

SOM(Self-Organization Map)을 이용한 다관절 로봇의 충돌회피 경로설계

이종우, 오석찬, 이종태

동국대학교 산업공학과

ABSTRACT

최근 몇년사이에 산업 전분야에서 로봇의 이용이 증가하고 있는데, 로봇 시스템의 주요목적은 작업영역내에서 작업물을 빠르고 정확하게 다른 장소로 이동시키는 것이다. 이러한 로봇의 이용에 있어서의 어려움 중 하나는 로봇이 목표점으로 움직이는 동안에 작업장내에 있는 장애물, 즉 각종 공구, 시설 등의 물체와의 충돌을 피할 수 있도록 프로그램 되어야 하며, 이를 위해서 많은 시간이 소요된다는 것이다.

본 연구에서는 SOM 네트워크를 이용하여 장애물이 존재하는 작업 공간에서 로봇이 장애물과 충돌없이 움직일 수 있는 경로를 구하기 위한 SOM의 응용방안을 소개한다. 본 연구에서는 SOM의 최적 size, 학습계수 요인을 고려하여 2관절 로봇의 충돌회피 경로 발견을 위한 시뮬레이션을 수행하였다.

1. 서론

로봇 시스템에서 매니플레이터의 목표점을 알려 주었을 때, 시스템이 로봇에서 경로상의 중간점들의 위치와 갯수를 매니플레이터가 장애물에 걸리지 않도록 결정할 수 있다면 편리할 것이다. 이를 위하여, 시스템은 매니플레이터 작업 영역 그리고 영역안의 모든 잠재적 장애물에 대한 모델을 가지고 있어야 한다. 이 장애물이 존재하는 곳에는 로봇과 장애물이 서로 충돌을 일으키지 않는 경로를 찾아야 한다는 점과 그 경로가 다른 가능한 경로보다 짧아야 한다는 두가지 점에서 고려되어야 한다.

충돌없는 경로계획을 수행하는 방법으로 제

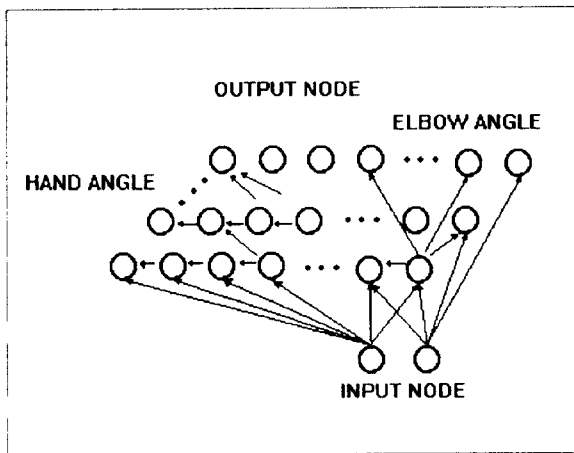
시되었던 것은 자유공간을 연결된 도형으로 나타내어, 충돌없는 경로를 찾는 것이었다. 그러나, 이것은 로봇 관절수에 지수적으로 복잡성이 증가하게 된다. 여러 방법에 의한 충돌회피 방법은 계산상의 복잡성으로 인해서 실시간에 처리되지 못하고 있다.

본 연구에서는 로봇 작업장에서 제시된 장애물에 대해서 충돌회피 경로를 찾을 수 있는 SOM을 설계하는 데 목적이 있다. SOM은 임의의 작업장에서 개별 end point 위치에 해당하는 유효점들을 입력받아 자기 조직화 하고, network내의 weights의 변화를 근거로 장애물을 인식하고 최단 경로를 설계할 수 있게 된다. 본 연구에서는 이러한 OM

의 최적 size, 학습계수, 최적 충돌회피 경로 등을 제시하고자 한다. 이를 위해서 여러 변동요인들을 고려한 시뮬레이션을 수행하고 그 성능을 시각적으로 평가하였다.

2. 본 론

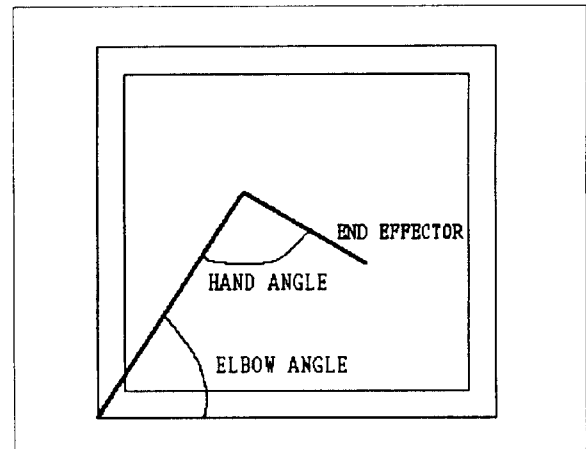
2.1 SOM Network의 구성



[그림 1] SOM NETWORK

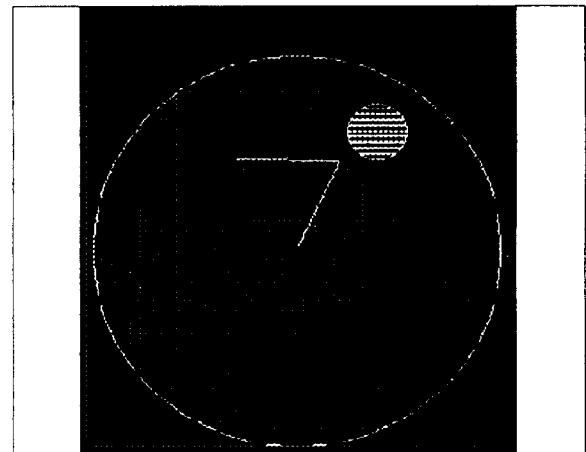
[그림 1]에서 보는 것같이 이 Network는 input값이 2차원인 SOM network로 이루어져 있다.

SOM network는 2개의 input node와 다수의 2차원 output node, 그리고 각 input node와 output node를 연결하는 weight로 구성된다.



[그림 2] ROBOT ARM

SOM network의 2차원 input값은 [그림 2]에서와 같이 Euclidean상의 로봇 팔의 끝점을 HAND ANGLE 과 ELBOW ANGLE, 2개의 각도로 치환한 것이다.



[그림 3] Simulation 프로그램의 로봇 팔

SOM Network는 [그림 3]에서와 같이 작업장내에서 로봇 팔의 end point가 갈 수 있는 임의의 점(각도로 치환된 점)을 입력받아, 이 입력값에 적용하면서 자기조직화를 수행한다. SOM Network의 자기조직화는 input node와 output node를 연결하는 weight를 변화시킴으로써 이루어진다. [그림3]에서 조그만 원은 장애물을 나타낸다.

input node와 output node 사이의 weight는 다음의 식에 의해 adaptive 하게 결정된다.

$$W_{X,ij}(t+1) = W_{X,ij}(t) + \eta \times (I_X - W_{X,ij}(t)) \times e^{-((I_X - W_{X,ij}(t))^2 / \sigma)} \dots (1)$$

$$W_{Y,ij}(t+1) = W_{Y,ij}(t) + \eta \times (I_Y - W_{Y,ij}(t)) \times e^{-((I_Y - W_{Y,ij}(t))^2 / \sigma)} \dots (2)$$

여기서 $W_{X,ij}$, $W_{Y,ij}$ 는 각각 input node x 와 node ij 및 input y 와 node ij 사이의 weight 값, I_X, I_Y 는 각각 input의 X값과 Y값을 나타낸다.

여기서 $I_Y - W_{Y,ij}(t)$ 는 SOM network의 input값과 output값사이의 거리를 나타내며, parameter $\sigma = 10000$ 으로 주어져 있다. 식 (1), (2)는 지수분포의 성질상 가까운 거리에 있는 weight에 대하여 더 큰 가중치를 주고 있음을 나타내고 있다. 이 때 최적의 Network size 를 구하기 위해서는 Neural Network의 특성상 weight 의 갯수와 최적 학습계수의 선정이 무엇보다 중요하지만, 이를 구할 수 있는 방법이 없기 때문에 여러 대안에 대해서 Simulation을 수행해야 한다.

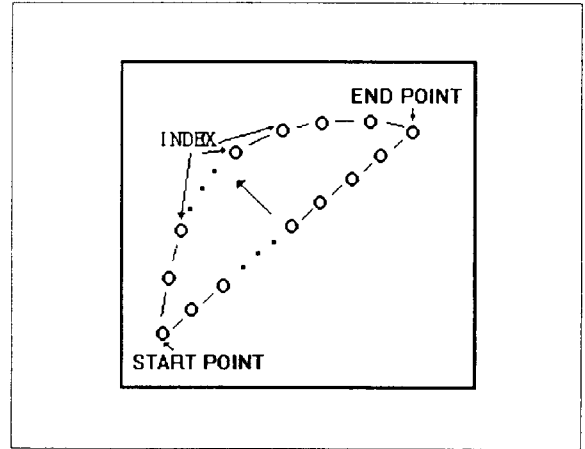
SOM Network는 로봇팔이 움직일 수 있는 작업장범위내에서도 충돌이 일어나지 않는 영역을 학습해야 하므로, 일단 로봇의 위치를 나타내는 input값이 들어오면 이 값이 장애물과의 충돌이 없는 즉, 유의한 점인지를 검사하고, 유의한 경우 input값으로 받아들여 조직화한다. 본 연구에서는 작업장로봇 팔의 최대 운동각도는 1축은 30°에서 330°까지, 2축도 30°에서 330°까지로 제한하였다.

학습이 끝나면 SOM network는 작업장내에서 로봇 팔의 start point와 end point를 입력받아 그 사이에서 충돌이 일어나지 않는 경로를 찾을 수 있게 된다.

SOM network는 1차 SOM network와 1차 SOM network로 구성되어 있다. 1차 SOM

network는 장애물없이 유효한 작업영역만을 학습하는 단계이다. 2차 SOM network는 1차 SOM network에서 이미 학습된 영역에서 장애물을 고려하여 유효한 작업 영역을 학습시키는 단계이다.

2.2 최단충돌회피경로 설계 알고리즘



[그림 4] WORMBODY

1차 SOM Network 와 2차 SOM Network 을 통해서 장애물을 인식한 경로들을 얻을 수 있다. 이 경우 많은 경로 중에서 start point 에서 출발하여 end point 까지 가는 최단 경로를 찾아야 한다. 이를 위해서 여기서 WORMBODY 의 개념을 사용했다.

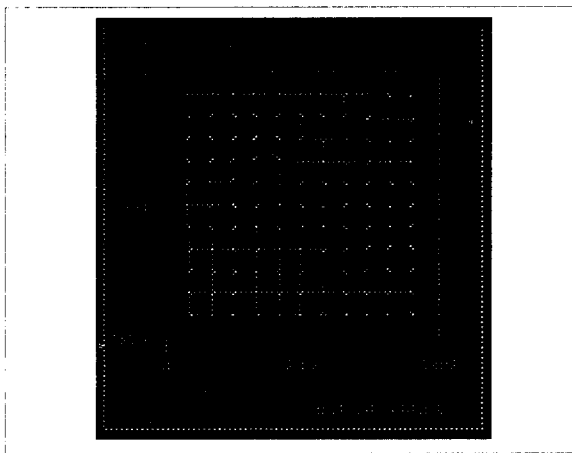
WORMBODY는 start point와 end point 사이를 잇는 경로를 나타내는 데, 이

WORMBODY는 start point와 end point를 잇는 직선 경로로서, 만약 충돌이 발생되지 않았다면 최단으로 갈 수 있었던 경로이다. WORMBODY값은 두 점을 잇는 index들의 선형 list로 표현된다. 이 index는 2차 SOM에서 장애물과의 충돌을 피한 형태로 위치 벡터로 생각한다. 2차 SOM에서의 이들 index가 지정하는 node의 위치를 연결하면 충돌회피 경로가 된다. [그림 4]는 학습되기 전에는 직선이었던 WORMBODY가 학습 후에는 곡선으로 변한 모습을 나타내고 있다.

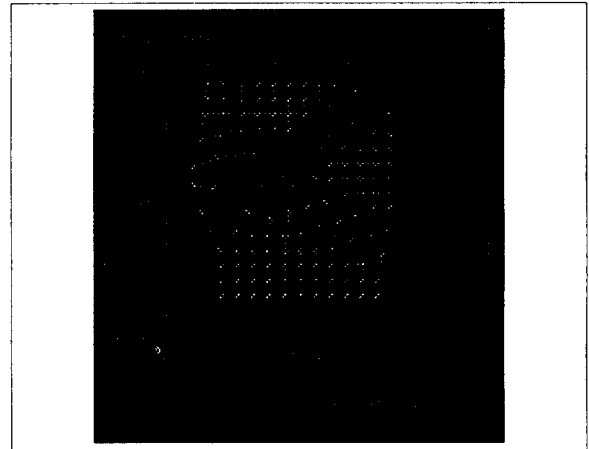
2.3 최적 SOM Network Structure 의 Parameter를 찾기 위한 Simulation

Neural Networks를 여러 문제영역에 적용하려는 시도들이 최근에 활발히 이루어지고 있다. 그러나 대부분의 경우, 그러한 시도과정에서 Network의 최적 size를 결정해야 하는 문제가 발생한다. 그러나 문제영역에 따르는 Neural Network의 성질상 이러한 최적 size를 정하기 위한 일반적인 방법이 존재하고 있지 않고, 다만 여러 실험을 거쳐 그 문제 영역에 맞는 Parameter를 찾아내어 Neural Network를 구성하는 방법밖에는 없는 것이 현재의 상황이다. Network의 size가 커지면 입력 data에 있을 수있는 잡음의 영향을 받아 오히려 오류가 커질 수있고, 반대로 size가 작아지면 입력값의 패턴을 잘 반영하지 못하는 단점이 있으므로 이들 사이의 절충이 필요하다. 학습계수 역시 그 값이 커지면 입력 data에 너무 민감하게 반응하고, 너무 적어도 Learning에 걸리는 시간이 오래 걸릴 수가 있다. 여기서는 Simulation을 통하여 output node 의 갯수는 100이 되고, 학습계수 η 가 0.0004일때가 최적임을 알아내었다.

3. 결 론



[그림 5] 학습전의 Simulation 준비 그림



[그림 6] 학습후 Simulation 결과 그림

[그림 5], [그림 6]에서 C 언어로 짠 프로그램으로 simulation을 한 결과 SOM Network로 장애물에 대한 충돌회피 경로들이 발견할 수 있는 SOM Network size와 각 Parameter들을 찾아 낼 수 있었다. 이 경로중에서 WORMBODY 을 이용하여 경로를 찾기 위한 index 값을 찾아 낼 수 있었다.

4. 추후 연구방향

보다 현실적인 로봇 운영상황에서는 작업장의 장애물이 미리 주어진 경우가 아니라 작업도중 발생하는 경우, 그 장애물에 대한 충돌회피경로를 실시간에 찾아야 하는 문제가 발생한다.

이 경우 SOM network의 자기조직화를 얼마나 신속히 할 수 있는가 하는 것이 실제로의 충돌회피경로 발견의 관건이 되어 이는 차후 연구과제가 될 것이다.

5. 참고 문헌

[국내문헌]

1. 김대수, "신경망 이론과 응용(I)", 하이테크정보, 1993
2. 김태운, 김홍복 공저, "메카트로닉스 산업용 로봇", 생능사, 1991

3. 남궁재찬, "ROBOT 공학의 기술", 기전연구사, 1990
4. 김주필, 최후곤, "다관절 로봇트를 위한 충돌회피경로 및 제어프로그램 생성", 추계산업공학회 발표논문집, 1994
5. 이상원, "학습하는 기계 신경망", Ohm사, 1993

[국외문헌]

1. James A. Freeman , David M. Skapura , "Neural Networks, Algorithms, Applications, and Programming Techniques", ADDISON-WESLEY, 1991
2. John J. Craig, "Introduction to Robotics and Control, Second Edition" ADDISON-WESLEY, 1990
3. Robert Hecht-Nielsen, "Neurocomputing ",ADDISON-WESLEY, 1990
4. Bill Horne, M. Jamshidi & Nader Vadi e, "Neural Networks in Robotics: A Survey" , J. Intelligent and Robotic System, Vol. 3, p51-66, 1990
5. San-Yih Pan, Jackson C. S. Yang, Gui-Zhong Qi, "A Neural Network Methodology to Solve Robotic Manipulator's Inverse Kinematic Problem", World Congress on Neural Network, Vol 3, P194-197, 1993
6. J. Heikkonen, P. Koikkalainen, E. Oja, " Self-Organizing Maps for Collision-free Navigation", World Congress on Neural Network, Vol 3, P141-144, 1993