

퍼지 제어를 이용한 여유자유도 로봇 팔의 장애물 우회에 관한 연구

황재석*(울산대 대학원 기계자동차공학부), 박찬호(울산대 대학원 기계자동차공학부),
이병룡(울산대 기계자동차공학부), 양순용(울산대 기계자동차공학부),
안경관(울산대 기계자동차공학부)

Study on the Collision Avoidance of a Redundant Robot Arm Using Fuzzy Control

J. S. Hwang(Mecha. Eng. Dept. UOU), C. H. Park(Mecha. Eng. Dept. UOU), B. R. Lee(Mecha. Eng. Dept. UOU), S. Y. Yang(Mecha. Eng. Dept. UOU), K. K. Ahn(Mecha. Eng. Dept. UOU)

ABSTRACT

In this paper, a motion control algorithm is developed using a fuzzy control and the optimization of performance function, which makes a robot arm avoid an unexpected obstacle when the end-effector of the robot arm is moving to the goal position. During the motion, if there exists no obstacle, the end-effector of the robot arm moves along the pre-defined path. But if there exists an obstacle and close to the robot arm, the fuzzy motion controller is activated to adjust the path of the end-effector of the robot arm. Then, the robot arm takes the optimal posture for collision avoidance with the obstacle. To show the feasibility of the developed algorithm, numerical simulations are carried out with changing both the positions and sizes of obstacles. It was concluded that the proposed algorithm gives a good performance for obstacle avoidance.

Key Words : Redundant Robot, Fuzzy Control, Collision Avoidance, Posture Optimization

1. 서론

현재 산업현장에서 사용되고 있는 로봇의 대부분은 용접작업, 도장작업 및 단순한 집기 및 놓기(pick and place) 작업에 이용되고 있다. 좀더 복잡한 작업환경이나 유연성(flexibility)이 필요한 작업환경에서 로봇을 사용하려면 예측할 수 없는 장애물에 대해서도 우회 성능을 갖는 것이 필요하다. 이를 위해 여러 가지 센서들을 조합하여 로봇의 움직임을 지능화하는 방법들이 최근에 많이 연구되고 있다. 로봇이 작업 영역을 이동할 때 발생하는 장애물을 우회하는데 사용되는 경로 계획법은 크게 2가지로 구분할 수 있다. 하나는 상태 공간법(configuration space method)⁽¹⁾⁽²⁾이며, 다른 하나는 인공전위장법(artificial potential field method)⁽³⁾⁽⁴⁾이다. 상태공간법은 로봇의 관절 운동이 n-차원의 상태공간에서 점의 운동으로 변환되며, 장애물 또한 상태공간에서 장애물 영역으로 치환된다. 따라서 장애물과 충돌이 발생하지 않는 안전한 경로를 구할 수

있게 되며, 이 경로로부터 각 관절의 회전각을 구하는 방법이다. 이 방법은 계산 시간이 많이 들어 적용에 한계가 있다. 인공전위장법은 인공적인 전위장 함수를 이용하는 방법인데 장애물이 있는 위치는 전위장값을 높게 할당하며, 로봇이 도달해야 하는 목표 위치는 전위장값을 최소값으로 할당해서 로봇이 목표위치에 도달하게 하는 방법이다. 그러나 이 방법은 이동용 로봇에서는 쉽게 적용이 가능하나 관절형 로봇에 적용하기는 매우 어렵다. 또한, 로봇이 목표를 찾아가는 도중에 지역최소점(local minimum)⁽⁵⁾에 빠지는 경우가 발생하기도 한다.

본 논문에서는 퍼지제어와 로봇팔의 자세최적화 함수를 이용하여 돌발적으로 발생하는 장애물을 우회할 수 있는 여유자유도 로봇팔의 장애물 우회 알고리즘을 개발하였다. 로봇의 이동 중에 장애물이 존재하는 경우 장애물의 크기 및 위치정보가 로봇 제어 시스템에 전달되면, 퍼지 운동제어기(fuzzy motion controller)는 로봇 팔 끝점의 이동방향을 결정해 준다. 그 다음 자세최적화 알고리즘을 이용하

여 장애물과의 충돌위험을 최소화하고 부드러운 움직임을 유지할 수 있는 최적화자세를 결정하게 된다. 이와 같은 과정을 반복하여 로봇은 목표위치에 도달하게 된다. 제안된 퍼지 운동제어기와 자세최적화 알고리즘의 타당성을 보여주기 위하여 장애물의 위치 및 크기를 바꾸어 가면서 다양한 전산모의 실험을 수행하였다.

2. 여유자유도 로봇팔의 장애물 우회 알고리즘

여유자유도 로봇은 작업을 수행하기 위하여 필요한 자유도(degree of freedom)보다 더 많은 자유도를 가지고 있는 경우를 말한다. 어떤 로봇 팔의 끝위치가 주어질 때 끝위치를 만족하는 로봇 팔의 자세는 무수히 많이 존재하게 된다. 따라서, 이러한 여유자유도를 이용하여 원하는 로봇의 작동요소를 최적화할 수 있다는 장점이 있다. 예를 들어 로봇 팔의 작동 중에 소요되는 에너지를 최소화시킨다든지, 특이점(singular point)을 피한다든지, 원하는 궤적을 따라서 속도 및 힘의 전달율(transmission ratio)을 최대화할 수 있는 등의 작업을 수행할 수 있다. 본 논문에서는 여유자유도를 가진 로봇이 움직일 때 돌발적으로 발생하는 장애물을 최적의 자세를 유지하면서 우회할 수 있는 알고리즘을 개발하고자 한다.

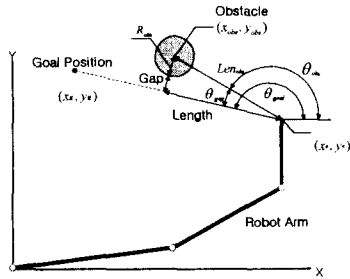


Fig. 1 Parametric Relations between a Robot Arm and an Obstacle

2.1 퍼지(Fuzzy) 운동제어기

로봇의 이동 중에 돌발적인 장애물이 존재하는 경우, 장애물의 크기 및 위치정보가 로봇제어 시스템에 전달되면, 퍼지 운동제어기(fuzzy motion controller)는 로봇 팔 끝점의 이동방향을 결정해 준다. Fig.1 는 로봇이 움직일 때 돌발적으로 발생한 장애물이 로봇 팔의 끝점의 진행방향 가까이 놓여져 있는 모습을 보여주고 있다. 퍼지제어를 이용하기 위하여 필요한 입력변수는 Gap 과 Length 로 선정하였다. Gap 은 로봇 팔의 끝점(end-effector)이 진행방향의 선상으로부터 장애물의 법선방향으

로 떨어진 거리로 정의되고, Length 는 장애물이 로봇 팔의 끝점에서 로봇 팔의 진행방향으로 떨어져 있는 거리로 정의된다. Fig.1 에서 $\theta_{goal}, \theta_{obs}$ 그리고 Len_{obs} 는 다음과 같이 구해 진다.

$$\theta_{goal} = \tan^{-1}\left(\frac{y_g - y}{x_g - x}\right) \quad (1)$$

$$\theta_{obs} = \tan^{-1}\left(\frac{y_{obs} - y}{x_{obs} - x}\right) \quad (2)$$

$$Len_{obs} = \sqrt{(x_{obs} - x)^2 + (y_{obs} - y)^2} \quad (3)$$

$$\theta_{gap} = \text{sgn}(x_{goal} - x_{start})(\theta_{goal} - \theta_{obs}) \quad (4)$$

퍼지제어기의 입력변수 Gap 과 Length 는 다음과 같은 식으로 표시된다.

$$Gap = Len_{obs} \cdot \sin \theta_{gap} - R_{obs} \quad (5)$$

$$Length = Len_{obs} \cdot \cos \theta_{gap}$$

그리고, 퍼지제어에 사용된 출력변수는 $\Delta\theta$ (로봇 팔의 끝점의 이동방향 수정각)로 선정하였다. 입력변수의 퍼지 세트 Gap 은 NB(Negative Big), NS(Negative Small), ZE(Zero), PS(Positive Small), PB(Positive Big)으로 선정하였고, Length 는 ZE(Zero), PS(Positive Small), PM(Positive Medium), PB(Positive Big)으로 선정하였다. 출력변수에 대한 퍼지세트 $\Delta\theta$ 는 ZE(Zero), PS(Positive Small), PM(Positive Medium), PB(Positive Big), PVB(Positive Very Big)으로 선정하였다. 소속함수(fuzzy membership function)는 Fig.2 에 나타나 있다.

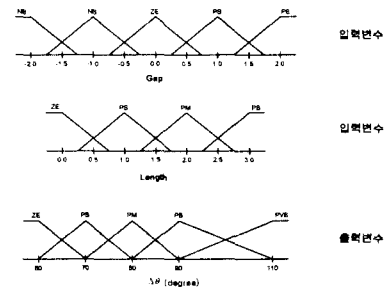


Fig. 2 Membership Function of Input and Output Variables

입력 및 출력변수의 퍼지세트에 대한 퍼지 제어 규칙인 FAM(Fuzzy Associative Memory)은 Table1 에 나타나 있다. FAM 이란 입력변수와 출력변수의 퍼지 세트간의 논리적으로 타당한 IF~THEN 규칙을 함축적으로 나열한 것이라고 볼 수 있다. 예를 들어 FAM 의 좌측상단의 블록을 달리 표현하면 다음과 같은 논리규칙으로 표시된다.

IF Length is ZE and Gap is NB,
THEN $\Delta\theta$ is PVB.

그리고, 퍼지제어기는 FAM 을 이용하여 임의의 입력변수가 입력되었을때, 비퍼지화(defuzzification method)⁽⁶⁾ 방법을 이용하여 명확한 출력 값을 제공하게 된다.

		Length			
		ZE	PS	PM	PB
Gap	NB	PVB	PVB	PVB	PB
	NS	PVB	PVB	PB	PB
	ZE	PVB	PB	PB	PM
	PS	PB	PB	PM	PS
	PB	PB	PM	PS	ZE

Table 1. Fuzzy Associative Memory (FAM)

2.2 여유자유도 로봇팔의 자세최적화 알고리즘

여유자유도를 갖는 로봇 팔의 속도는 높은 차원의 관절공간(joint space)에서 낮은 차원의 작업공간(task space)으로 식 (6)와 같이 치환(mapping)이 된다.

$$\dot{X} = J(\Theta)\dot{\Theta} \quad (6)$$

여기서, $X=[x_1, x_2, \dots, x_m]^T$ 는 작업좌표 벡터이고, $\Theta=[\theta_1, \theta_2, \dots, \theta_n]^T$ 는 관절좌표 벡터이며, J 는 $m \times n$ 자코비안 행렬이다. 식 (6)에 대한 역기구학 식은 식 (7)과 같이 구해진다.

$$\dot{\Theta} = J^+ \dot{X} + (I - J^+ J)K \quad (7)$$

식 (7)의 내부운동 중에서 로봇 팔이 원하는 성능함수(performance criterion)를 최대화 시켜 주도록 벡터의 값을 결정하여야 한다. 성능함수를 라고 했을 때, $P(\Theta)$ 를 최대화하기 위한 벡터 K 의 관계는 식 (8)와 같이 구해진다.⁽⁷⁾

$$K = \frac{\partial P}{\partial \Theta} k \quad (8)$$

식(8)에서 $\partial P / \partial \Theta = [\partial P / \partial \theta_1, \partial P / \partial \theta_2, \dots, \partial P / \partial \theta_n]^T$, 그리고 k 는 임의의 양의 상수이다. 따라서, 식 (8)의 관계가 유지되게 하는 내부운동(internal motion)은 항상 $P(\Theta)$ 를 증가시키는 방향으로 움직인다는 것을 의미한다. 본 논문에서는 로봇 팔이 이동 중에 발생할 수 있는 장애물에 충돌하지 않고 안전하게 우회하여 목적지에 도달하는 것이 중요한 요소이므로, 장애물이 존재하고 로봇 팔의 끝점의 위치가 결정된 경우에 로봇 팔의 각각의 링크(link)가 Fig.3 에서 보는 바와 같이 장애물로부터 가능한 멀리 떨어져 있는 자세를 취하도록 성능함수를 설정하였다.

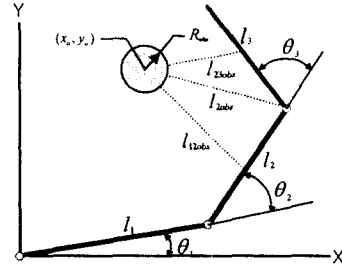


Fig. 3 Configuration of a Robot Arm and an Obstacle for Performance Criterion

$$P(\Theta) = C[C_{12}(l_{12obr} + \beta_{12}l_{2obr}) + C_{23}l_{23obr} + C_{23}(l_{23obr} + \beta_{23}l_{2obr}) + C_{det}\beta_{det}\theta_2 + C_{det}\beta_{det}\sqrt{\det(JJ^T)}] \quad (9)$$

식 (9)에서

$$l_{2obr} = [(l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2) - O_x)^2 + (l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2) - O_y)^2]^{1/2} - R_{obr}$$

$$l_{12obr} = \left[\left(l_1 \cos \theta_1 + \frac{l_2 \cos(\theta_1 + \theta_2)}{2} - O_x \right)^2 + \left(l_1 \sin \theta_1 + \frac{l_2 \sin(\theta_1 + \theta_2)}{2} - O_y \right)^2 \right]^{1/2} - R_{obr}$$

$$l_{23obr} = \left[\left(l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2) + \frac{l_3 \cos(\theta_1 + \theta_2 + \theta_3)}{2} - O_x \right)^2 + \left(l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2) + \frac{l_3 \sin(\theta_1 + \theta_2 + \theta_3)}{2} - O_y \right)^2 \right]^{1/2} - R_{obr}$$

그리고, 식 (9)의 오른쪽 4 번째 항은 2 번째 관절 각 θ_2 가 0 도 이하로 되는 경우, 즉 하관절(lower elbow)에서 상관절(upper elbow)의 형태로 로봇팔이 바뀌는 것을 방지하기 위하여 첨가하였고, 5 번째 항은 로봇 팔의 동작 중에 부드러운 운동, 즉 조작성(manipulability)를 높이기 위하여 첨가되었다.

3. 전산모의실험

성능을 확인하기 위하여 몇 가지 전산 모의실험을 수행하였다. 모의실험을 위하여 사용된 3 관절 로봇 팔의 크기는 $l_1 = 60$, $l_2 = 40$, $l_3 = 25$ 이다. 식 (9)에서 사용된 계수들은 $C=0.01$, $\beta_{12} = 0.5$, $\beta_{23} = 0.5$, $\beta_{det} = \pi/180$, $\beta_{det} = 0.01$ 과 같이 선정되었다. 모의실험을 하기 위한 입력변수는 초기위치에서의 로봇 팔의 끝점 x, y 좌표, 초기위치에서 로봇 팔의 3 번째 링크의 각도, 장애물 x, y 좌표와 반경, 그리고 목표 위치에서 로봇 팔의 끝점 x, y 좌표이다. 장애물의 형상은 원의 형태로 가정하였으며 장애물의 크기와 위치는 원의 반경과 중심의 위치로 표현하였다. 그리고 장애물의 위치와 로봇 팔의 끝점의 간격이 장애물 직경의 5 배 이내로 가까워졌을 때 퍼지제어기가 작동하여 로봇 팔의 끝점의 진행 방향을 수정하도록 하였고, 이때부터 퍼지제어와 동시에 로봇 팔의 최적화 자세를 찾는 과정을

수행하였다. 장애물의 위치에 따른 로봇 팔의 우회 실험 결과는 Fig.4 에서 Fig.7 과 같다.

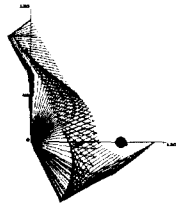


Fig.4 Simulation when the obstacle is close to start position: Start Position: (110,0,30), Goal Position: (0,110), Obstacle: (80,0,5)

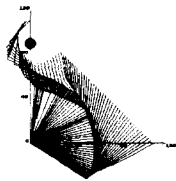


Fig.5 Simulation when the obstacle is close to the goal position: Start Position: (110,0,30), Goal Position: (0,110), Obstacle: (0,90,5)

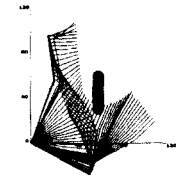


Fig.6 Simulation when the obstacle is moving : Start Position: (110,20, 45), Goal Position: (40,110), Initial Obstacle: (60,60,5)

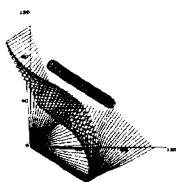


Fig.7 Simulation when the obstacle is moving : Start Position: (110,0, 30), Goal Position: (0,110), Initial Obstacle: (70,40,5)

4. 결론

본 논문에서는 여유자유도를 갖는 수평 다관절 로봇이 초기 위치에서 목표 위치까지 정해진 궤적을 따라 이동할 때 돌발적으로 발생할 수 있는 장

애물을 퍼지제어와 자세 최적화 기법을 이용하여 장애물을 우회할 수 있는 운동제어알고리즘을 개발하였다. 작업영역에서 로봇이 움직일 때 장애물이 충돌위험 영역으로 접근하게 되면 퍼지제어시스템이 작동하게 되어 로봇의 끝점의 이동 방향을 수정하게 하여 주어진 궤적을 이탈하게 하고 장애물을 우회하여 목표 위치에 도달하게 한다. 개발된 운동제어 알고리즘을 성능을 확인하기 위하여 다양한 모의실험을 수행하였다. 장애물의 크기와 위치를 변경하여 로봇이 장애물을 성공적으로 우회하는지 조사하였으며, 또한 이동장애물에 대해서도 적용하여 만족스러운 결과를 얻었다. 본 논문에서는 단순히 모의실험만으로 이루어졌으나, 비전 센서 (vision sensor)나 거리 센서 등을 이용하여 장애물의 크기나 위치를 실시간적으로 계산하게 되면 실제 작업에도 쉽게 적용이 가능하다고 본다.

후 기

본 연구는 과학재단 지원 울산대학교 지역협력 센터(RRC)지원으로 수행되었으며, 관계자 여러분께 감사 드립니다.

참고문헌

1. T.Lozaano-Perez, "Spatial Planning: a Configuration Space Approach", IEEE TRans. Comp, 1983, pp.108-120
2. T.Lozaano-Perez, "A Simple Motion-Planning Algorithm for General Robot Manipulators", IEEE Journal of Robotics and Automation, 1987, Vol.3, No.3, pp.224-238
3. O.Khatib, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots", Int. J. of Robotic Research, Vol.5, No.1, 1986, pp.90-98
4. J.Barraguang and J.C.Latombe, "Robot Motion Planning: A Distributed Representation Approach", Int. J. of Robotics Research, Vol.10, No.6, 1991, pp.628-649
5. 이병룡, "포텐셜함수를 이용한 자율주행 로봇들 간의 충돌예방을 위한 주행제어 알고리즘의 개발", 한국정밀공학회, 제 15 권, 제 6 호, 1998, pp.107-115
6. L.Wang, "Adaptive Fuzzy Systems and Control", Prentice Hall
7. L.Wang, "Adaptive Fuzzy Systems and Control", Prentice Hall, Yoshikawa T, "Analysis and Control of Robot Manipulator with Redundancy", Robotics Research: 1st Int. Sym, MIT Press, 1984, pp.735-747