

다관절 로봇 손의 설계

°백상은, 진상태, 성홍석, 이쾌희
서강대학교 공과대학 전자공학과

Design on the Articulated Robotic Hand

°Sangeun Baek, Sangtae Chin, Honseok Seong, Kwaehi Lee
Department of Electric Engineering, Sogang University

Abstract

An articulated, multifinger mechanical hand can carry out grasping and manipulation operations on objects of different type and shape. In this paper, the architecture of the mechanical hand is presented. Joints are driven by two antagonist tendons. Strain gauges are used to derive tendon tensions, and located in the palm of the hand. Angular deflection of the joints is measured by Hall effect sensors attached to the joints. A multiprocessor-based architecture for controlling the hand is illustrated.

I. 서론

다지 다관절 손 (Hand)은 단순한 형태의 end effector와는 달리 그 작업능력에 있어 유연성과 적응성이 뛰어나다. 이것은 다양한 형태의 물체를 조작할 수 있을 뿐만 아니라 로봇 팔 (Arm)을 움직이지 않고도 작업수행을 가능하게 한다. 따라서 로봇 팔과 손의 결합은 팔에 의한 넓은 작업공간과 손에 의한 빠른 응답성을 제공한다. 본 논문에서는 다지 다관절 손에 관해서만 언급한다. 서강 손은 tendon으로 구동되는 12개의 관절과 24개의 DC 모터들, 각각 12개의 위치와 tension 센서들, 그리고 앞으로 추가될 4개의 손끝 (Tactile) 센서들로 구성되어 있다. 로봇 손의 작업 경로 계획을 세우고 위

치 및 힘 (Force) 제어를 하며 물체와의 적절한 접촉력을 계산하는 것은 단일 프로세서로는 부적합하다. 본 시스템의 제어기는 다중 프로세서 구조를 가지고 있고, 이들 프로세서는 호스트 컴퓨터로부터 각각 적당한 작업을 분배 받는다.

II. 서강 손의 구조

1. 기구학적 구조

로봇 손의 손가락의 갯수와 공간적 위치, 각 링크의 길이를 결정하는 것은 매우 어려운 일이다. 이러한 매개변수들은 경험에 의해서나 인간의 손 구조를 모방하여 결정하는 것이 보통이다. 본 연구실에서는 [1]에서 제안한 한 손가락이 3개의 관절을 가진 구조로 설계하였다. [1]에 의하면, 작업영역에서의 입력 토크로부터 출력 토크로의 오차 전이 (Error propagation)를 최소화하기 위해서는 자코비안 행렬이 다음의 두 조건을 만족해야 한다: (1) 각 열은 직교이고 (2) 각각의 열벡터들은 같은 크기를 가진다. 이러한 조건을 만족하는 점들을 균등점 (Isotropic point)이라고 한다. 그림 1은 손가락의 균등점을 나타낸 것이다.

서강 손은 4개의 손가락을 가지고 있다. 각 손가락의 위치는 대상 물체를 가장 적절한 위치에서 조작할 수 있도록 인간의 손 구조와 실험에 의해서 결정하였다. 그림 2는 간략화 된 서강 손의 구조를 보여주고, 그림 3은 각 관절의 좌표를 표시하고 있다. 각 손가락이 물체와 접촉하는 경우는 그림 4와 같이 8가지가 있다. [2]에 의하면, 손마다

접촉을 고려하여 물체를 잡는 것이 손끝만으로 잡는 것 보다 높은 안정성을 가진다. 서강 손은 손마다 접촉은 고려하지 않고, 손끝 접촉만을 고려한다. 접촉 형태는 마찰이 있는 점접촉 (Point contact)으로 간주한다.

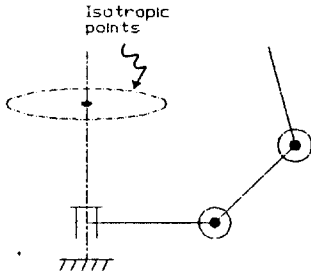


그림 1. 손가락의 구조

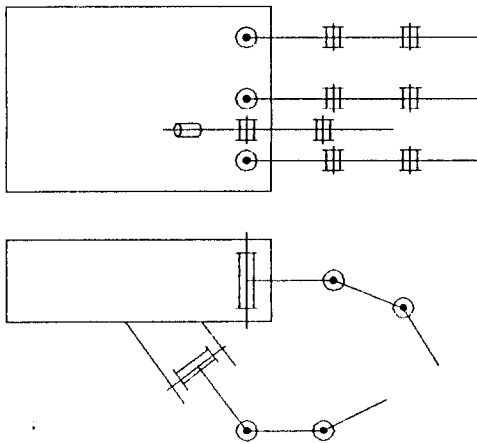


그림 2. 서강 손의 구조

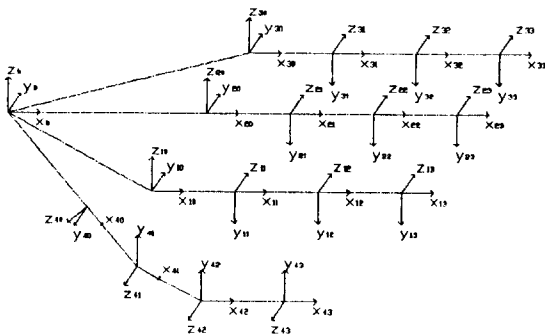


그림 3. 각 관절의 좌표

2. 동력 전달 구조

서강 손의 각 관절은 tendon과 pulley에 의해 구동된다. 그림 5은 서강 손의 동력 전달 구조를 보이고 있다. 하나의 관절에는 그것을 구동하는 한쌍의 pulley가 고정되어 있고, 서로 반대방향으로 연결된 두개의 tendon이 각각의 pulley에 연결되어 있다. 관절을 원하는 방향으로 움직이기 위해서는 해당되는 tendon을 당겨주면 된다. Tendon은 항상 적당한 tension을 유지하여야 한다. 이것은 로봇 손의 손목 부분에 위치한 tendon tension 센서에 의해 tension이 감지되고 실시간 제어기는 이를 유지시켜준다.

관절에 인가되는 토크는 tendon의 tension을 측정하고 수학적 변환을 함으로써 얻어진다[3]. 이때 이 변환식은 pulley 반경의 함수로 표현된다. 다음은 tendon의 tension과 속도, 그에 해당하는 관절의 토크와 각속도와의 관계식을 나타낸다.

$$\dot{\mathbf{x}} = R\mathbf{t} \quad (1)$$

$$\mathbf{t} = R^{-1}\dot{\mathbf{x}} \quad (2)$$

$$\mathbf{u} = R^T\mathbf{m} \quad (3)$$

$$\mathbf{m} = R^{-T}\mathbf{u} \quad (4)$$

여기서,

\mathbf{u} 는 6차원 tendon 속도 벡터,

\mathbf{t} 는 6차원 tendon tension 벡터,

\mathbf{m} 는 6차원 관절 각속도 벡터,

\mathbf{r} 는 6차원 관절 토크 벡터,

R 은 pulley 반경으로 결정되는 상수 행렬

이다.

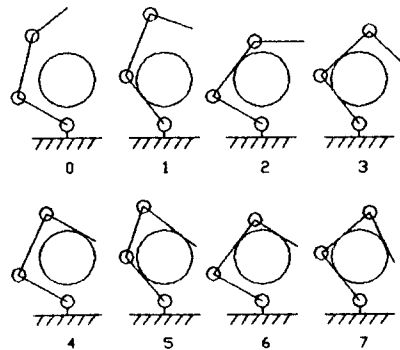


그림 4. 접촉 형상

$$R = \begin{bmatrix} r_1 & -r_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & r_2 & -r_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & r_3 & -r_3 \end{bmatrix} \quad (5)$$

위에서는 각 관절사이의 간섭 (interference)과 tendon과 pulley사이의 마찰이 없다고 가정하였다.

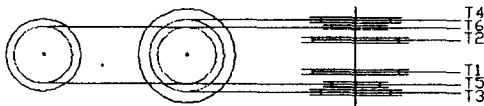
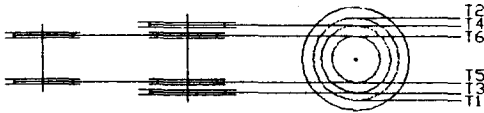


그림 5. 동력 전달 구조

3. 위치 및 tendon tension 센서

로봇 손을 제어하기 위해서는 각 관절의 위치와 tendon의 tension 측정이 필요하다. [1]에서는 DC 모터에 연결한 Incremental Encoder로부터의 회전각과 tendon tension을 측정하여 관절의 위치각을 알아내었다. [6]에서는 각 관절에 홀 효과(Hall effect) 센서를 직접 부착하여 보다 신뢰성 있는 값을 얻었다. 서강 손에서도 홀 효과 센서를 이용하여 관절의 위치각을 측정한다. 그림 6는 각 관절에 위치한 센서를 보여주고 있다.

Tendon tension 센서는 각 관절에 인가되는 토크를 유추해 낼 수 있고, tendon에 일정한 tension을 유지하기 위해서도 필요하다. 이 센서는 스트레인 게이지 (Strain gauge)의 쌍으로 구성되며 그림 7은 서강 손의 내부에 구성된 센서를 보여준다.

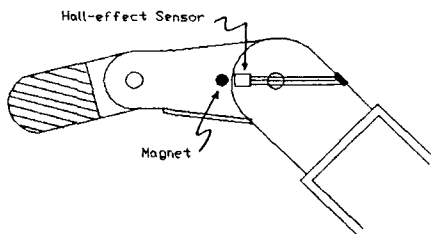


그림 6. 위치 센서

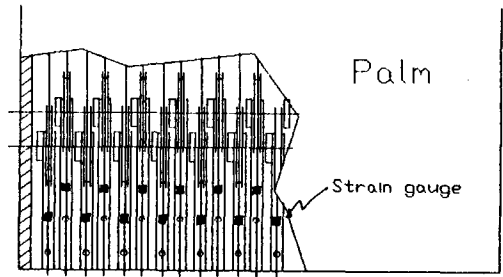


그림 7. Tendon tension 센서

III. 서강 손의 제어기 구조

정교한 제어 규칙을 수행하거나 많은 수의 자유도, 구동기 (Actuator), 그리고 센서를 가진 로봇 시스템에서는 처리할 계산량이 매우 많다. 예를 들면, 귀환 (Recursive) Newton-Euler 방법으로 동력학 방정식을 풀 경우, n개의 자유도를 가지면 일반적으로 132n 번의 곱셈과 111n - 4 번의 덧셈이 필요하다[7].

서강 손은 3개의 관절을 가진 손가락이 4개 있고, 400Hz 이상의 서보 비율 (Servo rate)이 요구되어, 매초당 633,600번의 곱셈과 526,400번의 덧셈이 필요하다. 또한, 경로 계획과 기구학 방정식, 각종 센서 정보의 처리, 그리고 물체와의 접촉력 계산을 고려한다면 단일 프로세서에 의한 로봇 손의 제어는 부적합하다[8][9]. 서강 손의 제어기는 다중 프로세서 구조를 가지고 있어 각각의 프로세서는 하나의 손가락 제어를 담당한다. 호스트 컴퓨터는 이들 프로세서에게 작업을 할당하며 전체적인 로봇 손의 작업을 관리한다.

1. 제어기의 하드웨어

그림 8은 서강 손의 제어기의 전체적인 구조를 나타낸다. 주 개발 환경인 호스트 컴퓨터는 IBM 486 PC이고 이것은 전역 (Global) 버스를 통하여 실시간 제어기들과 정보를 교환한다. 실시간 제어기들은 각기 국부 (Local) 버스를 가지고 있어 메모리 및 각종 센서와 구동기를 조작한다.

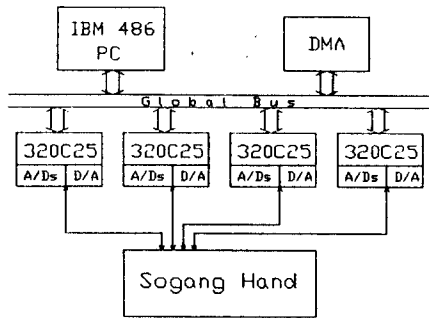


그림 8. 서강 손 제어기의 구조

A. 호스트 컴퓨터

호스트 컴퓨터는 작업 경로 계획 및 역기구학 방정식을 풀며, 최적 접촉력을 계산하게 된다. 또한 상위 수준 언어 (High-level language)를 사용하여 제어 알고리즘을 개발할 수 있는 환경을 제공한다. 그래픽 기능을 이용하여 경로 계획 및 위치 제어, 그리고 접촉력 계산의 모의실험 (Simulation)도 가능하다.

B. 실시간 제어기

실시간 제어기는 TI사의 320C25 DSP와 기본 작업용 및 제어 알고리즘 탑재용 메모리, A/D 및 D/A 변환기로 구성된다. 또한 호스트 컴퓨터와의 통신을 위한 전역 메모리를 가진다. 호스트 컴퓨터는 작업 경로점과 필요한 접촉력을 계산하여 해당 실시간 제어기의 전역 메모리로 전송한다. 실시간 제어기는 이것을 읽어 그에 해당되는 위치로 손가락을 이동시키거나 물체를 잡는데 필요한 접촉력을 발생시킨다. 서로간의 통신은 인터럽트에 의한 메시지 전달 (Message passing) 방식에 의해 이루어진다.

C. Direct Memory Access (DMA)

실시간 제어기의 제어 알고리즘 탑재용 메모리는 RAM으로 구성되어 있다. 이것은 제어 알고리즘의 수정 및 변경을 용이하게 하기 위함이다. 이때, 알고리즘은 호스트 컴퓨터의 메모리나 디스크로부터 DMA 제어기에 의해 곧바로 실시간 제어기의 메모리로 전송될 수 있다. 이것은 데이터의 전송속도를 높여준다. 또한 호스트 컴퓨터는 자신의 메모리에 데이터를 읽고 쓰는 방식으로 전송 가능하다.

2. 제어기의 소프트웨어

서강 손의 제어 시스템은 작업 관리 (Job Scheduling) 프로세스, 제어 (Control) 프로세스, 그리고 통신 (Communication) 프로세스로 구성된다.

A. 작업 관리 프로세스

동력학 방정식을 풀거나 센서로부터의 정보를 읽는 것은 주기적으로 이루어져야 한다. 이 프로세스는 처리해야 할 작업과 그 주기 및 소요시간을 호스트 컴퓨터로부터 전달 받고 그 요구조건에 맞게 작업량과 시간을 조절한다. 여러 작업이 요구될 때의 우선순위는 작업 주기가 가장 작거나 작업 처리 시간이 가장 짧은 것 부터 먼저 실행시킨다. 이 프로세스는 제어기 모든 프로세스를 관리한다.

B. 제어 프로세스

여기에는 위치 제어와 힘 제어 알고리즘, 센서로부터 정보수집이 포함된다. 이 프로세스는 서강 손의 제어를 위한 핵심이 되는 부분이며 작업 관리 프로세스로부터 가장 높은 우선순위를 받는다.

C. 통신 프로세스

호스트 컴퓨터와 실시간 제어기 사이의 정보교환을 담당한다. 정보는 메시지 전달 방식으로 이루어지며, 정보교환의 시작은 인터럽트에 의해 발생한다. 메시지의 형식은 실시간 제어기의 고유 ID와 정보의 종류로 구성되는 헤더와 정보의 내용으로 이루어진다.

IV. 결 론

본 논문에서는 다지 다관절 손의 기구학적 구조와 동력 전달 구조, 위치 및 tension 측정 센서, 그리고 시스템의 제어기 구성에 관하여 논하였다.

서강 손은 일반적인 end effector와는 달리 다양한 물체를 조작할 수 있도록 설계하였다. 또한 각 관절의 구동기를 로봇 손의 외부에 설치하게 하여 기계적인 구조도 간단하고 관절의 응답성도 높일 수 있다.

관절의 위치의 알아내기 위한 홀 효과 센서는 [1]에서와 같이 tendon의 tension을 측정하여 간접적으로 얻는 방법이 아니라, 관절에서 직접 측정하기 때문에 그 신뢰성이 크다. 또한 tendon의 tension도 관절에 인접한 곳에서 측정함으로써 잡음에 의한 영향을 줄일 수 있다.

서강 손의 제어 시스템은 다중 프로세서 구조로 되어있어 실시간 제어를 가능하게 해준다. 호스트 컴퓨터로 IBM PC를 사용하기 때문에 제어 규칙의 개발이나 수정이 용이하다.

[8] Narasimhan, S., Siegel, D.M., Hollerbach, J.M., "CONDOR: An Architecture for Controlling the Utah-MIT Dexterous Hand," *IEEE Transactions on Robotics and Automation*, Vol. 5, No. 5, Oct. 1989, pp. 616-627.

[9] R.P. Paul and H. Zhang, "Design of a robot force/motion server," in *Proc. IEEE Conf. on Robotics and Automation*, San Francisco, CA, Apr. 1986, pp. 1878-1883.

참고 문헌

[1] Salisbury, J.K. & Craig, J.J. "Articulated Hands: Force Control and Kinematic Issues," *International Journal of Robotics Research*, Vol. 1, No. 1, Spring 1982, MIT Press.

[2] 백주현, 정낙영, 서일홍, 최동훈, "손마디 접촉을 고려한 다지 다관절 로봇손의 최적 접촉력 결정 방법," 전기학회논문지, 제40권, 제8호, 1991년 8월.

[3] Salisbury, J.K. "Design and Control of an Articulated Hand," *1st International Symposium on Design and Synthesis*, Tokyo, July 1984.

[4] S. Hirose & S. MA, "Coupled Tendon-driven Multijoint Manipulator," *IEEE International Conference on Robotics and Automation*, Sacramento, CA, Apr. 1991, pp. 1268-1275.

[5] S.C. Jacobsen, J.E. Wood, D.F. Knutti, K.B. Biggers and E.K. Iversen, "The Utah/MIT Dexterous Hand: Work in Progress," *The International Journal of Robotics Research*, Vol. 4, No. 3, 1984, pp. 21-50.

[6] S.C. Jacobsen, E.K. Iversen, D.F. Knutti, R.T. Johnson, and K.B. Biggers, "Design of the Utah/MIT Dexterous Hand," in *Proc. IEEE Int. Conf. on Robotics and Automation*, San Francisco, CA, Apr. 1986, pp. 1520-1532.

[7] C.S.G. Lee, "Robot Arm Kinematics, Dynamics, and Control," *IEEE Computer*, Vol. 15, No. 12, Dec. 1982, pp. 62-80.