

Least Square를 이용한 수직다관절 Manipulator의 새로운 원호 경로 보간 방법

정원지*, 이춘만⁺, 김대영⁺⁺, 서영교⁺⁺⁺, 홍형표⁺⁺⁺

(논문접수일 2002. 12. 31, 심사완료일 2003. 6. 27)

A New Circular Curve Fitting of Articulated Manipulators Using Least Squares

W. J. Chung*, C. M. Lee⁺, D. Y. Kim⁺⁺, Y. K. Seo⁺⁺⁺, H. P. Hong⁺⁺⁺

Abstract

This paper presents a new circular curve fitting approach of articulated manipulators, based on least square. The approach aims at gaining the interpolation of circle from n data points, under the condition that the fitted circle should pass both a starting point and an ending point. First a spherical fitting should be performed, using least squares. Then the circular curve fitting can be resulted from the intersection of the fitted sphere and the plane obtained from 3 points, *i. e.*, a starting point, an ending point, and the center of a sphere. The proposed algorithms are shown to be efficient by using MATLAB-based simulation.

Key Words : Circular Interpolation(원호보간), Articulated Manipulator(6축 수직다관절 로봇), Least Square(최소 자승법)

1. 서론

오늘날 공작기계나 6축 수직다관절 로봇이 자동화에 기여하는 바가 크다. 특히 도장이나 용접 등에서 로봇의 능력은 더욱 돋보인다. 로봇을 이용하여 용접(welding)이나, 도장(painting) 작업을 수행하거나 또는, 로봇의 이송 경로에

장애물이 존재하는 경우와 같이 제한된 작업 환경에서 주어진 경로를 따라가야 하는 경우, 로봇의 경로는 크게 PTP(Point to Point Path)와 CP(Continuous Path)의 두 가지 방법에 의해 결정된다⁽¹⁾.

로봇이 용접이나 도장 작업을 하고자 하는 경로가 원일 경우에는 보통 CP를 사용한다. 하지만 몇 개의 공간상의

* 주저자, 창원대학교 기계설계공학과 (wjchung@sarim.changwon.ac.kr)

주소: 641-773 경남 창원시 사림동 9번지

+ 창원대학교 기계설계공학과

++ 창원대학교 기계설계공학과 대학원

+++ 두산메카텍(주) 연구개발부

점들만을 이용하여 연속성을 유지하는 원을 찾기란 쉽지가 않다. 왜냐하면 2차원 평면상이 아닌 3차원 공간상에 존재하는 n 개의 점 데이터(작업자가 입력하는 점 데이터)들을 획득하여 근사(interpolation)나 보간을 사용해서 원을 찾아야 하기 때문이다.

공작기계에는 적용 가능한 Biarc를 이용한 원호보간법이 있다⁽²⁾. 또한 로봇에도 Biarc를 이용하여 적용 가능한 원호보간법이 있다. 하지만, 이것은 단지 세 점만을 보간할 뿐이다. 따라서 세 점 이상 n 개의 점 데이터들을 이용해서 원호보간 뿐만 아니라 그 점들에 가장 근접한 원까지 구현 가능한 알고리즘을 제시한다. 원호보간은 원을 구한 후 시점과 종점이 이루는 각도만큼만 취하면 된다. 따라서, 본 논문에서는 원의 구현을 중점적으로 다룬다.

작업자는 입력한 점 데이터들이 작업하고자 하는 원상에 존재하는지 확인할 수 없다. 따라서, 입력하는 점 데이터들 중에서 로봇 작업의 특성상 시점과 종점은 반드시 작업하고자 하는 원상에 존재하고, 나머지 점들은 오차를 지닌다고 가정한다. 즉, 연속된 동작을 하기 위해서는 구하고자 하는 원의 시점과 종점을 반드시 지나야 하기 때문이다.

이와 같은 가정 하에 n 개의 점 데이터가 주어지면 시점과 종점을 지나면서 나머지 점 데이터들에 대해서는 최소의 오차를 지니는 원을 구현한다.

2. 근접한 구(Sphere)의 구현

2.1 교시한 점 데이터들에 가장 근접한 구

주어진 점들에 가장 근접한 구를 찾기 위해서 다음과 같은 구속조건을 갖춘다. 첫째, 시점과 종점은 반드시 구하고자 하는 원상에 존재한다.(물론 이 점들이 일치할 수도 있지만, 일치하지 않음을 원칙으로 한다.) 둘째, 시점과 종점의 사이 $n-2$ 개의 점 데이터들은 원상에 존재하지 않아도 되지만, 구하고자 하는 원으로부터 멀리 떨어진 점 데이터들의 사용은 지양한다. 이 두 가지 조건이 만족되면 먼저 최소자승법(Least squares)⁽³⁾을 이용하여 가장 작은 오차를 포함한 구를 찾는다.

2.1.1 구의 일반식

구의 일반식은 다음과 같다.

$$x^2 + y^2 + z^2 + ax + by + cz + d = 0 \quad (1)$$

n 개의 점 data $(x_1, y_1, z_1), (x_2, y_2, z_2), \dots, (x_n, y_n, z_n)$ 가 있을 때, 시점 (x_1, y_1, z_1) 와 종점 (x_n, y_n, z_n) 은 반드시 구상에 있어야 하므로, 아래의 조건식이 성립해야 한다.

$$x_1^2 + y_1^2 + z_1^2 + ax_1 + by_1 + cz_1 + d = 0 \quad (2)$$

$$x_n^2 + y_n^2 + z_n^2 + ax_n + by_n + cz_n + d = 0 \quad (3)$$

$(n-2)$ 개의 점 data, 즉 $(x_2, y_2, z_2), (x_3, y_3, z_3), (x_{n-1}, y_{n-1}, z_{n-1})$ 에 대해서는 최소자승법을 적용한다. 이 경우에는 식 (1)이 오차 함수(Error Function)가 된다.

$$\sum_{i=2}^{n-1} E^2 = \sum_{i=2}^{n-1} (x_i^2 + y_i^2 + z_i^2 + ax_i + by_i + cz_i + d)^2 \quad (4)$$

식 (4)를 계수 a, b, c, d 에 대해 편미분을 해서 Zero로 두면 다음과 같다.

$$\frac{\partial \sum_{i=2}^{n-1} E^2}{\partial a} = 2 \sum_{i=2}^{n-1} (x_i^2 + y_i^2 + z_i^2 + ax_i + by_i + cz_i + d)x_i = 0 \quad (5)$$

$$\frac{\partial \sum_{i=2}^{n-1} E^2}{\partial b} = 2 \sum_{i=2}^{n-1} (x_i^2 + y_i^2 + z_i^2 + ax_i + by_i + cz_i + d)y_i = 0 \quad (6)$$

$$\frac{\partial \sum_{i=2}^{n-1} E^2}{\partial c} = 2 \sum_{i=2}^{n-1} (x_i^2 + y_i^2 + z_i^2 + ax_i + by_i + cz_i + d)z_i = 0 \quad (7)$$

$$\frac{\partial \sum_{i=2}^{n-1} E^2}{\partial d} = 2 \sum_{i=2}^{n-1} (x_i^2 + y_i^2 + z_i^2 + ax_i + by_i + cz_i + d) = 0 \quad (8)$$

식 (2), (3), (5), (6), (7), (8)을 Matrix Form으로 다시 정리하면,

$$A \cdot x = b \quad (9)$$

단,

$$A = \begin{bmatrix} x_1 & y_1 & z_1 & 1 \\ \sum_{i=2}^n x_i^2 & \sum_{i=2}^n x_i y_i & \sum_{i=2}^n z_i x_i & \sum_{i=2}^n x_i \\ \sum_{i=2}^n x_i y_i & \sum_{i=2}^n y_i^2 & \sum_{i=2}^n y_i z_i & \sum_{i=2}^n y_i \\ \sum_{i=2}^n z_i x_i & \sum_{i=2}^n y_i z_i & \sum_{i=2}^n z_i^2 & \sum_{i=2}^n z_i \\ \sum_{i=2}^n x_i & \sum_{i=2}^n y_i & \sum_{i=2}^n z_i & n-2 \\ x_n & y_n & z_n & 1 \end{bmatrix} \in R^{6 \times 4} \quad (10)$$

$$x = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \in R^4 \quad (11)$$

$$b = \begin{bmatrix} -(x_1^2 + y_1^2 + z_1^2) \\ -\sum_{i=2}^n (x_i^3 + x_i y_i^2 + x_i z_i^2) \\ -\sum_{i=2}^n (x_i^2 y_i + y_i^3 + y_i z_i^2) \\ -\sum_{i=2}^n (x_i^2 z_i + y_i^2 z_i + z_i^3) \\ -\sum_{i=2}^n (x_i^2 + y_i^2 + z_i^2) \\ -(x_n^2 + y_n^2 + z_n^2) \end{bmatrix} \in R^6 \quad (12)$$

이 된다.

2.1.2 Pseudoinverse⁽⁴⁾

식 (9)를 살펴보면, 식은 6개이고 미지수는 4개이다. (inconsistent) A 의 계수(rank)가 4이면, 식 (9)는 해(解)를 가지며 해(解)의 개수는 많아야 1개이다.

일반적으로,

$$A \cdot x = b \quad (13)$$

$A \in R^{m \times n}$, $x \in R^n$, $b \in R^m$ 에서 $\text{rank } A = n$ ($m \geq n$) 이라고 하자.(식 (9)에서 $m=6$, $n=4$) b 는 A 의 열 공간(column space)에 존재하지 않으므로,

$$E = \|Ax - b\| \quad (14)$$

식 (14)를 최소로 하는 해(解) \bar{x} 를 구하는 최소자승법 문제가 된다. 이때, \bar{x} 를 최소자승법의 해(解)라 한다.

(1) E 를 최소화하는 최소자승법 문제는 열 공간에서 b 에 가장 가까운 점 \bar{b} 즉, $\bar{b} = A\bar{x}$ 를 찾는 것이다. 기하학적으로 \bar{b} 는 b 의 열 공간 상(上)의 투영(projection)이 된다. 열 공간에 수직인 벡터는 왼쪽 영공간(left nullspace)에 존재하므로,

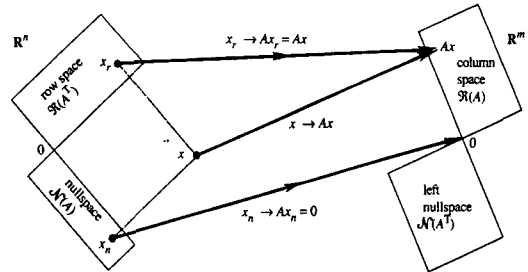


Fig. 1 The action of a matrix A

$$A^T (b - \bar{b}) = 0 \quad (15)$$

왼쪽 영공간은 Fig. 1과 같다⁽¹⁾.

식 (15)를 다시 쓰면,

$$A^T (b - A\bar{x}) = 0 \quad (16)$$

이다. 즉,

$$A^T A \bar{x} = A^T b \quad (17)$$

(2) (1)방법을 좀 더 쉽게 다음과 같이 접근해 볼 수 있다.

오차 벡터(Error vector) $(b - A\bar{x})$ 는 A 의 모든 열 벡터(column vector)와 직교하므로,

$$\begin{aligned} a_1^T \cdot (b - A\bar{x}) &= 0 \\ a_2^T \cdot (b - A\bar{x}) &= 0 \\ &\vdots \\ a_n^T \cdot (b - A\bar{x}) &= 0 \end{aligned} \quad (18)$$

식 (18)을 다시 정리하면,

$$\begin{bmatrix} - & a_1 & - \\ - & a_2 & - \\ & \vdots & \\ - & a_n & - \end{bmatrix} \begin{bmatrix} | \\ b - A\bar{x} \\ | \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (19)$$

즉,

$$A^T (b - A\bar{x}) = 0 \quad (20)$$

따라서,

$$\underline{A}^T \underline{A} \bar{x} = \underline{A}^T b \quad (21)$$

이 된다.

$\text{rank}(\underline{A}^T \underline{A}) = n$ 이고, $\underline{A}^T \underline{A} \in R^{4 \times 4}$ 이므로 $\underline{A}^T \underline{A}$ 는 역행렬을 구할 수 있다. 따라서,

$$\begin{aligned} \bar{x} &= (\underline{A}^T \underline{A})^{-1} \underline{A}^T b \\ &= \underline{A}^+ b \end{aligned} \quad (22)$$

where,

$$\begin{aligned} \underline{A}^+ &= (\underline{A}^T \underline{A})^{-1} \underline{A}^T; \text{ Left Inverse} \\ (\underline{A}^+ \underline{A} &= I_{n \times n}) \text{ 이다.} \end{aligned}$$

왼쪽 역(Left Inverse)은 Pseudoinverse의 한 종류이다. 참고로, 오른쪽 역(Right Inverse)은 다음과 같다.

$$\begin{aligned} \underline{A}^+ &= \underline{A}^T (\underline{A} \underline{A}^T)^{-1}; \text{ Right Inverse} \\ \text{Rank } \underline{A} &= m \quad (m \leq n) \quad (\underline{A} \underline{A}^+ = I_{m \times m}) \end{aligned}$$

2.1.3 Equation of Spherical

식 (9)를 Pseudoinverse를 이용해서 평면 x 의 값을 구할 수 있다.

$$x = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \quad (24)$$

식 (1)에 식 (24)에서 구한 a, b, c, d 값을 대입하면 구의 방정식을 얻을 수 있고, 식 (1)을 다시 정리하면 다음과 같다.

$$\begin{aligned} \left(x + \frac{a}{2}\right)^2 + \left(y + \frac{b}{2}\right)^2 + \left(z + \frac{c}{2}\right)^2 \\ = \frac{a^2 + b^2 + c^2}{4} - d \end{aligned} \quad (25)$$

이 때 구의 중심과 반지름은 다음과 같이 된다.

$$\text{구의 중심} : \left(-\frac{a}{2}, -\frac{b}{2}, -\frac{c}{2}\right) \quad (26)$$

$$\text{구의 반지름} : \sqrt{\frac{a^2 + b^2 + c^2}{4} - d} \quad (27)$$

3. 원호보간 (Circular Interpolation)

3.1 원호보간

교시한 시점, 종점 및 앞서 구한 구의 중심을 $(x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3)$ 라 두고 이 세 점을 평면의 방정식 $ax + by + cz + d = 0$ 에 대입한다. (단, 종점 $(x_2, y_2, z_2) =$ 점 데이터 (x_n, y_n, z_n) 이고, 구의 중심 $(x_3, y_3, z_3) = \left(-\frac{a}{2}, -\frac{b}{2}, -\frac{c}{2}\right)$ 이다.)

$$ax_1 + by_1 + cz_1 + d = 0 \quad (28)$$

$$ax_2 + by_2 + cz_2 + d = 0 \quad (29)$$

$$ax_3 + by_3 + cz_3 + d = 0 \quad (30)$$

b, c, d 를 a 에 대해 정리하면 다음과 같다.

$$b = \frac{\{(x_1 - x_2)(z_3 - z_2) - (x_3 - x_2)(z_1 - z_2)\}}{\{(y_3 - y_2)(z_1 - z_2) - (y_1 - y_2)(z_3 - z_2)\}} \times a \quad (31)$$

$$c = \frac{\{(x_1 - x_2)(y_3 - y_2) - (x_3 - x_2)(y_1 - y_2)\}}{\{(z_3 - z_2)(y_1 - y_2) - (z_1 - z_2)(y_3 - y_2)\}} \times a \quad (32)$$

$$\begin{aligned} d = -a \left[x_1 + \frac{y_1 \{(x_1 - x_2)(z_3 - z_2) - (x_3 - x_2)(z_1 - z_2)\}}{\{(y_3 - y_2)(z_1 - z_2) - (y_1 - y_2)(z_3 - z_2)\}} \right. \\ \left. + \frac{z_1 \{(x_1 - x_2)(y_3 - y_2) - (x_3 - x_2)(y_1 - y_2)\}}{\{(z_3 - z_2)(y_1 - y_2) - (z_1 - z_2)(y_3 - y_2)\}} \right] \quad (33) \end{aligned}$$

평면의 방정식 $ax + by + cz + d = 0$ 에 구한 b, c, d 를 대입하고, a 로 나누면

$$\begin{aligned} x + \left[\frac{\{(x_1 - x_2)(z_3 - z_2) - (x_3 - x_2)(z_1 - z_2)\}}{\{(y_3 - y_2)(z_1 - z_2) - (y_1 - y_2)(z_3 - z_2)\}} \right] y \\ + \left[\frac{\{(x_1 - x_2)(y_3 - y_2) - (x_3 - x_2)(y_1 - y_2)\}}{\{(z_3 - z_2)(y_1 - y_2) - (z_1 - z_2)(y_3 - y_2)\}} \right] z \\ \left[x_1 + \frac{y_1 \{(x_1 - x_2)(z_3 - z_2) - (x_3 - x_2)(z_1 - z_2)\}}{\{(y_3 - y_2)(z_1 - z_2) - (y_1 - y_2)(z_3 - z_2)\}} \right. \\ \left. + \frac{z_1 \{(x_1 - x_2)(y_3 - y_2) - (x_3 - x_2)(y_1 - y_2)\}}{\{(z_3 - z_2)(y_1 - y_2) - (z_1 - z_2)(y_3 - y_2)\}} \right] = 0 \end{aligned} \quad (34)$$

과 같이 된다. 이식을 매개변수를 사용하여 시뮬레이션 하였다.

즉, 앞에서 구한 구의 방정식은,

$$(x - x_c)^2 + (y - y_c)^2 + (z - z_c)^2 = R^2 \quad (35)$$

이고, 평면의 방정식은,

$$x + by + cz + d = 0 \quad (36)$$

이다.

여기에 매개변수를 도입한다.

$$x = t \quad (37)$$

$$x + by + cz + d = 0 \rightarrow t + by + cz + d = 0$$

$$y = -\left(\frac{t + cz + d}{b}\right) \quad (38)$$

식 (37), (38)을 식 (35)에 대입하고, 근의 공식에 의해 'z'를 구하면 식 (39)와 같다.

$$z = \frac{(z_c - pq) \pm \sqrt{(pq - z_c)^2 - (p^2 + 1)k}}{(p^2 + 1)} \quad (39)$$

$$\text{단, } p = \left(\frac{c}{b}\right), \quad q = \left\{\left(\frac{t+d}{b}\right) + y_c\right\},$$

$$k = q^2 + z_c^2 + (t - x_c)^2 - R^2$$

이다.

식 (39)를 식 (38)에 대입하면 식 (40)과 같이 된다.

$$\begin{aligned} y &= -\left(\frac{t+d}{b}\right) - \left(\frac{c}{b}\right)z \\ &= -\left(\frac{t+d}{b}\right) - p \cdot \left[\frac{(z_c - pq) \pm \sqrt{(pq - z_c)^2 - (p^2 + 1)k}}{(p^2 + 1)} \right] \end{aligned} \quad (40)$$

4. Simulation

4.1 Simulation output I

다섯 개의 점일 경우 매트랩 프로그램으로 시뮬레이션 한 결과이다.

(2.5,6.5,4.5),(2.7,6.3,4.7),(2.9,6.1,4.8),(3,5.9,5),(3.2, 5.8, 5.1)의 5개의 점 데이터를 입력한 경우 Fig. 2와 같은 원을 얻을 수 있고, Fig. 3과 같이 원호보간이 가능하다.

4.2 Simulation output II

두산메카텍(주)에서 실제 사용되어지는 일곱개의 점일 경우 매트랩 프로그램으로 시뮬레이션 한 결과이다. 마찬가지로 Fig. 4와 같은 원을 얻을 수 있고, Fig. 5와 같이 원호보간이 가능하다.

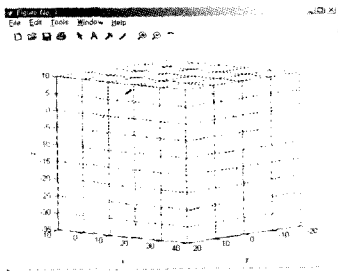


Fig. 2 Output I-1

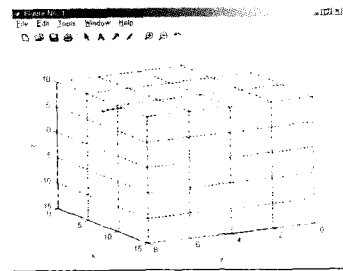


Fig. 3 Output I-2

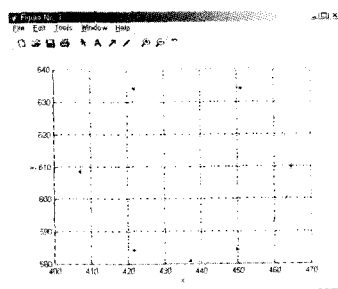


Fig. 4 Output II-1

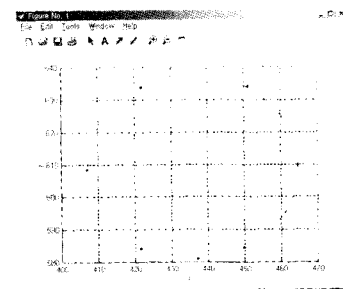


Fig. 5 Output II-2

5. 결론

기존의 원호보간 방법은 공작기계에 사용되는 Biarc를 이용한 원호보간 방법을 사용하였다. 연속된 로봇 작업 시 시점과 종점을 반드시 지나야 하는 특성과 사용자가 정확한 점 데이터들을 입력할 수 없다는 점등을 고려해야 한다. 따라서 본 논문에서는 3 점만을 이용하여 원호보간하고 그것들을 연결하는 기존의 방법 대신 n 개의 점 데이터들을 이용하여 원호보간이 가능하도록 하였다.

먼저 공간상의 임의의 n 개의 점 데이터들에 대하여, 시점과 종점을 지나면서 나머지 모든 점 데이터들에 가장 근접한 구를 구현하였다. 다음으로 시점, 종점, 구의 중심(원의 중심과 일치)을 이용하여 평면의 방정식을 구하고, 구한 평면의 방정식과 구의 방정식을 이용하여 가장 근접한 원을 찾았다. 부연하면, 구를 구의 중심을 지나는 평면으로 자르면 그 평면상에는 원이 나타나는 것과 같다. 원호보간은 결국 구한 원의 일부분이기 때문에 원에서 시점과 종점이 이루는 각도만큼만 나타내면 되었다.

시뮬레이션을 통하여 본 논문에서 제안한 방법의 타당성을 검증하였다. 이 방법은 6축 수직다관절(articulated manipulator) 뿐만 아니라 6축 이하의 로봇이나 공작기계에도 응

용이 가능할 것으로 기대된다.

후 기

이 논문은 2002년도 창원대학교 연구비 및 두산메카텍(주)의 지원에 의한 것이다. 본 연구는 한국과학재단 지정 창원대학교 공작기계기술연구센터(RRC)의 지원에 의한 것이다.

참 고 문 헌

- (1) Craig, J. J., 1992, *Introduction to Robotics Mechanics and Control*, Addison-Wesley, Inc.
- (2) Shon, T. Y., Yang, M. Y. and Cho, H. D., 1997, "An Improved NC-code Generation Method for Circular Interpolation", *KSPE*, Vol. 14, No. 11, pp. 77~83.
- (3) Kreyszig, E., 1993, *Advanced Engineering Mathematics*, John Wiley & Son, Inc.
- (4) Strang, G., 1989, *Linear Algebra and Its Applications*, Harcourt Brace Jovanovich.