

신경회로망을 이용한 반사적인 무인차 제어

김유석, 이장규

서울대학교 공과대학 제어계측공학과, 자동화공동연구소

Reflexive Autonomous Vehicle Control Using Neural Networks

Yoo Seok Kim, Jang Gyu Lee

Automation and Systems Research Institute & Dep. of
Control and Instrumentation Eng., Seoul Nat'l Univ.

ABSTRACT

In this paper, we have shown a new approach of neural networks for mobile robot motion control under an indoor refracted environment. The vehicle has two powered wheels and four passive casters which support a free motion. And it also uses sonar sensors, infrared sensors, internal odometer, and contact sensors. Two experiments were conducted to demonstrate our objectives. The first one is that the vehicle executes a reflexive motor control to maintain a constant distance to the boundary. The second one is that as well as the boundary following, the vehicle makes a block obstacle avoidance during its path. Without prior knowledge of external environment, we have accomplished the tasks by employing a simple, reactive stimulus-response neural network scheme associating sensor data with the vehicle's action.

1. 서론

무인차(AGV)를 어떤 지점에서 목표지점까지 이동시키기 위해 자율항법시스템을 구현하는 일은 많은 경우를 고려해야 하며 특히 주어진 환경에 제한적인 센서정보를 적절히 활용해야 하는 등 어려운 문제를 포함하고 있다. 따라서 복잡한 상황에 대처하기 위한 지능적이고 자율적인 항법기법이 요구된다.

무인차 운동제이기법에는 크게 두가지 접근방법이 있다. 첫째, 주어진 환경의 정확한 표현(environment representation)과 그에 근간하는 주행상의 모든 경우에 대한 계층적이고 자세한 계획이 없는 지능적이고 자율적인 행위를 얻는 방법이 없다고 주장하는 것이다.[1][2] 둘째, 어떤 특정한 주변환경 모델에 독립적임을 강조하면서 무인차의 행위는 환경을 감지하는 센서들로부터의 자극에 반응하는 관점으로만 정의될 수 있으며 주행지도가 필요없다고 보는 것이다.[3][4][5] 현실적으로 복잡한 환경하에서 자세한 주변세계를 표현한다는 것은 어려워지며 발생할 수 있는 여러가지 경우를 고려하는 주행계획 등은 매우 비현실적이다. 특히, 동적인 환경하에서는 그와 같은 일이 불가능해지므로 그에

다른 계획은 소용이 없게 된다. 또한 생소한 환경에 놓인 무인차는 그 환경의 지식이 없으므로 적절한 주행을 하지 못하게 된다. 따라서 심리학과 생물학에서 지지를 받는 Brook의 소전제(subsumption) 제어구조가 반사적인 제어를 주장하는 사람들에게서 많은 관심을 받게 되었다.[6] 그런 방향에서 여러가지 방법들이 제시되었는데 본 논문에서는 그와 같은 센서정보와 무인차 제어를 연관짓는 지능적인 기법인 신경회로망기법을 이용하여 기초적인 주행을 실험해 보인다. 이 방법은 굴곡이 있는 경계를 일정한 간격을 두고 따라가기와 주행상에 나타난 단순한 블럭 장애물을 회피하는 일에 적용된다. 이와 같은 새로운 방법은 환경의 표현이나 선지식이 필요없는 단순한 기법으로 실시간제어(real-time control)를 수행하므로 무인차 운동제어에 적합한 방법으로 제시될 수 있다.

2. 무인차의 구성과 센서

실험에 사용된 무인차는 운동과 조정을 수행하는 두개의 구동 바퀴와 네개의 캐스터(caster)로 구성되었으며 폭은 750mm 길이는 700mm 높이는 280mm인 실내 실험실 주행용이다. 무인차에는 동력과 차량제어보드 그리고 센서제어보드가 내장되어 있다. 센서는 초음파센서와 적외선센서를 24개까지 부착할 수 있는데 실험에서는 6개의 초음파센서와 3개의 적외선센서만을 사용하였다. 또한 내부주행기가 있어 $\pm 32m$ 까지의 X, Y 좌표계 범위내에서 무인차의 위치와 현재의 방향정보를 알 수 있으며 그 정보들은 거리센서에 의해 수정되기도 한다. 그리고 접촉센서가 무인차의 앞뒤 범피내에 있어 장애물이나 사람과의 충돌을 감지한다. 무인차는 주컴퓨터(host computer)와 2개의 독립된 RS 232C를 통해 연결된 차량제어보드와 센서제어보드에 의해 제어된다. 차량제어 보드는 주어진 거리만큼 가라, 주어진 반경으로 주어진 각만큼 돌아라, 또는 명령된 회전속도로 움직여라 등과 같은 여러가지 명령을 주컴퓨터로부터 받아 모터를 제어한다. 센서제어 보드는 초음파센서를 스캔링(scanning)하고 그 값을 읽어 주컴퓨터가 요구할때 보낸다. 실제 실험에 있어서 초음파센서의 안정적인 측정가능거리는 약 140cm였다. 그리고 센서들의 정보가 가끔 정상치에

서 벗어나는 경우가 있어 한번의 거리측정에 여러번의 측정을 수행하여 일정한 임계치를 벗어나는 측정값은 제외하고 그외의 측정값으로부터 평균을 구하는 동적평균법(dynamic averaging)을 이용하여 센서의 부정확성을 보상한다. 또한 초음파센서의 정보를 유용하게 하기 위하여 센서감지시에는 센서와 면과의 각을 직각에서 크게 벗어나지 않도록 하였다. (약 25도) 적외선 센서는 물체의 존재 유무만을 감지할 수 있지만 직진성이 좋아 초음파센서가 갖는 단점을 극복할 수 있다.

3. 신경회로망제어기 모델

신경회로망의 여러가지 모델들이 제시되고 있으나 제어분야에 있어서는 실제 적용하기에는 아직도 많은 문제점들이 있다. 어떤 순간에 가장 최적의 제어출력이 무엇인지를 알기가 어렵다는 점과 복잡한 상황속에서 실시간으로 운동제어를 한다는 점이 난점으로 나타난다.

그러나 본 논문에서는 질신벌레 등의 화학적 주성에 따른 목적지향적인 운동성에 대한 단세포 생물의 생리학적인 모델과 자동장치이론(automata theory)의 개념들을 이용하여 무인차 운동 제어모델을 설계한다. [7][8][9] 무인차는 센서에서의 정보를 바탕으로 자유운동공간을 확보할 수 있으며 그 범위안에서 여러가지 방법으로 이동할 수 있다. 이와 같은 상황은 단세포생물의 환경과 유사한 속성이 많으므로 생물학적인 모델에서 제어기의 개념을 얻을 수 있다.

무인차가 굴곡이 있는 경계를 일정한 거리를 유지하며 따라가기 위해서는 경계와 수직인 상태에서의 거리를 감지하여야 한다. 따라서 무인차 운동제어는 경계와 평행을 맞추는 행위와 일정한 거리를 유지하기 위해 이동하는 위치조정행위, 주행상에 나타난 장애물을 피하기위한 회피행위, 그리고 위 과정들이 이루어졌을 때 전진방향으로 진행하는 직진행위 등으로 구성이 된다.

평행을 맞추는 과정에서 경계와의 거리를 측정하는 오른쪽의 두 초음파센서 값의 차이에 따라 오른쪽, 왼쪽으로 - 즉, 두 센서 값의 차이의 부호 - 각기 5개의 뉴론을 공간적으로 할당한다. 이것은 센서를 통한 정보표상(information coding)을 위한 것이며 할당된 뉴론들은 담당항 구역에 대한 위치표상(place coding)을 수행한다. 무인차가 지금 평행을 맞추기 위한 어떤 상태에 있을 경우, 해당하는 뉴론이 선택되어 그 가중치에 계수조정(scaling)이 되어 회전비율(무인차의 제어명령중 하나인 jog 모드에서의 turn rate)이 출력된다. 그 결과 상태가 출력전 보다 나빠졌으면 - 오른쪽의 두 센서 값의 차이가 더 커졌으면 - 그 출력에 가담한 뉴론의 가중치를 조정하고 좋아졌으면 그대로 나뉜다. 이와 같은 방식으로 각 뉴론들은 각각이 선택될 때마다 상태가 좋아지는 방향으로 출력을 내게되어 면과의 평행이 이루어지게 된다. 평행은 보통 두 센서 값의 차이가 3mm 이하로 맞추어지게 한다.

위치조정은 평행이 맞은후, 경계와의 일정한 거리를 유지하기 위해, 주행기를 이용하여 이동하는 행위이다. 여기서도 이동해야

하는 거리에 따라 - 오른쪽, 왼쪽 - 뉴론들이 할당되며 그 방식은 평행 담당 뉴론들의 경우와 같다. 이동은 이동해야 하는 거리의 반만큼을 간후 대칭으로 다음 반을 이동하는 S자형이 된다. 이때도 할당된 뉴론들이 선택되어 그 가중치 값에 계수조정이 되어 특정한 회전비율이 출력된다. 이 경우는 무인차가 경계와 큰 각으로 벌어지게 되므로 초음파센서의 정보가 의미가 없어진다. 따라서 이동전에 상황을 파악하여 이동 중이거나 이동 후에 경계와의 충돌의 가능성을 최소화 한다. 또 큰 거리를 이동해야 하는 경우는 시작점에서부터의 방향이 직각을 넘는 경우가 생기게 되므로 시작점에서 좌우 80도 까지로 이동 각을 한정한 후 그 이상의 이동은 직진으로 수행된다. 또한 이때는 직진 속도를 줄여서 앞으로 이동하는 폭을 적게한다. 이것은 움직이기 전에 자유공간을 확보했다 할지라도 센서의 시각지대나 예측 못한 장애물을 피하기 위해 불확실한 상황에서의 이동을 적게하기 위함이다. 위치조정에 있어서 선택된 뉴론의 가중치 조정은 평행의 경우와 마찬가지로 출력의 영향이 바로 전 상태보다 나빠졌으면 가중치를 조정하고 좋아졌으면 그대로 나뉜다. 가중치들은 담당항 거리에 비례하도록 0에서 1사이의 초기값을 갖는다.

다음은 위에서 설명한 각 뉴론들의 가중치 조정 알고리즘을 보인다.

$$W(t+1) = W(t) + C * |DELTA_RATE| * DELTA_DIS * W(t) \quad (1)$$

$$DELTA_RATE = \text{sigmoid}\{(|JOGRATE(t+1) - JOGRATE(t)| / (\max(|JOGRATE(t+1)|, |JOGRATE(t)|) + 1))\} \quad (2)$$

$$DELTA_DIS = \text{tanh}\{(DISTANCE(t+1) - DISTANCE(t)) / (\max(|DISTANCE(t+1)|, |DISTANCE(t)|) + 1)\} \quad (3)$$

$$JOGRATE(t) = C1 * W(t) * SIGN \quad (4)$$

여기서 W는 각 뉴론들의 가중치이고 JOGRATE는 명령된 회전(jog)에서의 회전비율인 deg/sec이다. 그 값은 선택된 제어모드에서 선택된 뉴론의 가중치 값에 이동하는 방향인 SIGN이 계수조정되어 나타난다. 보통 회전비율은 30 deg/sec이하에서 출력되었다. DISTANCE는 평행제어인 경우는 측면의 두 센서간의 측정거리값의 차이이고 위치조정제어인 경우는 이동거리를 나타낸다. 이와 같은 과정에서 센서값과 모든 거리값은 모두 mm 단위이다. DELTA_RATE는 양의 값을 갖게 되므로 0과 1사이의 값으로 시그모이드(sigmoid function) 함수를 이용하여 사상을 하였고, DELTA_DIS는 -1과 +1사이로 하이퍼탄젠트(hypertangent function) 함수를 이용하여 사상하였다. C, C1은 각각 평행제어와 위치조정제어에서의 가중치 조정계수이다.

이와 같은 기법에서 한 제어루프의 간격은 수백 msec이므로 가중치는 아주 조금씩 변화하게 되므로 전체적인 시스템은 안정화 된다. 즉, 무인차가 수행하는 제어가 정상적으로 이루어질 때 가중치의 변화는 거의 없게 되므로 전체적인 운동제어는 적절히 이루어진다. 신경회로망 제어기 모델의 구조는 그림 1과 같다.

거리센서나 내부주행기로부터의 정보가 생기면 제어모드에 따라 평행오차와 위치오차등으로 처리가 된 후 해당하는 신경회로

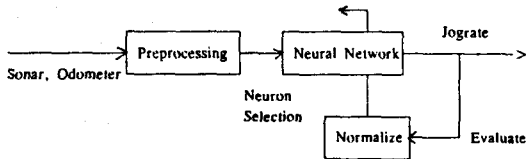


그림 1. 신경회로망제어기 모델 구조

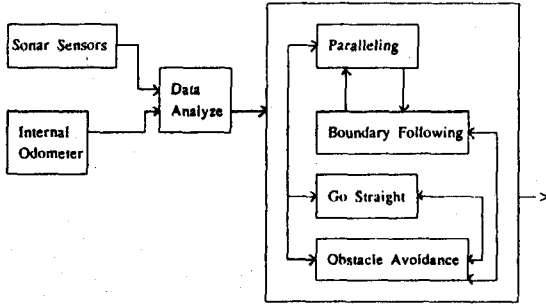


그림 2. 무인차 운동제어 구조

망의 뉴런이 할당되고 그 가중치와 계수조정을 통한 회전비율의 값이 출력된다. 제어행위가 가해진 후 전 상태와의 비교를 통해 해당하는 뉴런의 가중치조정이 이루어진다. 그림 2는 전체적인 제어구조를 보인다.

먼저 거리센서 정보로부터 평행(paralleling)을 수행한 후 장애물감지(obstacle detection)를 하면서 경계따라가기(boundary following)를 수행한다. 경계따라가기 후에는 반드시 장애물감지를 하면서 평행제어를 수행하고 평행과 경계따라가기 그리고 장애물회피(obstacle avoidance)가 모두 선택되지 않으면 직진모드가 수행된다. 장애물회피가 선택되면 회피동작을 수행하고 그 다음은 장애물을 통과하기 위해 직진모드와 경계따라가기, 평행제어 등이 이루어진다. 이와 같은 과정은 하나의 루프에서 매년 검사되어 수행되므로 무인차주변의 환경과 주행상태에 따라 선택되는 제어행위의 순차(sequence)는 다양하게 나타난다.

4. 신경회로망을 이용한 항법실험 결과

본 논문에서는 제어기모델에서 설명된 신경회로망을 이용하여 굴곡이 있는 경계를 일정한 거리를 유지하며 따라가는 일과 주행상에 나타난 장애물을 회피하는 일 등 두가지를 실험해 보인다.

먼저, 신경회로망을 이용한 평행행위 제어를 실험해 보았다. 초기의 오른쪽 두 센서값의 차이에 따라 그 차이에 비례하여 회전비율이 출력되어 점차 그 차이를 줄여가면서 면과의 평행을 조절한다. 이 과정에서 두 센서사이의 간격은 주변환경에 따라 달라져야 하는데, 가령 일정한 벽면이면 그 간격이 커도 상관이 없으나, 매우 불규칙하고 그 복잡성이 조밀할 경우 가능한 한도내에서 두 센서 사이의 배치간격은 작아져야 한다. 그러나 너무 작아지면 두 센서사이의 간섭문제가 야기될 수 있으므로 적당한 간격에서 만족해야 한다. 본 실험에서는 평행운동시 전진방향의 속도는 0으로 하여 제자리에서 평행을 맞추도록 하였는데 경우에 따라서는 약간씩 이동하면서 평행을 맞추는 것이 필요하다. 만약

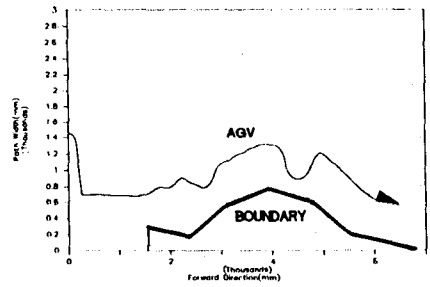


그림 3. 굴곡이 있는 경계를 따라가는 무인차경로

지금 위치에서 무인차가 계속 평행 맞추기를 시도해도 면의 상태가 나빠 평행이 맞지않게 된다면 이 상태를 벗어나도록 조금 전진하여 평행을 맞추 필요가 있기 때문이다. 보통의 경우 평행제어는 과도상태(overshoot)없이 몇단계를 거쳐 지정된 정밀도로(약 3mm 이내) 평행을 맞춘다.

그림 3은 무인차가 굴곡이 있는 경계를 약 70cm의 간격을 두고 따라가는 것을 보인다. 초기에 무인차는 벽면으로부터 약 146cm 떨어진 지점에서 시작하여 평행제어를 수행한 후 약 76cm를 명령된 회전에 의해 이동해서 벽면과 평행을 맞춘 후 직선운동을 한다. 이 과정에서 무인차는 이동하는 거리가 크므로 앞에서 설명했듯이 처음에는 회전운동을 하다가 초기와의 이동각이 커져 그 다음부터는 거의 직선운동을 하는 것을 볼 수 있다. 이때는 직선운동이 다시 회전운동으로 바뀌지않고 이동거리를 다온 후에 각도만을 변경 했음을 알 수 있다. 이것은 전진방향으로의 최소이동을 하기 위함이다. 다시 평행을 맞춘후 직선운동을 한 것으로 나타나는데 실제로는 직진과 평행 그리고 약간씩 생기는 오차에 의한 위치조정 등이 계속 반복되었다. 벽면이 변하는 지점부터 무인차는 경계와 70cm의 간격을 유지하는 경계따라가기를 반복적으로 수행하면서 이동을 한다. 이때 평행과 위치조정이, 정밀한 평행조건과 약간의 직진과정속에 평행을 맞추어야 하는 이유, 그리고 평행이 다 끝난 후 위치조정이 이루어지는 순서 등의 원인으로 그림에서도 알 수 있듯이 약간씩 어긋나고 있음을 볼 수 있다. 간혹 경계와 센서사이의 각이 한계범위를 넘어서 위치조정이 되거나 할 경우는 왜란이 있는 운동이 이루어짐을 볼 수 있다. 특히 면이 튀어나온 부분에서는 초음파센서들의 성능이 불안정한 경우가 나타나기도 한다. 그러나 상대적으로 면이 들어가는 부분에서는 더 나은 성능을 나타낸다. 무인차의 궤적자료는 처음 평행을 맞춘 후를 기점으로 하여 내부주행기를 통해 얻었으

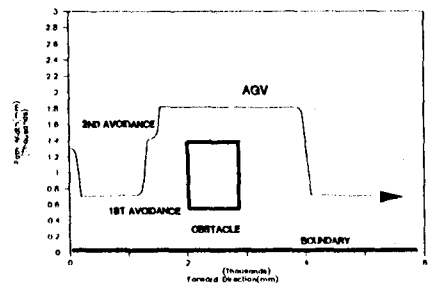


그림 4. 주행상에 나타난 장애물을 회피하는 무인차경로

며 통과하는 길의 폭은 3 m이고 길이는 약 7m이었다.

그림 4는 직선 경로에서 무인차가 불러 장애물을 회피하는 것을 보여준다. 벽면과 70cm를 맞추어 진행하다가 무인차의 앞의 두 센서가 장애물을 감지하는데 두 센서가 모두 감지할 경우는 70cm의 폭을 이동하고 다시 하나의 센서만이 감지를 하여 다시 40cm의 폭만큼 이동이 이루어졌음을 알 수 있다. 이 경우는 회피동시 전진속도를 줄여 최대한 수평이동을 하였는데 그것은 반복되는 회피운동시 장애물과의 거리를 최대한 멀리 떨어지게 하기 위함이다. 장애물감지는 무인차와 장애물간의 간격이 90cm 가 될 때 발생하도록 하였다. 무인차는 장애물의 크기를 단지 앞의 두 센서만을 사용하여 알 수 밖에 없으므로 그것에 따라 장애물 감지시 회피폭을 결정한다.

그림 5는 굴곡이 있는 주행로상에 장애물이 있는 경우의 무인차의 운동을 보여준다. 무인차는 경계를 일정한 간격을 두고 따라가다가 장애물과의 거리가 120 cm가 될 때 회피를 시작한다. 회피 후 그 방향에서 장애물을 만날 때까지 직진운동을 한 후 장애물을 만나면 다시 경계 따라가기를 하여 장애물을 통과한다. 장애물을 통과한 후 처음 회피가 시작된 지점까지를 목표로 하여 앞 센서를 통한 감지를 하면서 경로복귀운동을 한다. 그 후의 운동은 다시 경계 따라가기를 수행하면서 진행한다. 여기에서 어려운 문제는 감지한 장애물의 크기를 모르기 때문에 어떤 방식으로 장애물을 통과하는가이다. 우선 그 장애물의 대략적인 폭은 알 수 있으므로 회피의 폭은 결정될 수 있고 문제는 장애물에 접근하는 것이다. 따라서 전진방향의 자유공간이 확보되었으므로 장애물을 만날 때까지 측면센서를 가동시키면서 직진한다. 측면센서가 장애물을 감지하면 그때부터 장애물을 통과하기 위해 장애물의 경계를 따라간다. 이 과정은 보통의 경계따라가기와 같은 동작이다. 측면센서가 장애물의 통과완료를 감지하면 그 순간부터 일정한 거리를 더 진행한 후 경로복귀운동을 시작한다. 그 이유는 무인차의 급작스런 운동시 장애물의 모서리 등과 같은 부분과의 충돌을 피하기 위함인데 그런 상황은 센서의 사각지대가 되는 경우가 대부분이다. 복귀운동은 무인차의 상태가 불안정하므로 처음 장애물회피가 시작된 위치의 X좌표를 목표로 이동하면서 앞 센서로 감지를 계속한다. 목표지점이나 앞 센서에 의한 운동변경이 감지되면 다시 경계따라가기가 시작된다.

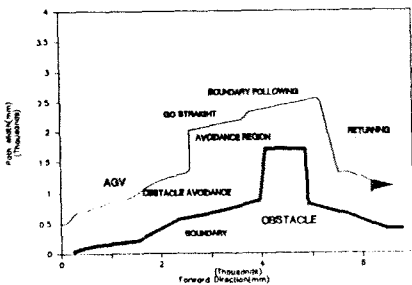


그림 5. 장애물을 회피하며 경계를 따라가는 무인차 경로

5. 결론

본 논문에서는 제한된 환경하에서 경계를 따라가고 장애물을 회피하는 무인차의 운동제어에 단순한 신경회로망이 사용될 수 있음을 실제로 해 보았다. 환경의 어떤 표현과 주행지도 없이도 거리센서정보와 내부주행기만을 사용하여 반사적인 자율항법제어가 이루어졌다. 평행과 이동을 담당하는 뉴론들은 각기 선택된 상태에서 적합한 제어행위를 취하게 되며 그러한 동작이 조합되어 굴곡이 있는 경계따라가기가 이루어진다. 그러나 더욱 복잡한 상황에서도 지능적인 무인차제어가 이루어지기 위해서는 상황의 판단이나 경험에 의한 추론 등의 기능이 요구되는데 이러한 작용은 퍼지논리를 이용하는 것이 효과적일 것이다. 따라서 신경회로망과 퍼지논리제어기가 서로의 단점을 보충하는 방향으로 결합한 뉴로-퍼지제어기와 같은 구성이 무인차의 지능제어에 더욱 적합할 것이다.

참고 문헌

- [1] Ulrich Rembold, Paul Levi, "Sensors and control for autonomous robots", *Proc. of Intelligent Autonomous Systems*, pp. 79-95, 1986.
- [2] G. Honderd, W. Jongkind, and C.H.van Aslst, "Sensor and navigation system for a mobile robot," *Proc. of Intelligent Autonomous Systems*, pp.258-264, 1986.
- [3] Ronald C. Arkin, "Motor Schema - Based Mobile Robot Navigation", *Int'l j. of robotics and research*, vol. 8, no. 4, pp. 92-112, 1989.
- [4] M. H. Soldo, "Reactive and preplanned control in a mobile robot," *Proc. 1990 IEEE Conf. of Robotics and Automation*, pp. 402-406, 1990.
- [5] M. J. Mataric, "Environment learning using a distributed representation," *Proc. 1990 IEEE Conf. of Robotics and Automation*, pp. 1128-1132, 1990.
- [6] Rodney Brooks, "A Layered Intelligent Control System for a Mobile Robot", *IEEE Trans. on Robotics and Automation*, vol. 2, no. 1, pp. 14-23, April 1986.
- [7] Andrew Barto, "From chemotaxis to cooperativity : abstract exercises in neuronal learning strategies," *Proc. of the Neuron as a Computational Unit*, pp. 73-98, 1988.
- [8] Kumpati S. Narendra and M.A.L.Thathachar, "Learning automata - a survey," *IEEE Trans. System, Man, and Cybernetics*, vol. 4, No. 4, July, 1974.
- [9] Y. S. Kim and J. G.Lee, "Design of a pole-balancing controller using neural networks," *Proc. of int'l joint conf. on Neural Networks*, vol.2, pp. 619-622, 1990.