-1} * T_6 = U_3 \quad (29)$$

$$A_3^{-1} * A_2^{-1} * A_1^{-1} * T_6 = U_4 \quad (30)$$

$$A_4^{-1} * A_3^{-1} * A_2^{-1} * A_1^{-1} * T_6 = U_5 \quad (31)$$

$$A_5^{-1} * A_4^{-1} * A_3^{-1} * A_2^{-1} * A_1^{-1} * T_6 = U_6 \quad (32)$$

산업용 FANUC robot의 inverse kinematics의 해를 구하는 데 있어 기하학적 해석은 FANUC robot의 γ , β , α 축이 end-effector의 위치와 접근 방향에 관계되는 기하학적인 특성 때문에 해를 찾기가 불가능하다 (그림 3).

또한 일반적인 교육용 robot와는 달리 FANUC robot의 6개 모든 joint angle이 end-effector의 위치와 접근 방향에 관계되기 때문에 arm matrix T_6 가 매우 복잡하며 따라서 대수적 방법으로는 각각의 joint angle의 함수로 표현되므로 6개의 해를 유일하게 구할 수 없고 어느 하나의 joint angle 값을 알아야 한다. 식(33) ~ 식(37)은 θ_4 가 주어질 때 나머지 5개의 joint angle을 얻을 수 있는 대수적 방법의 결과이다.

$$\theta_1 = \tan^{-1} \left[\frac{\pm \sqrt{(Px-25Ax)^2 + (Py-25Ay)^2 - (10c_4+3)^2}}{10c_4 + 3} \right] - \tan^{-1} \left[\frac{Px-25Ax}{Py-25Ay} \right] \quad (33)$$

$$\theta_2 = \tan^{-1} \left[\frac{\pm \sqrt{A^2 + B^2 - C^2}}{C} \right] - \tan^{-1} \left[\frac{B}{A} \right] \quad (34)$$

여기서 $A = (Px-25Ax)c_1 + (Py-25Ay) - 20$

$$B = Pz - 25Az$$

$$C = [A^2 + B^2 - (10s_4+13)^2 + 60^2 - 55^2] / 120$$

$$\theta_3 = \tan^{-1} \left[\frac{\pm \sqrt{(10s_4+13)^2 + 55^2 - D^2}}{D} \right] - \tan^{-1} \left[\frac{55}{10s_4+13} \right] \quad (35)$$

여기서 $D = Ac_2 - Bs_2 - 60$

$$\theta_5 = \tan^{-1} \left[\frac{[(Axc_1+Ays_1)c_{23}-Azs_{23}]c_4 + (Axs_1-Ayc_1)s_4}{(Axc_1+Ays_1)s_{23} + Azc_{23}} \right] \quad (36)$$

$$\theta_6 = \tan^{-1} \left[\frac{[(Nxc_1+Nys_1)c_{23}-Nzs_{23}]s_4 - (Nxs_1-Nyc_1)s_4}{[(Oxc_1+Oys_1)c_{23}-Ozs_{23}]s_4 - (Oxs_1-Oyc_1)c_4} \right] \quad (37)$$

이상과 같이 FANUC robot은 구조적으로 매우 복잡하므로 일반적인 inverse kinematics 해법으로는 그 해를 찾기가 불가능하다. 기존의 반복법을 사용하여 해를 구해낼 수는 있으나 시간이 너무 오래 걸려서 실용적이지 못하므로 이분법을 이용하여 θ_4 를 구하고 θ_4 를 제외한 나머지 $\theta_1 \sim \theta_6$ 의 값들은 대수적 방법을 이용하여 구한 식(33) ~ 식(37)에 의하여 구하였다.

3.1 이분법[7]

이분법이란 중간값 정리를 토대로 한 방법이다. 구간 $[a, b]$ 에서 연속인 함수 $f(x)$ 가 주어지고 $f(a)$ 와 $f(b)$ 가 반대 부호를 갖는다고 하면 중간값 정리에 의해 $f(p) = 0$ 를 만족하는 p 가 a 와 b 사이에 존재한다. 이분법은 그림 4와 같이 구간 $[a, b]$ 를 p 점을 포함하게 계속 이등분하는 것이다. 먼저 $a_1 = a$, $b_1 = b$ 라 하고 p_1 을 $[a, b]$ 의 중점이라 하자. 만일 $f(p_1) = 0$ 이면 $p = p_1$ 이고 $f(p_1) \neq 0$ 이면 $f(p_1)$ 은 $f(a_1)$ 이나 $f(b_1)$ 중 하나와 같은 부호를 갖는다. 만일 $f(p_1)$ 이 $f(a_1)$ 과 같은 부호를 갖는다면 p 는 p_1 과 b_1 사이에 있게 되므로 $a_2 = p_1$, $b_2 = b_1$ 으로 한다. 만일 $f(p_1)$ 이 $f(b_1)$ 과 같은 부호를 갖는다면 p 는 a_1 과 p_1 사이에 있게 되므로 $a_2 = a_1$, $b_2 = p_1$ 이 된다. 이제 구간 $[a_2, b_2]$ 에 앞의 과정을 반복 적용한다.

3.2 이분법을 이용한 해법

본 논문에서는 θ_4 의 전 구간($-190^\circ \sim 190^\circ$)을 8.4° 씩

나는 후 첫 구간의 양 끝값 $\theta_4 = -190^\circ$, $\theta_4 = -181.6^\circ$ 를 이용하여 각각의 θ_4 에 대한 $\theta_1 \sim \theta_6$ 를 식(33)~식(37)에 의해 구한 후, 이 값과 input T_6 의 각 term(N, O, A, P)의 값을 이용하여 이분법을 수행한다. 첫번째 구간에서 근을 찾은 후에는 다음 구간으로 넘어가서 근을 찾도록 하여 input T_6 에 대한 모든 inverse kinematics의 해를 구하도록 하였다.

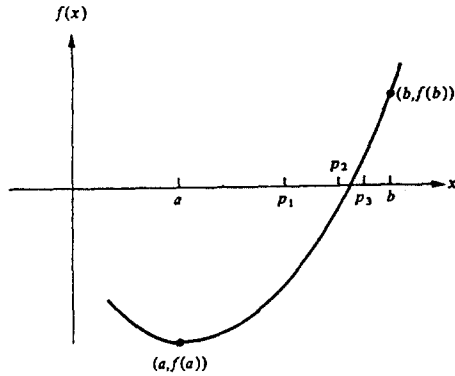


그림 4. 이분법

먼저 식(28)의 U_2 matrix에서 식(38)~식(50)을 얻는다.

$$U_2 = A^{-1} * T_6 = \begin{bmatrix} c1 & 0 & -s1 & 20c1 \\ s1 & 0 & c1 & 20s1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} Nx & Ox & Ax & Px \\ Ny & Oy & Ay & Py \\ Nz & Oz & Az & Pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (38)$$

$$= \begin{bmatrix} Nx_1 & Ox_1 & Ax_1 & Px_1 \\ Ny_1 & Oy_1 & Ay_1 & Py_1 \\ Nz_1 & Oz_1 & Az_1 & Pz_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

여기서

$$Nx_1 = c1Nx + Nys1 \quad (39)$$

$$Ox_1 = c1Ox + Oys1 \quad (40)$$

$$Ax_1 = Axc1 + Ays1 \quad (41)$$

$$Px_1 = -20 + c1Px + Pys1 \quad (42)$$

$$Ny_1 = -Nz \quad (43)$$

$$Oy_1 = -Oz \quad (44)$$

$$Ay_1 = -Az \quad (45)$$

$$Py_1 = -Pz \quad (46)$$

$$Nz_1 = c1Ny - Nxs1 \quad (47)$$

$$Oz_1 = c1Oy - Oxs1 \quad (48)$$

$$Az_1 = Ayc1 - Axs1 \quad (49)$$

$$Pz_1 = c1Py - Pxs1 \quad (50)$$

또한 식(14)에서 다시 U_2 를 구할 수 있다.

$$U_2 = A_2 * A_3 * A_4 * A_5 * A_6$$

$$= \begin{bmatrix} Nx_1 & Ox_1 & Ax_1 & Px_1 \\ Ny_1 & Oy_1 & Ay_1 & Py_1 \\ Nz_1 & Oz_1 & Az_1 & Pz_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (51)$$

$$Nx_1 = s4s6c23 + c6(c4c5c23 + s5(-s23)) \quad (52)$$

$$Ox_1 = c6s4c23 - s6(c4c5c23 + s5(-s23)) \quad (53)$$

$$Ax_1 = c4s5c23 + c5s23 \quad (54)$$

$$Px_1 = 60c2 + (13 + 10s4 + 25c4c5)c23 + (-55 + 25c5)s23 \quad (55)$$

$$Ny_1 = (s4s6 + c6c4c5)s23 + c6s5c23 \quad (56)$$

$$Oy_1 = (c6s4 - s6c4c5)s23 - s6s5c23 \quad (57)$$

$$Ay_1 = -c5c23 + c4s5s23 \quad (58)$$

$$Py_1 = 60s2 + (13 + 10s4 + 25c4c5)s23 + (55 - 25c5)c23 \quad (59)$$

$$Nz_1 = -c5c6s4 + c4s6 \quad (60)$$

$$Oz_1 = c4c6 + c5s4s6 \quad (61)$$

$$Az_1 = -s4s5 \quad (62)$$

$$Pz_1 = 3 + 10c4 - 25s4s5 \quad (63)$$

이 두 U_2 matrix의 각 term은 서로 같은 값을 가지므로 식 (64)~식(75)이 모두 0(zero)임을 알 수 있다.

$$Nx = c23(c4c5c6 + s4s6) - s23s5c6 - (Nxc1 + Nys1) \quad (64)$$

$$Ny = s23(c4c5c6 + s4s6) + c23s5c6 - (-Nz) \quad (65)$$

$$Nz = -s4c5c6 + c4c6 + (Nxs1 - Nyc1) \quad (66)$$

$$Ox = -c23(c4c5s6 - s4c6) + s23s5s6 - (Oxc1 + Nys1) \quad (67)$$

$$Oy = -s23(c4c5s6 - s4c6) - c23s5s6 - (-Oz) \quad (68)$$

$$Oz = s4c5s6 + c4c6 + (Oxs1 - Oyc1) \quad (69)$$

$$Ax = c23c4s5 + s23c5 - (Axc1 + Ays1) \quad (70)$$

$$Ay = s23c4s5 - c23c5 - (-Az) \quad (71)$$

$$Az = -s4s5 + (Axs1 - Ayc1) \quad (72)$$

$$Px = 25(c23c4s5 + s23c5) + c23(10s4 + 13) - 55s23 + 60c2 - (Pxc1 + Pys1 - 20) \quad (73)$$

$$P_y = 25(s23c4s5-c23c5)+c23(10s4+13) +55c23+60s2-(-P_z) \quad (74)$$

$$P_z = -25s4s5+10c4+3+(Pxs1-Pycl) \quad (75)$$

선택된 θ_4 에 의해 구해진 $\theta_1 \sim \theta_6$ 의 값과 input T_6 의 각 term(N, O, A, P)의 값들을 식(64)~식(75)에 대입하여 그 결과가 (+)인지 (-)인지 구별한다. 구간의 양 끝점에서의 부호가 다르면 그 구간 안에서 이분법을 수행하고 부호가 같으면 다음 구간으로 넘어가서 위의 과정을 반복하여 이를 θ_4 의 전 구간에서 수행한다. 이렇게 찾아진 θ 값에 의한 T_6 matrix와 input T_6 matrix와의 오차가 0.01 이하일 때의 $\theta_1 \sim \theta_6$ 를 input T_6 matrix에 대한 inverse kinematics의 해로 결정한다.

III. 실험 결과 및 결론

FANUC robot의 kinematics와 inverse kinematics를 θ_4 의 이분법에 의해 구해보았다. Kinematics의 수행결과는 표 2에 나타내었고 inverse kinematics의 수행결과는 표 3에 나타내었다. 표 3에서 알 수 있듯이 해를 구하는 평균시간이 3.3초가 걸렸다. 이전의 반복법에 의한 계산보다는 시간이 많이 단축되긴 하였으나 robot을 실시간으로 운용하기에는 아직도 미흡하다. 좀 더 빠른 처리시간을 위해 θ_4 의 전구간에서의 이분법이 아닌 일부구간에서의 이분법이나 θ_1 을 중심으로 inverse kinematics를 수행하는 방법에 대한 연구가 진행되어야 할 것이다.

표 2 kinematics의 수행 결과

입력 각도 [단위 °]			출력 각도 T_6 [단위 cm]					
θ_1	θ_2	θ_3	N_x	O_x	A_x	P_x		
θ_4	θ_5	θ_6	N_y	O_y	A_y	P_y		
			N_z	O_z	A_z	P_z		
0.00	0.00	0.00	-0.00000	0.00000	1.00000	99.99999		
0.00	0.00	0.00	0.00000	1.00000	0.00000	13.00000		
0.00	0.00	0.00	-1.00000	0.00000	-0.00000	73.00000		
1.80	22.52	-35.51	-0.73682	0.56613	0.36955	101.16490		
30.71	-36.31	-57.91	0.48687	0.82359	-0.29094	7.21863		
			-0.46907	-0.03444	-0.88248	7.84125		
43.83	1.94	-22.57	-0.02305	0.95868	-0.28353	44.27524		
-47.07	-60.88	-104.91	0.76313	0.20009	0.61447	78.26968		
			0.64582	-0.20221	-0.73622	39.21586		
56.22	24.55	-35.90	0.92909	-0.27637	0.24577	66.56961		
-158.34	57.41	-83.65	-0.23283	-0.95339	-0.19190	74.21296		
			0.28735	0.12107	-0.95014	12.09231		
92.93	16.77	-48.49	-0.05313	0.11194	0.99229	16.68515		
-87.27	86.80	-108.43	0.49899	-0.85776	0.12348	93.76852		
			0.86497	0.50170	-0.01028	31.24058		
-5.20	16.69	-57.89	-0.85314	0.51609	-0.07614	77.26419		
-5.68	-36.95	-21.14	0.51207	0.85634	0.06666	7.46632		
			0.09961	0.01788	-0.99486	-6.54969		
-46.91	28.42	-0.17	0.86364	-0.42181	0.27601	87.14022		
-4.79	61.33	-75.26	0.50238	0.76535	-0.40230	-76.85538		
			-0.04154	0.48611	0.87290	88.26338		

-85.25	23.67	-30.25	0.74834	0.26139	0.60962	24.79461
0.65854	-0.40266	-0.63574	-108.56477			
96.62	33.96	-172.12	0.07929	0.87723	-0.47347	18.05704
112.45	9.62	-2.56	-0.10883	-0.11348	-0.98756	-59.19853
-0.97920	-0.15889	0.12616	80.98143			
94.74	60.17	-170.23	-0.17123	0.98075	-0.09383	57.38503

표 3. Inverse kinematics의 수행 결과

입력 T_6 [단위 cm]		출력 각도 [단위 °]			시간 error		
0.00000	0.00000	1.00000	99.99999	0.00	0.00	0.00	3
0.00000	1.00000	0.00000	12.99999	0.00	0.00	0.00	9.5E-11
1.00000	0.00000	0.00000	72.99999	1.80	22.52	-35.51	3
-0.73682	0.56613	0.36955	101.16490	30.71	-36.31	-57.91	2.14E-4
0.48687	0.82359	-0.29094	7.21863	43.83	1.94	-22.57	4
-0.46907	-0.03444	-0.88248	7.84125	-158.34	-60.88	-104.91	2.95E-4
0.92909	-0.27637	0.24577	66.56961	45.46	25.64	-28.83	1
-0.23283	-0.95339	-0.19190	74.21296	20.81	24.55	-35.90	7.43E-4
0.28735	0.12107	-0.95014	12.09231	56.22	24.55	-35.90	1
-0.05313	0.11194	0.99229	16.68515	-158.34	57.41	-83.65	9.61E-4
0.49899	-0.85776	0.12348	93.76852	93.37	21.15	-29.49	5
0.86497	0.50170	-0.01028	31.24058	91.32	-86.47	96.90	7.72E-4
-0.85314	0.51609	-0.07614	77.26419	92.93	16.77	-48.49	5
0.51207	0.85634	0.06666	7.46632	-87.27	86.80	-108.43	5.94E-4
0.09961	0.01788	-0.99486	-6.54969	-5.20	16.69	-57.89	0
0.86364	-0.42181	0.27601	87.14022	-5.68	-36.95	-21.14	7.24E-4
0.50238	0.76535	-0.40230	-76.85538	0.86497	0.50170	-0.01028	2
-0.04154	0.48611	0.87290	88.26338	-4.79	61.33	-75.26	1.41E-4
0.74834	0.26139	0.60962	24.79460	-85.25	23.67	-30.25	3
0.65854	-0.40266	-0.63574	-108.56477	96.62	33.96	-172.12	4.17E-4
0.07929	0.87723	-0.47347	18.05704	-83.69	19.60	-48.73	3
-0.10883	-0.11348	-0.98756	-59.19853	-117.67	-35.23	40.73	4.93E-4
-0.97920	-0.15889	0.12616	80.98143	112.45	9.62	-2.56	2
-0.17123	0.98075	-0.09383	57.38501	94.74	60.17	-170.23	2.17E-4
				112.71	5.29	-22.64	2
				-96.94	-60.31	32.94	7.35E-4
평균 시간 3.3 초				평균 오차 5.31E-4			

참 고 문 헌

- [1] "FANUC ROBOT Arc Mate User's Manual", Fanuc Ltd., 1987
- [2] C. S. G. Lee, R. C. Gonzalez, and K. S. Fu, "Tutorial on Robotics" IEEE Computer Society, 1986.
- [2] C. S. G. Lee, "Robot arm kinematics, dynamics, and control", vol. 15, no. 12, pp. 62-80, Dec. 1982.
- [4] R. P. Paul, B. Shimano, and G. E. Mayer, "kinematics control equations for simple manipulators", IEEE Trans. on System, Man, and Cybernetics, vol. SMC-11, no. 6, pp. 449-455, June 1981.
- [5] M. Vukobratovic and M. Kircanski, "Kinematics and trajectory synthesis of manipulation robots", Scientific Fundamentals of Robotics, vol. 3, pp. 138-155, 1986.
- [6] C. S. G. Lee, and M. Ziegler, "Geometric approach in solving inverse kinematics of PUMA robots", IEEE Trans. on Aerospace and Electronic System, vol. AES-20, no. 6, pp. 695-706, Nov. 1984.
- [7] R. L. Burden, "Numerical Analysis", 생능 출판사, pp. 28-34, 1987.12.