

분리된 조작도를 이용한 여유자유도 로봇의 최적 자세에 관한 연구

Study on the Optimal Posture for Redundant Robot Manipulators Based on Decomposed Manipulability

이 지 흥, 원 경 태
(Jihong Lee, and Kyoung Tae Won)

Abstract : The conventional robot manipulability is decomposed into linear manipulability and angular manipulability so that they may be analysed and visualized in easy way even in the case of 3 dimensional task space with 6 variables. After the Jacobian matrix is decomposed into linear part and angular part, constraint on joint velocities is transformed into linear task velocity and angular task velocity through the decomposed Jacobian matrices. Under the assumption of redundant robot manipulators, several optimization problems which utilize the redundancy are formulated to be solved by linear programming technique or sequential quadratic programming technique. After deriving the solutions of the optimization problems, we give graphical interpretations for the solutions.

Keywords : manipulability, decomposed Jacobian, optimal trajectory planning, manipulability polytopes

I. 서론

여유 자유도 로봇을 이용하면 작업의 성능을 일반 비 여유 자유도 로봇보다 향상시킬 수 있고 다양한 환경 조건에 대처할 수 있기 때문에 이에 대한 연구가 현재 로봇틱 분야에서 활발히 이루어지고 있다[8][9]. 여유 자유도 로봇이나 비 여유 자유도 로봇의 몸체를 설계하거나 또는 로봇의 형태를 결정하려고 할 때 기구학적인 설정을 평가할 수 있는 척도가 필요하게 되며 이러한 여러 가지 평가 기준 중 조작도(manipulability)라는 것이 있다[1]. 그러나 연구 초창기에 제안된 조작도는 물리적으로 서로 일치하지 않은 요소들을 묶어서 해석하기 때문에 실제 로봇의 조작도를 정확히 제시하지 못했고[2], 또 일반적으로 로봇들이 작업하는 환경의 3개 이상의 변수들을 필요로 하는 경우가 많아(예를 들면 3차원 작업공간은 3개의 직선운동 변수와 3개의 회전운동 변수에 의해서 기술된다) 이에 대한 조작도를 시각적으로 나타내는데 어려움을 겪고 있다[3].

일반적인 로봇 작업공간은 서로 다른 물리적 단위를 가진 두개의 요소로 이루어져 있기 때문에[4] 본 연구에서는 기존 조작도를 선 운동량과 회전 운동량으로 분리하여 각각 분리하여 다룸으로써 이러한 물리적 불일치를 해결하고자 한다. 본 논문에서는 자코비안 행렬에서 직선 운동과 회전 운동에 관련된 부분을 서로 분리시킴으로써 직선 조작도와 회전 조작도를 분리하여 정의하였다. 직선 조작도를 이용해 작업 공간에서 임의의 또는 주어진 방향을 따라 로봇 매니퓰레이터의 위치를 쉽게 변환할 수 있는 정도를 정량적으로 나타낼 수 있고, 또

회전 조작도를 이용해 작업 공간에서 임의의 또는 주어진 방향을 중심으로 로봇 매니퓰레이터의 자세 변환의 용이성을 정량적으로 정의할 수 있게 하였다. 그리고 조작도를 구현하는 방법에 있어서 2 노움으로 관절 각속도 제약조건을 기술하여 얻은 조작도 타원에 의한 방법은 로봇의 각 관절의 제약조건을 작업 공간으로 정확하게 변환시키지 못하므로[3][5] 여기서는 조작도 다각형에 의한 조작도 개념을 이용하여 무한대 노움으로 관절 속도의 제약 조건이 주어진 경우에 대해 중점으로 다루었다 [6]-[8].

여유 자유도 로봇의 작업공간상에서의 속도에 대한 여유 자유도를 최대한으로 활용하기 위해서 본 연구에서 정의된 직선 조작도와 회전 조작도를 목적 함수 및 제약 조건에 포함시키는 다양한 최적화 문제들을 정의하였다. 이러한 최적화 문제들은 선형 프로그래밍 문제 또는 2차 방정식 프로그래밍(quadratic programming)과 같은 구속 조건하의 일반 최적화 문제로 정리 될 수 있으므로 [7] 주어진 로봇의 자세에 대한 각 최적화 문제의 해를 구하여 그 결과를 수치적으로 또 그림으로 분석하였다.

본 논문의 구성은 다음과 같다. 먼저 II 장에서는 I 장의 서론에 이어 현재까지 연구된 조작도 해석 방법들에 대해 소개하고 III 장에서는 II 장에서 소개한 조작도 해석을 기존의 방법으로 하였을 때의 문제점과 이것을 해결하기 위해 기존의 3 차원 공간의 6 변수들을 각각 3 개의 변수들로 분리하여 조작도 해석을 하는 방법에 대해 기술한다. 그리고 IV 장에서는 본 논문에서 시도한 조작도 해석 방법을 이용하여 여유 자유도 로봇을 조작할 때 고려해볼 수 있는 몇 가지 문제들을 항상 전체 최적값(global minimum)을 갖는 최적화 문제로 구체화하고, V 장에서는 그 문제들에 대한 예제들의 해들과 그 결과들의 특성, 그리고 기존의 조작도 해석보다 나아진

접수일자 : 1998. 6. 19., 수정완료 : 1998. 12. 22.

이지흥, 원경태 : 충남대학교 메카트로닉스 공학과

※본 연구는 한국과학재단 핵심연구과제 지원하에 이루어졌습니다.

점들에 대해서 기술한다. 마지막으로 본 논문의 의의와 앞으로 추후 연구에 대해서는 VI 장에서 언급한다.

II. 조작도 지수

1. 조작도 타원

우선, n개의 회전 관절을 갖는 로봇이 m 차원의 작업 공간에서 작업을 하고 있을 때 이 로봇의 정기구학식(forward kinematic equation)을 다음과 같이 나타내기로 하자.

$$X = f(q) \tag{1}$$

여기서 X는 m 차원의 공간상에서 작업하는 로봇의 위치 및 자세에 대한 정보를 나타내는 벡터이고, $q = [q_1, q_2, \dots, q_n]^T$ 는 로봇의 각 관절 각을 나타내는 벡터이다. 또한 작업하는 로봇이 여유 자유도 로봇인 경우를 감안하여 다음과 같이 가정한다.

$$n \geq m \tag{2}$$

속도끼리의 관계를 얻기 위해 (1)을 미분하면,

$$\dot{X} = J(q) \dot{q} \tag{3}$$

가 얻어지는데, 여기서 \dot{X} 는 작업 공간 속도벡터 그리고 $J(q)$ 는 자코비안 행렬이다. 또한 본 논문에서는 설명을 쉽게 하기 위해 각 관절 각속도 \dot{q}_i 는 그 관절의 최대 속도에 대해 단위 화되어 있다고 가정한다. 즉, \dot{q}_i 의 범위는 (4)와 같다.

$$|\dot{q}_i| \leq 1, \tag{4}$$

이러한 수식의 간략화는 (3)에서 자코비안 행렬 J에 행렬 $R = \{1/\dot{q}_{max,1}, 1/\dot{q}_{max,2}, \dots, 1/\dot{q}_{max,n}\}$ 을 곱해서 얻은 의사 자코비안 행렬을 사용함으로써 얻어낼 수 있다 [3][4].

조작도 타원은 관절속도 변수 벡터 \dot{q} 이 $\|\dot{q}\|_2 = \sqrt{\sum \dot{q}_i^2} \leq 1$ 이라는 조건으로부터 다음과 같이 얻을 수 있다[1].

$$\dot{X} J^T J \dot{X} \leq 1 \tag{5}$$

여기서 $\|\dot{q}\|_2$ 는 벡터 \dot{q} 의 2-노름을 나타내고, J^+ 는 여유 자유도 로봇인 경우에는 J의 의사역행렬(pseudo-inverse)이 되고, 그렇지 않은 경우에는 J의 역행렬(inverse)이 된다. 이 조작도 타원은 주어진 관절 속도의 범위 내에서 작업공간에서 로봇의 작업기가 움직일 수 있는 속도 영역을 표시하는 도형이 되고 이전의 많은 연구에서 이러한 척도를 정량적으로 정의하였는데 그것은 다음과 같으며 조작도 지수라 불린다[1].

$$\omega_e = \sqrt{\det[J(q)J^T(q)]} = \sigma_1 \sigma_2 \dots \sigma_m, \tag{6}$$

여기서 σ_i 는 J의 특이값(singular value)을 나타내는 스칼라이고, 이 조작도 지수 ω_e 는 조작도 타원의 부피와 비례한다.

또 작업공간에서 주어진 방향으로 로봇 작업기를 얼마나 잘 움직일 수 있나하는 척도를 (5) 식으로부터 기술한 연구도 있으나 사용된 관절각도 제약 조건이 각 관절의 회전 속도가 고려된 것이 아니고, $\|\dot{q}\|_2 \leq 1$ 라는 모든 관절의 구동 에너지의 최소화라는 의미로서 정의된 제약조건으로부터 유도된 식이라 주어진 최대 관절 속도를 제한하는 물리적인 의미는 적다고 볼 수 있다. 그러므로 주어진 작업공간 방향으로의 운동은 다음절의 조작도 다각형에 의해 정량화 하는 것이 바람직하다.

2. 조작도 다각형

본 절에서는 로봇의 각 관절 공간상에서 주어진 제약 조건으로 $\|\dot{q}\|_2 \leq 1$ 라는 제약조건 대신 $\|\dot{q}\|_\infty = \max |\dot{q}_i| \leq 1$ 라는 제약 조건을 이용하여 이에 대응되는 작업 공간상의 영역을 결정하는 또 다른 방법에 대해서 설명한다. 즉 앞 절에서는 주어진 로봇의 각 관절각과 그 관절들의 최대 각속도에 대한 제약조건이 2-노름으로 주어진 경우에 작업 공간상에서 로봇의 속도 영역이 타원으로 표현되는 방법에 대해 설명하였는데 보통의 로봇의 경우에는 관절 회전 속도에 대한 제약 조건이 2-노름으로 주어지기보다는 각 관절별로 관절회전 속도의 최대 허용값이 주어지는 경우가 많다. 이 경우도 앞에서와 같이 단위화된 최대값으로 수식을 정리하고 각 관절 각속도의 제약조건을 ∞ -노름으로, 즉 $\|\dot{q}\|_\infty = \max |\dot{q}_i| \leq 1$ 으로 하고 작업공간에서의 허용 영역을 구하면 결과적으로 다각형(polytope)꼴로 구해지는데 이에 대해 설명하고자 한다.

우선 (3)을 성분별로 다시 정리하면 (7)과 같은 식을 얻을 수 있다.

$$\begin{aligned} \dot{X} &= [J_1 J_2 \dots J_n] [\dot{q}_1 \dot{q}_2 \dots \dot{q}_n]^T \\ &= \dot{q}_1 J_1 + \dot{q}_2 J_2 + \dots + \dot{q}_n J_n, \end{aligned} \tag{7}$$

여기서 J_i 는 자코비안 행렬 J의 i번째 열 벡터이다. 그리고 \dot{X} 은 계수가 $|\dot{q}_i| \leq 1$ 인 자코비안 행렬 J의 열 벡터들의 선형 조합으로 이루어져 있다고 볼 수 있다. 따라서 각 계수들의 최대 또는 최소 값들의 조합으로 \dot{X} 에 대한 점들을 구한 다음, 필요 없는 내부의 점들을 제거하는[7] 과정을 거쳐 남은 점들을 연결하면 다각형 꼴의 허용 영역을 얻을 수 있고 이것을 조작도 다각형이라고 한다.

이렇게 구해진 조작도 다각형에 대해서도 작업공간에서 임의의 방향으로의 조작도와 특정한 방향으로의 조작도가 정량적으로 계산될 수 있는데 이는 다음과 같다. 즉, [5]에서 설명한 바와 같이 임의의 방향으로의 조작도 지수는 조작도 다각형의 부피를 나타내는 (8)과 같이 나타낼 수 있다.

$$\omega_p = \sum S_k \tag{8}$$

여기서 S_k 는 스칼라로 다음과 같이 정의될 수 있다.

$$S_k = \frac{1}{d_n} |\det([V_{k_1}, V_{k_2}, \dots, V_{k_n}])|. \tag{9}$$

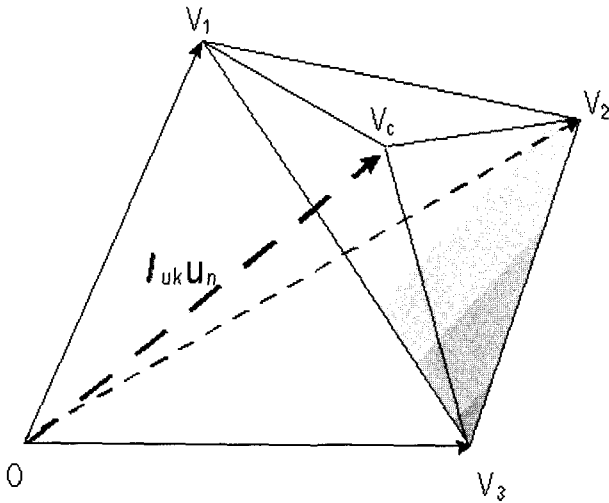


그림 1. 단위 다각 썰의 분해.

Fig. 1. Cone-cell decomposed by vector u in 3 dimensional task space.

그리고 (9)에서 d_n 은 작업 공간의 차원에 따라 정해지는 상수이고, $[V_{k_1}, V_{k_2}, \dots, V_{k_n}]$ 는 k 번째 원시 다각 썰의 꼭지점들의 집합으로 $n \times n$ 정방 행렬이다[5].

또한 작업공간에서 특정한 방향으로의 조작도는 선형 대수적 방법이나 수치 해석적 방법으로 구할 수 있다. 선형 대수적 방법은 우선 3차원 다각형인 경우 조작도 다각형의 원시 꼭지점 집합 V_{k_i} 와 다각형의 중심에서부터 시작하는 주어진 작업공간에서의 벡터 $u = \sum_{k=1}^n \lambda_k V_{k_i}$, $\lambda_{k_i} \geq 0 = l_{uk} u_n$ 에 대하여, 원점에서부터 다각형과 작업 공간 속도 벡터의 교차점까지의 거리를 결정하는 식을 유도한다. 그림 1에서 보듯이 전체 다각썰의 부피는 벡터 $l_{uk} u_n$ 에 의해 분해되는 단위 다각썰 즉, 그림 1의 경우에는 꼭지점 $\{O, V_c, V_1, V_2\}$ 으로 이루어진 단위 다각썰과 꼭지점 $\{O, V_c, V_2, V_3\}$ 으로 이루어진 단위 다각썰의 부피의 합으로 다음과 같이 표현된다.

$$S_k = \frac{1}{d_n} l_{uk} \{ |det([u_n, V_{k_1}, \dots, V_{k_n}])| + |det([V_{k_1}, u_n, \dots, V_{k_n}])| + \dots + |det([V_{k_1}, \dots, V_{k_{n-1}}, u_n])| \} \quad (10)$$

여기서 $u_n = u / \|u\|_2$ 이다. (9)와 (10)을 보면 각 꼭지점들의 좌표가 주어지면 조작도 다각형 공간에서 주어진 벡터 u 방향의 최대 길이인 l_{uk} 가 계산되어 그 방향으로의 조작도 값을 얻을 수 있다.

위 방법 외에 특정한 방향으로의 조작도는 수치 해석적으로도 얻을 수 있다. 즉 우선 주어진 단위 벡터 \dot{X}_n 에 대해서 $\dot{X}_n = J \dot{q}$ 의 무한대 노름 해(infinite norm solution)인 \dot{q}^* 를 구한 다음, 그 벡터의 무한대 노름 해의 성분 중 다각형의 경계선에 가장 가까운 성분을 구하고 그 성분의 최대값까지의 여유도와 관계된 량 γ 를 다음으로 구하여 그것의 역수를 취하면 된다. 이를 정리하면 다음과 같다.

정리 1 : 작업 속도공간상에서 단위 벡터 \dot{X}_n 가 주어졌을 때 \dot{q}^* 을 (11)의 해 중에서 무한대 노름이 최소인 해라고 하고 여유도 지수 γ 를 (12)식으로 구한다고 하자. 그러면 γ^{-1} 가 $\|\dot{q}\| \leq 1$ 이라는 제약 조건하에서 $J(q) \dot{q}$ 가 가질 수 있는 \dot{X}_n 방향으로의 최대 길이가 된다. 즉, 그 방향으로의 최대 조작도 값이 된다.

$$\dot{X}_n = J(q) \dot{q} \quad (11)$$

$$\gamma = \max_i |\dot{q}_i^*| \quad (12)$$

증명 : 우선 (11)의 양변에 γ^{-1} 을 곱함으로써 \dot{X} 의 방향을 가진 작업 공간상의 최대 속도 벡터에 대해 (13)과 같은 공식을 얻을 수 있다.

$$\gamma^{-1} \dot{X}_n = \gamma^{-1} J(q) \dot{q}^* \quad (13)$$

여기서 γ^{-1} 은 스칼라이므로, (13)은 다음과 같이 변환될 수 있다.

$$\gamma^{-1} \dot{X}_n = J(q) \gamma^{-1} \dot{q}^* = J(q) \hat{q}^* \quad (14)$$

(14)는 선형 방정식이고 $\|\hat{q}^*\|_\infty = 1$ 이므로 $\|\dot{q}\|_\infty \leq 1$ 이라는 제약조건하에서 γ^{-1} 보다 큰 확장 계수를 찾을 수 없다.

III. 작업 공간 속도의 분리

다각형 방법에 의한 조작도 해석은 전체적으로나 특정한 방향으로의 조작도에 대한 정확한 해를 제공하고 3차원까지 명확한 기하학적 해석이 가능하다. 그러나 이 방법은 타원에 의한 방법보다 구현하기 위한 계산 량이 많고 또 보통 로봇이 사용되는 직선운동 3자유도, 회전운동 3자유도로 모두 6 차원 변수를 가지면 기하학적 해석이 매우 어렵다. 따라서 본 논문에서는 6 개의 변수를 갖는 일반적인 3 차원 작업공간에서의 조작도 다각형을 좀 더 쉽게 분석하기 위해서 6 개의 변수를 구성하는 요소들을 물리적 특성에 따라 두 개의 3 차원 공간들로 분리하고자 한다. 현재 로봇이 사용되고 있는 작업 속도 공간은 3 차원의 선속도 공간과 3 차원의 회전속도 공간으로 이루어져 있으며 우리는 이것을 (15)와 같이 표현할 수 있다.

$$\dot{X} = [V_x, V_y, V_z, \omega_x, \omega_y, \omega_z]^T \quad (15)$$

(15)로부터 (3)을 다음과 같이 선속도 공간과 회전속도 공간에 대한 식으로 분리할 수 있다.

$$V = [V_x, V_y, V_z]^T = J_v \dot{q} \quad (16)$$

$$\omega = [\omega_x, \omega_y, \omega_z]^T = J_\omega \dot{q} \quad (17)$$

여기서

$$J = \begin{bmatrix} J_v \\ J_\omega \end{bmatrix} \quad (18)$$

로 자코비안 행렬을 분리했음을 밝히고, V_x, V_y, V_z 및

$\omega_x, \omega_y, \omega_z$ 는 스칼라이다.

(16)과 (17)을 보면 V 와 ω 가 각각 3×1 벡터이고 따라서 일반 6 변수 작업 속도공간을 (16)과 (17)을 통해 정의된 선속도 공간과 회전속도 공간에 대한 조작도로 각각 분리한 다음 2장에서 소개한 다각형 방법을 적용하여 조작도 해석을 한다면 각 공간이 3 변수를 가지므로 수치적인 해석은 물론 그림에 의한 해석도 매우 쉬워진다. 이렇게 얻은 다각형들을 이용하여 어떤 특정한 방향으로의 조작도뿐만 아니라 분리된 공간에 대한 임의의 작업 방향에 대한 조작도도 분석하여 정량적으로 기술할 수 있다. 즉 작업 속도 공간이 분리되어 있으므로 사용자가 직면한 문제에 따라 독립적인 조작도 분석을 할 수 있다. 예를 들어 로봇 조작시 주요 관심 되는 부분이 선속도에 대한 것이라면 (16)에 의한 조작도 지수를 최적화 문제의 성능 지수로 사용할 수 있으며 그 반대의 경우라면 최적화 문제의 성능 지수로서 (17)에 의한 조작도 지수를 사용할 수 있다.

IV. 최적화 문제

이번 장에서는 작업 속도 공간을 선속도와 회전속도로 분리하여 다루자 할 때 마주칠 수 있는 여러 가지 문제들에 대해 정의한다. 우선 (16)과 (17)로부터 직선운동과 회전운동에 대한 전달 비를 정의하고자 한다. (16)에서 V 의 크기와 방향을 분리하면 (16)은 다음과 같이 나타낼 수 있다.

$$\alpha V_n = J_v \dot{q} \tag{19}$$

여기서 α 는 스칼라이고 V_n 은 단위 벡터이다. (19)의 양변에 V_n^T 를 곱하면 다음과 같은 식을 얻을 수 있다.

$$\alpha V_n^T V_n = V_n^T J_v \dot{q} \tag{20}$$

그리고 여기서 $V_n^T V_n = 1$ 이므로 (20)은,

$$\alpha = V_n^T J_v \dot{q} \tag{21}$$

와 같이 간단화 된다. 위와 같은 방식으로 회전속도 부분에 대해서도 다음과 같이 구할 수 있다.

$$\beta \omega_n = J_\omega \dot{q} \tag{22}$$

$$\beta \omega_n^T \omega_n = \omega_n^T J_\omega \dot{q} \tag{23}$$

$$\beta = \omega_n^T J_\omega \dot{q} \tag{24}$$

(21) 및 (24)의 α, β 는 각각 V_n 과 ω_n 의 함수이므로 추후 $\alpha(V_n), \beta(\omega_n)$ 으로 각각 표기하기로 한다.

이상의 수식들을 이용하여 분리된 각 작업속도 공간에서 속도 전달비를 정의할 수 있다.

정의 1 : 로봇의 각 관절 속도가 (4)와 같이 주어졌을 경우 (21)의 $\alpha(V_n)$ 는 작업 공간상에서 단위 벡터 V_n 방향으로의 선속도 전달비(linear velocity transmission ratio)라 하고, (24)의 $\beta(\omega_n)$ 는 작업 공간상에서 단위 벡터 ω_n 방향으로의 회전 속도 전달비(angular velocity

transmission ratio)라고 한다.

여기서 속도 전달비는 주어진 속도 방향으로의 크기 전달을 나타내는 계수라는 의미로 사용되었다.

선속도와 회전속도를 분리하여 다루고자 하는 여러 가지 최적화 문제에 대해 설명하기에 앞서, 선형 프로그래밍 문제(linear programming)와 제약조건이 있는 최적화 문제(constrained optimization problem)의 표준적인 형태에 대해서 간단히 소개하고자 한다.

LP : 일반적인 선형 프로그래밍(LP) 문제는 (26)을 만족하면서 (25)에 대한 해를 구하는 형태로 이루어져 있다[14].

$$\min_x \text{ or } \max_x : f^T x \tag{25}$$

$$Ax \leq b \tag{26}$$

여기서 f 는 x 의 계수에 해당하는 상수 값들을 나타내는 벡터이고, A 는 상수 행렬을, 그리고 b 또한 상수 벡터가 된다. 즉 선형 프로그래밍 문제에서는 목적 함수와 제약 조건식 모두 선형 함수로 이루어져 있음을 알 수 있다.

COP : 수치적인 방법에 의해 해결될 수 있는 제약조건식이 있는 최적화 문제(COP)는 일반적으로 (28), (29) 그리고 (30)을 만족시키면서 (27)의 해를 찾는 형태로 되어 있다. 그리고 (28)은 등식 제약조건이고, (29)는 부등식 제약조건, 그리고 (30)은 최적화 문제의 변수인 x 의 범위에 대한 제약 조건이다.

$$\min_x \text{ or } \max_x : f(x) \tag{27}$$

$$h_i(x) = 0; i = 1, 2, \dots, p \tag{28}$$

$$g_i(x) \leq 0; i = 1, 2, \dots, k \tag{29}$$

$$x_{il} \leq x_i \leq x_{iu} \tag{30}$$

여기서 x_{il} 과 x_{iu} 는 각각 최적화 문제의 i 번째 변수 x_i 가 가질 수 있는 최소 값과 최대 값을 나타내고, p 와 k 는 각각 등식 제약조건의 개수와 부등식 제약조건의 개수를 의미한다.

이제부터는 앞서 소개한 정의 및 표현들을 이용하여 (15)와 같은 작업공간의 속도 벡터가 주어졌을 때 (16)과 (17)을 이용하여 선속도와 회전속도로 분리하여 다루는 여러 가지 문제에 대해 정리하고자 한다. 그리고 다음의 문제들에 대한 목적 함수나 제약조건식은 모두 볼록(convex function)하므로 그것들의 해는 항상 전체 최적값(global optimal solution)을 갖는다[14].

첫 번째 문제는 로봇 작업기를 주어진 축을 중심으로 회전하는 속도를 일정하게 하면서 이동속도를 최대로 하는 각 관절 속도를 구하는 내용인데 두 번째, 세 번째, 네 번째 문제도 물리적 의미가 수식에 의해 쉽게 해석될 수 있으므로 자세한 설명은 생략하기로 한다.

문제 1 : (16), (17) 그리고 (4)의 제약 조건하에 $\alpha(V_n)$ 의 최대값을 찾아라.

이 문제는 회전속도의 방향과 크기를 일정하게 유지

하면서 주어진 선속도 방향으로의 최대 크기를 갖는 속도를 찾는 문제로서 LP 방법을 이용하여, (4), (32) 그리고 (33)을 만족시키면서 (31)의 해가 되는 \dot{q} 을 찾는 문제로 정리될 수 있다.

$$\max : V_n^T J_v \dot{q} \quad (31)$$

$$\begin{bmatrix} J_{o1} \\ J_{o2} \\ J_{o3} \end{bmatrix} \dot{q} = \begin{bmatrix} \omega_{n1} \\ \omega_{n2} \\ \omega_{n3} \end{bmatrix} \quad (32)$$

$$\begin{bmatrix} V_{n2}J_{o1} - V_{n1}J_{o2} \\ V_{n3}J_{o1} - V_{n1}J_{o3} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (33)$$

여기서 J_{oi} 는 행렬 J_o 의 i 번째 행 벡터이고 V_{ni} 는 벡터 $V_n = V / \|V\|$ 의 i 번째 성분, 그리고 J_{vj} 는 행렬 J_v 의 j 번째 행 벡터이다. 참고로 (32)는 (17)의 제약 조건을 만족시키는 식이고, (33)은 선속도 벡터 V_n 의 방향을 유지하기 위한 제약 조건 식으로서 $\dot{V}_n = \alpha * \dot{V}$ 이고, $\dot{V}_n = \{ \dot{x}_n, \dot{y}_n, \dot{z}_n \}^T$, $\dot{V} = \{ \dot{x}, \dot{y}, \dot{z} \}^T$ 라고 할 때 벡터 \dot{V}_n , \dot{V} 은 같은 방향을 갖게 하는 조건(34)와 (35)를 (26)과 같이 선형 프로그램의 일반적인 제약 조건식 형태로 재구성한 것이다.

$$\dot{x}_n / \dot{y}_n = \dot{x} / \dot{y} \quad (34)$$

$$\dot{x}_n / \dot{z}_n = \dot{x} / \dot{z} \quad (35)$$

문제 2 : (16), (17), (4) 그리고 $\beta(\omega_n) \geq 0$ 의 제약 조건하에 $\alpha(V_n)$ 의 최대값을 찾아라.

이 문제는 회전 속도의 방향만을 유지하면서 주어진 선속도 방향으로 최대로 로봇을 움직이고자 할 때 정의되는 문제로서 회전 속도 크기를 구속하지 않는다는 것을 제외하고는 문제 1과 같다. 즉 문제 1의 목적 함수를 나타내는 (31)은 이 문제에서도 그대로 적용될 수 있다. 그리고 문제 1의 구속 조건식들 중 회전 속도의 방향과 크기 모두를 구속하는 (32)대신 회전 속도의 방향만을 구속하는 (36)으로 대체하는 것 외에 나머지 제약 조건식들을 문제 2에 그대로 적용할 수 있다.

$$\begin{bmatrix} \omega_{n2}J_{o1} - \omega_{n1}J_{o2} \\ \omega_{n3}J_{o1} - \omega_{n1}J_{o3} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (36)$$

여기서 ω_{ni} 는 회전속도 벡터 ω_n 의 i 번째 성분이고, J_{oj} 는 행렬 J_o 의 j 번째 행 벡터이다.

문제 3 : (16), (4) 그리고 (37)의 제약 조건하에서 $\alpha(V_n)$ 의 최대값을 찾아라.

$$\|\omega\|_2 \leq \omega_0 \quad (37)$$

이 문제는 회전 속도의 크기를 나타내는 길이를 ω_0 로 제한하면서 로봇을 주어진 선속도 방향으로 최대 속도로 움직이고자 할 때 적용되는 문제로서 목적 함수로는 (31)을, 구속 조건 식으로는 (4), (33) 그리고 (37)을 이용하여 정리될 수 있다. 그리고 (37)은 다음과 같이 구체화되어 실제 문제에 적용된다.

$$\dot{q}^T J_o^T J_o \dot{q} - \omega_0^2 \leq 0 \quad (38)$$

즉, 이 문제도 회전 속도의 방향에 대한 구속이 없다는 것을 제외하고 문제 1, 2와 매우 비슷한 형태로 이루어져 있음을 알 수 있다. 그러나 (38)을 보면 회전 속도의 크기의 제한이 타원의 식 형태로 되어 있어서 문제 1, 2 그리고 문제 4에서처럼 선형 프로그래밍(LP)으로 풀 수가 없다. 따라서 이 문제는 제약 조건식이 있는 최적화 문제(COP)를 이용하여 풀어야 하는데 본 논문에서는 이 중 SQP(Sequential Quadratic Programming) 방법을 사용하여 해를 구하기로 한다.

문제 4 : (4), (16) 그리고 (39)의 제약조건하에서 $\alpha(V_n)$ 의 최대값을 구하라.

$$\|\omega\|_\infty \leq \omega_0 \quad (39)$$

문제 4는 회전 속도의 방향에 대한 구속이 없고 단지 크기에 대한 구속만 있다는 점에서 ω 문제 3과 비슷하다. 차이점이 있다면 문제 3은 회전 속도의 크기가 회전 속도 벡터의 2-노름으로 주어졌고 이 문제는 (39)와 같이 회전 속도 벡터 ω 의 무한대 노름으로 주어졌다는 점이다. 그리고 실제 문제에서 적용될 때는 (39)는 (40)과 같이 정리하여 사용한다.

$$\begin{bmatrix} J_{o1} \\ J_{o2} \\ J_{o3} \end{bmatrix} \dot{q} \leq \begin{bmatrix} \omega_0 \\ \omega_0 \\ \omega_0 \end{bmatrix} \quad (40)$$

여기서 J_{oi} 는 행렬 J_o 의 i 번째 행 벡터이다.

V. 예제

앞서 설명한 문제들에 대한 예제들을 다루기 위해 그림 2의 8축 여유자유도 로봇 ARM-II(Advanced Research Manipulator II)를 이용하였다[9]. 표 1은 ARM-II 로봇의 기구학적 파라미터들에 대해 보여주고 있다.

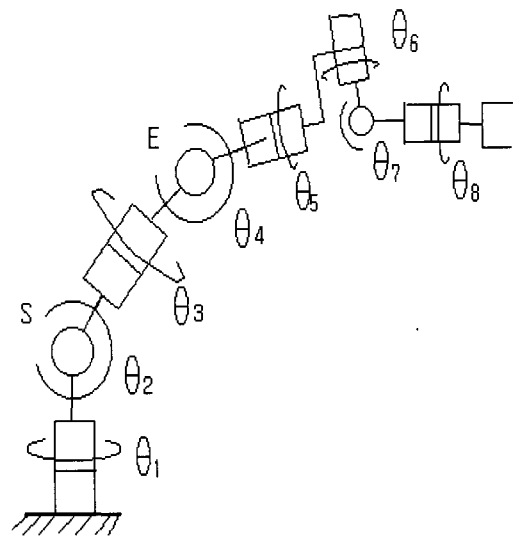


그림 2. 8축 여유 자유도 로봇(ARM-II).
Fig. 2. Redundant eight-degree-of-freedom advanced research manipulator II (ARM-II).

표 1. ARM-II 로봇의 기구학적 매개 변수.
Table 1. Kinematic parameters of ARM-II.

i	a_{i-1} (degree)	a_{i-1} (cm)	d_i (cm)	θ_i (degree)
1	0	0	0	θ_1
2	90	0	0	θ_2
3	-90	0	d_3	θ_3
4	90	0	0	θ_4
5	-90	0	d_5	$\theta_5 - 90$
6	-90	0	0	$\theta_6 + 90$
7	90	0	0	$\theta_7 - 90$
8	90	0	0	θ_8

표 2. 최적화 문제들에 대한 예제 결과.
Table 2. Simulation results for the optimization problems.

	α	β	ω_n
문 제 1	0.3794	0.1346	$\begin{bmatrix} 0.4867 \\ 0.3244 \\ 0.8111 \end{bmatrix}$
문 제 2	0.4172	-6.949e-007	$\begin{bmatrix} 0.4867 \\ 0.3244 \\ 0.8111 \end{bmatrix}$
문 제 3	0.4797	-	$\begin{bmatrix} -0.4618 \\ 0.7843 \\ -0.4057 \end{bmatrix}$
문 제 4	0.4801	-	$\begin{bmatrix} -0.3772 \\ 1 \\ 0.5627 \end{bmatrix}$

표 2는 앞절에서 설명한 문제들에 대해 로봇 자세가 $q = [20, -10, 10, 50, -20, 10, 60, 40]^T$ (degree)이고, 선속도 단위 벡터가 $V_n = [0.8018, 0.5345, 0.2673]^T$ (cm/sec)로 주어졌을 경우의 시뮬레이션 결과들을 보여주고 있다. 여기서 주어진 로봇 자세에 대한 초기 조건은 임의로 설정한 것이고, 선속도 및 회전속도의 제약 조건을 줄 때 고려해야 할 점은 문제 3과 문제 4에서는 회전 속도 벡터 ω 의 방향에 대한 구속이 주어지지 않았으므로 그 문제들의 ω_n 들은 ω 의 방향이 구속 조건으로 주어진 문제 1, 2의 $\omega_n = [0.4867, 0.3244, 0.8111]$ (rad/sec)들과는 다르고 또 그때의 β 값들은 의미가 없다고 볼 수 있다. 그리고 시뮬레이션은 각 문제에 주어진 ω 의 크기는 1보다 작다는 가정 하에서 이루어졌다.

표 2로부터 각 문제들에 따른 α 값들의 크기는 다음과 같은 순서로 이루어져 있음을 알 수 있다.

$$\alpha_{probl} \leq \alpha_{probl2} \leq \alpha_{probl3} \leq \alpha_{probl4} \quad (41)$$

(41)과 같은 특성은 각 문제들의 구속 정도에 따라 얻어지게 되는데 본 논문의 시뮬레이션에 적용된 각 문제들에 따른 구속 정도는 표 3과 같다.

표 3을 보면 각 문제들의 구속 정도는 (42)와 같고, 즉 이것은 (41)에서 보여주는 각 문제들에 따른 α 값들의 크기 관계하고 반대가 됨을 알 수 있다.

$$const_{probl} \leq const_{probl3} \leq const_{probl2} \leq const_{probl4} \quad (42)$$

여기서 $const_{probl}$ 는 i 번째 문제의 구속정도를 의미한다.

표 3. 각 최적화 문제들의 구속 정도.
Table 3. Constraints of each optimal problem.

문 제	구속 조건	ω 의 방향 구속	ω 의 크기 구속	ω 의 크기 구속 정도
문 제 1		있음	있음	주어진 ω 의 크기
문 제 2		있음	있음	$\ \omega\ _2 \leq 1$
문 제 3		없음	있음	$\ \omega\ _2 \leq 1$
문 제 4		없음	있음	$\ \omega\ _\infty \leq 1$

표 4. 무한대 노음 방법을 이용하여 얻은 α 값.
Table 4. α 's for the problems obtained by minimum l_∞ solution technique.

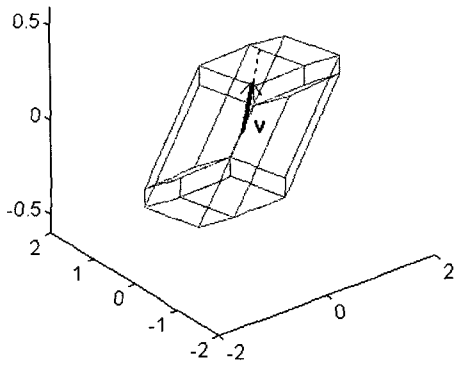
문 제	α 값
문 제 1	0.3794
문 제 2	0.4172
문 제 3	0.4797
문 제 4	0.4801

표 4는 표 2에서 얻을 수 있는 \dot{X} 에 대해서 무한대 노음 해[12]를 구하는 방법을 이용하여 얻은 α 값들에 대해 나타내고 있다.

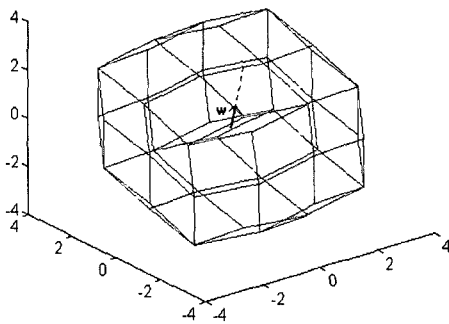
표 2와 4의 결과들을 보면 결국 표 2에 있는 문제들의 α 값들과 표 4에 있는 문제들의 α 값들이 서로 같다는 것을 알 수 있고, 따라서 문제 1에서 4가지에서 구해진 \dot{X} 는 선속도 및 회전속도로 이루어진 작업 속도 공간상의 조작도 다각형의 경계 상에 있는 것을 알 수 있다.

표 2에 나타난 문제 1과 문제 3에 대한 결과적인 선속도와 회전 속도 조작도를 각각 그림 3과 그림 4에 나타내었다. 그림 3의 (a)와 그림 4의 (a)를 보면 표 2에 있는 문제 1과 문제 3에 대한 α 값들의 크기에 대해서 식적으로 쉽게 비교할 수 있다. 그리고 이러한 그림들을 통해 앞서 말한 것처럼 선속도와 회전 속도가 함께 존재하는 6 변수 작업 속도 \dot{X} 가 조작도 다각형의 경계에 있는 반면, 3 차원으로 각각 분리한 선속도 및 회전 속도 벡터 V , ω 는 조작도 다각형의 안쪽에 있는 것을 알 수 있고, 따라서 회전 속도를 조절하여 선속도의 크기를 주어진 벡터의 방향으로 더 키우거나 또는 선속도를 조절하여 회전 속도의 크기를 주어진 벡터의 방향으로 더 키울 수 있음을 알 수 있다.

표 5에서는 앞의 시뮬레이션의 경우와 같은 로봇 자세와 선속도, 회전 속도의 단위벡터가 주어졌을 때 2장에서 소개한 무한대 노음을 이용한 방법을 통해 얻은 선속도의 최대 크기 γ^{-1} 와 표 2의 α 값들에 대해 비교하였다. 표 5를 통해 선속도와 회전 속도 중 원하는 쪽의 성능 향상은 작업 속도 공간을 분리하여 조작도 분석을 통해 이루어 질 수 있음을 알 수 있다. 물론 이러한 이득은 여유 자유도에 기인했음을 다시 한번 밝혀 둔다. 마지막으로 표 2와 5의 문제 2에 대한 β 값이 음수가 나왔는데 이는 수치 해석적인 최적화 문제 해결시 설정하는 구속 조건에 대한 허용 오차 때문에 생긴 현상이다.



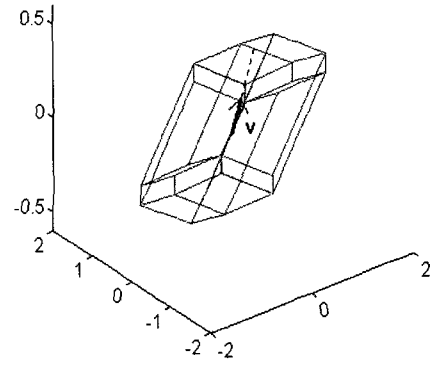
(a)



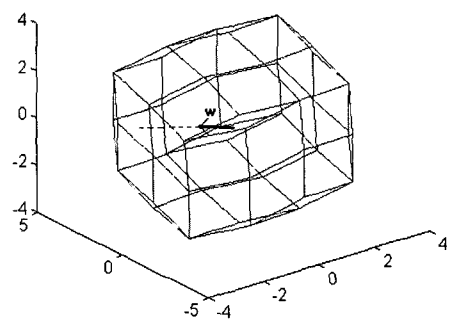
(b)

그림 3. (a) 문제 1에 대한 결과 그림(선속도 조작도), (b) 문제 1에 대한 결과 그림(회전 속도 조작도).

Fig. 3. (a) Visualization for the solution of problem 1 with polytope. (linear manipulability), (b) visualization for the solution of problem 1 with polytope (angular manipulability).



(a)



(b)

그림 4. (a) 문제 3에 대한 결과 그림(선속도 조작도), (b) 문제 3에 대한 결과 그림(회전 속도 조작도).

Fig. 4. (a) Visualization for the solution of problem 3 with polytope. (linear manipulability), (b) visualization for the solution of problem 3 with polytope. (angular manipulability).

표 5. 결과 비교.

Table 5. Comparison of the results.

문 제	α	β	γ^{-1}	이 득	
무한대 넘 방법	-	-	0.1348	정 의	값
문 제 1	0.3794	0.1346	-	a/γ^{-1}	2.8145
문 제 2	0.4172	-6.949*10e-7	-	a/γ^{-1}	3.0949
문 제 3	0.4797	-	-	a/γ^{-1}	3.5586
문 제 4	0.4801	-	-	a/γ^{-1}	3.5616

VI. 결론

본 논문에서는 자코비안 행렬을 분리함으로써 기존의 조작도를 선속도 조작도와 회전 속도 조작도로 구분하여 다룰 수 있도록 하였다. 분리된 각 조작도는 3 변수만을 가지므로 기하학적인 해석이 편리하고, 여유 자유도 로봇에 대한 기존의 기하학적 조작도를 이용하여 각각의 속도를 최적화하는 방법에 비해 기존의 조작도를 분리하여 다룸으로써 각각의 작업공간 속도를 좀 더 효율적으로 최적화 할 수 있게 함으로써 작업공간 속도에 대한 여유 자유도를 최대한 활용할 수 있게 되었다.

여유 자유도 로봇의 이러한 여유자유도의 최대 활용을 위해서 분리된 조작도를 이용하여 각각의 해를 구하는 문제를 표준형의 선형 프로그래밍이나 구속 조건이 있는 최적화 문제로 정리하였다. 또한 그러한 최적화 문제의 해가 기존의 조작도를 사용하는 방법에 비해 나은 해가 구해짐을 보였다.

본 연구에서는 기하학적인 요소만을 고려하여 속도에 관련된 조작도만 다루었으나 앞으로는 로봇 매니플레이터의 동적 특성까지 포함된 경우에 작업공간상에서 직선 방향 또는 비트는 방향으로의 힘 또는 파워를 최적화 하는 문제를 다루는 것과 이러한 여유 자유도 로봇의 동적 특성을 고려하게 될 때 발생하는 영 운동 동력학(null motion dynamics)이 로봇 매니플레이터 동적 조작도에 미치는 영향에 대해 연구할 예정이고, 또 이러한 분리된 조작도를 통해 경로 추적 제어에 적용하는 문제에 대한 연구를 진행할 예정이다.

참고문헌

[1] T. Yoshikawa, "Analysis and control of robot manipulator with redundancy," *Robotics Research, The First International Symposium, The MIT*

- Press, pp. 735-747, 1984.
- [2] K. L. Doty, C. Melchiorri, E. M. Schwartz and Claudio Bonivento, "Robot manipulability," *IEEE Trans. Robotics and Automation*, vol. 11 no. 3, June, 1995.
- [3] J. Lee, "Optimal configuration control for redundant robot manipulators," *Proc. of Int'l Conf. on Automation, Robotics, Control, and Vision*, pp. 2441-2445, Singapore, Dec., 1996.
- [4] A. Bowling and O. Khatib, "Analysis of the acceleration characteristics of non-redundant manipulators," *IEEE/RSJ IROS'95.*, pp. 323-328, 1995.
- [5] J. Lee, "A study on the manipulability measures for robot manipulators," *IEEE/RSJ IROS'97.*, pp. 1458-1465, 1997.
- [6] G. M. Ziegler, *Lectures on Polytopes*, Springer-Verlag, 1995.
- [7] D. R. Chand and S. S. Kapur, "An algorithm for convex polytopes," *Journal of Association for Computing Machinery*, vol. 17, no. 1, pp. 78-86, 1970.
- [8] F. T. Cheng, et. al., "Resolving manipulator redundancy under inequality constraints," *IEEE Trans. Robotics and Automation*, vol 10. no. 1., Feb., 1994.
- [9] R. L. Williams II, *Kinematic equations for Control of the Redundant Eight-Degree-of-Freedom Advanced Research Manipulator II*, Tech. Report NASA TM-4377, 1992.
- [10] Y. Nakamura, *Advanced Robotics, Redundancy and Optimization*, Addison Wesley, 1991.
- [11] R. V. Benson, *Euclidean Geometry and Convexity*, McGraw-Hill Book Company, pp. 188-192, 1966.
- [12] J. A. Cadzow, "A finite algorithm for the minimum l_{∞} solution to a system of consistent linear equations," *SIAM J. Number, Anal.*, vol. 10, no. 4, pp. 607-617, 1973.
- [13] A. S. Deo and I. D. Walker, "Minimum infinity-norm inverse kinematic solution for redundant manipulators," *Proc. of IEEE conf. of Robotics Automation'93*, pp. 388-394, 1993.
- [14] J. S. Arora, *Introduction to Optimum Design*, McGraw-Hill Series in Mechanical Engineering, pp. 81-87, 1989.

이 지 흥

제어 · 자동화 · 시스템공학 논문지 제3권, 제5호, 참조.

원 경 태

제어 · 자동화 · 시스템공학 논문지 제3권, 제5호, 참조.