

지능형 하이브리드 자기 동조 기법을 이용한 강건 제어기 설계

The Robust Control System Design using Intelligent Hybrid Self-Tuning Method

*권혁창, **하상형, *서재용, *조현찬, ***전홍태

* 한국기술교육대학교 정보기술공학부
** 기술신용보증기금 서울기술평가센터
*** 중앙대학교 전자전기공학부

*Hyeuk-Chang Kwon, **Sang-Hyung Ha, *Jae-Yong Seo, *Hyun-Chan Cho, ***Hong-Tae Jeon

* School of Information Technology Eng, Korea University of Technology and Education
** Seoul Technology Appraisal Center of Korea Technology Credit Guarantee Fund
*** Division of Electronic & Electrical Eng, Chung Ang University
E-mail : kwonhc@kut.ac.kr

Abstract

This paper discuss the method of the system's efficient control using a Intelligent hybrid algorithm in non-linear dynamics systems. Existing neural network and genetic algorithm for the control of non-linear systems work well in static states, but it be not particularly good in changeable states and must re-learn for the control of the system in the changed state. This time spend a lot of time. For the solution of this problem we suggest the intelligent hybrid self-tuning controller. it includes neural network, genetic algorithm and immune system, it is based on neural network, and immune system and genetic algorithm are added against a changed factor. We will call a change factor an antigen. When an antigen broke out, immune system come into action and genetic algorithm search an antibody. So the system is controled more stably and rapidly. Moreover, The Genetic algorithm use the memory address of the immune bank as a genetic factor. So it brings an advantage which the realization of a hardware easy.

1. 서론

현재의 역학 시스템들은 나날이 복잡해지고 있으며 정밀성을 요하고 있다. 이러한 역학 시스템들은 인간이 할 수 없는 어려운 환경에서 임무를 수행하고 정밀성과 지속성을 요구하는 부분에서 수요가 날로 늘어가고 있다.

역학 시스템을 이용한 작업을 원활히 수행하기 위해서는 시스템에 대한 모델링이 이루어져야 하며 제어의 정확성을 위해 수학적 방식이 요구된다. 하지만 역학 시스템들이 갖는 비선형성을 성

확히 측정할 수 없거나 수식화가 불가능하기도 하다. 이와 같이 실제 물리 시스템을 수학적 모델로 표현할 때 정확히 측정할 수 없거나 수식화할 수 없는 물리적 특성들을 고려하기 위하여 불확실성(uncertainty)이란 용어를 사용한다. 이와 같은 불확실성이 존재하는 시스템에 대한 기존의 수학적 제어 이론으로는 강건한 제어기(robust controller)의 구성이 어렵다.

따라서 최근에는 이것을 해결하기 위해 기존의 수리적인 접근 방식보다는 인간의 사고능력과 적

능력을 모방한 지능형(intelligent controller)이 제안되고 이를 응용한 제어기의 구성에 대한 연구가 활발히 진행되고 있다. 여기에는 신경망 제어와 유전 알고리즘 제어 기법 등이 있다. 이와 같은 지능형 기법은 고도의 비선형 시변 시스템(nonlinear time-varying system)에 대한 제어기 설계에 적용시, 강건하고 적응성이 있는 장점을 발휘할 수 있다. 그러나 인간의 뇌를 모방한 신경망, 자연계의 진화의 속성을 이용한 유전 알고리즘 등과 같은 지능형 제어 알고리즘은 동적인 시스템과 제어 환경과의 상호 작용을 통해서 얻어진 정보를 바탕으로 제어 변수를 조정하는 공통점을 가지고 있지만, 새롭게 변하는 외부 환경이나 시스템의 변화에 대한 재학습 및 재진화가 필요하다.

본 논문에서 제안할 제어 시스템은 기존의 시스템이 갖지 못했던 안정성과 빠른 적응성을 가지며 하드웨어로 구현하기가 용이해 실제 제어 시스템에서의 사용이 평이하도록 고안했다.

제안된 시스템의 우수성은 고도의 비선형성을 갖는 로봇 매니플레이터의 제어 시뮬레이션을 통해 보일 것이다.

2. 지능형 하이브리드 자기 동조 제어 시스템

2.1 지능형 하이브리드 자기 동조 제어기

지능형 하이브리드 자기 동조 기법에서 시스템의 중추적 제어는 BackPropagat ion방식의 신경망 제어를 사용하여 비선형 제어가 가능하게 했으며 입출력 정보를 이용한 weight 조절을 통한 자기 동조를 수행한다. 여기에 면역 시스템을 결합하여 불확실 변이 인자에 대응하도록 설계되었다. 즉, 불확실 변이 인자를 항원으로 인식하여 항원에 대응하기 위한 항체(신경망의 weight 집단)를 면역 시스템을 통하여 검색이 가능하도록 하였다. 또한 면역 알고리즘에 유전 알고리즘을 결합하여 면역 시스템의 항체 검색시 전역적인 해를 찾아낼 수 있도록 했으며 유전 알고리즘의 유전 인자를 메모리 주소를 사용함으로써 하드웨어적 구성이 쉽도록 구현되었다. 이와 같이 면역 시스템과 유전 알고리즘을 사용하여 신경망이 자기 동조를 보다 안정적으로 빠르게 이루어내도록 하였다.

2.2 시스템의 구조

제안한 제어 시스템의 블럭도는 [그림 1]에 보여지는 것과 같다.

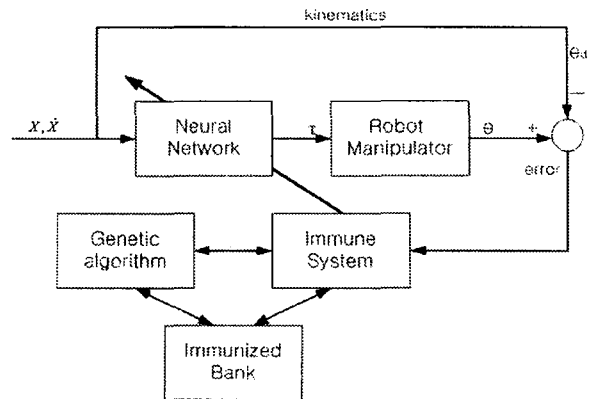


그림 1 하이브리드 자기 동조 기법을 사용한 시스템 블럭도

위의 블럭도를 순서대로 설명하면 우선 로봇 매니플레이터의 끝단을 원하는 속도를 가지고 원하는 위치로 이동시키기 위해 신경망의 입력으로 속도, 위치 값을 입력한다. 신경망은 이미 학습된 weight값으로 필요한 토크를 출력하여 로봇 매니플레이터에 입력한다. 로봇 매니플레이터는 입력된 토크에 따라 움직이며 각 관절각을 얻을 수 있다. 이 값을 위치를 관절각으로 바꾸어주는 kinematics식에 의해 도출된 희망 각과 비교한다. 만약 에러가 발생되었다면 에러 패턴을 조사하여 면역 체계 시스템에 사용함으로써 최적 항체를 검색하여 신경망에 새롭게 검색된 항체를 적용하여 에러를 제거하게 된다. 만약 최적 항체가 검색되지 않는다면 가장 근접한 항체를 선택하여 신경망에 적용시켜 재학습 시킴으로써 빠른 학습이 가능하도록 하였다.

2.3 immune back

2.3.1 항체 구성

immune bank에 저장되는 항체의 구성 요소의 첫번째는 특정 상황에 대한 학습을 마쳤을 때의 신경망의 weight 집단이고, 두번째는 에러패턴이다. 이 두요소를 짝을 이루어 메모리에 저장시킨다.

항체의 첫번째 구성요소인 신경망의 weight 집단은 신경망의 Backpropagation 방법을 이용한 학습을 통하여 얻어낸다. 그리고 두번째 구성요소인 에러패턴은 시스템의 초기 설정 항체를 기준으로 삼아 제어를 수행했을 때 발생된 error패턴을 저장한 것으로, 에러가 발생되었으므로 재학습을 하여 새로운 weight값을 도출하고 이 값과 error패턴을 짝지어 저장하게 된다.

2.3.2 메모리에 항체 저장

항원에 대한 항체를 메모리에 저장하는데 있어

중요한 것은 근접한 항체끼리 유사성을 가지도록 저장하는 것이다. 이것은 항체를 생성하는데 무작위로 생성하는 것이 아니고 순차적으로 생성함으로써 이루어진다. 예를 들면 무게가 항원으로 작용할 때 이 항원을 1kg에서 20kg까지 1kg의 간격을 가지고 학습을 시키는데 순서대로 1kg, 2kg, ..., 20kg의 순으로 항체를 생성함으로써 근접한 항체끼리는 비슷한 성질을 갖도록 한다.

2.4 제어 알고리즘

2.4.1 항체 검색

로봇 매니퓰레이터에 에러가 발생되면 우선 에러 패턴을 만들고 이 에러 패턴을 immune bank에 이미 저장되어 있는 에러 패턴과 병렬적으로 비교한다. 그래서 발생한 에러 패턴과 맞아 떨어지는 에러 패턴이 있다면 이것과 짝을 이루는 신경망 weight 집단을 신경망에 대입하여 제어를 수행함으로써 에러가 제거된다.

그러나 발생한 에러 패턴과 맞아 떨어지는 것이 없을 경우 임의의 항체를 선택하여 유전 알고리즘을 이용하여 항체 주소를 유전 인자로 한 유전 연산자를 수행함으로써 적합한 항체를 검색하게 된다. 여기서 적합도는 발생한 에러 패턴과 기존의 에러 패턴간의 xor 연산을 수행한 결과를 가지고 판별하게 된다. 차이가 없을 수록 적합도가 높은 것이다. 이처럼 유전 연산자로 전역적 해의 추적이 가능한 것은 근접한 항체끼리 유사한 특성을 갖고 있기 때문이다.

2.4.2 새로운 항원에 대한 항체 생성

유전 연산자에 의해 선택된 전역적 해는 항원에 부합하는 완벽한 항체는 아니며 존재하는 항체들 중에서 가장 부합하는 항체이므로 이 항체의 신경망 weight 집단을 신경망 제어기에 적용시킨 후 완벽한 제어가 가능하도록 재학습이 필요하다. 하지만 가장 부합하는 항체를 가지고 신경망 학습을 시킴으로써 기존 학습보다 빠른 학습이 가능하다.

3. 평가

평가에 사용된 로봇 매니퓰레이터의 시뮬레이션은 그림2에 보여지는 것과 같이 Two-Link Robot Arm을 이용하였다. Robot Arm의 끝단인 원을 그리도록 제어를 수행한다.

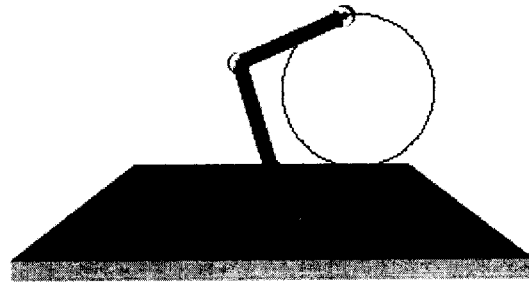


그림 2 시뮬레이션

우선 시뮬레이터가 10kg의 물체를 잡고 원을 그린다는 기본 가정을 하여 10kg에 대응하는 항체(신경망 Weight)를 신경망 제어기에 적용한다. 이제 여기에 불확실성 요소를 입력하기 위해서 12kg의 물체로 대체시킨다. 그 결과는 다음과 같다.

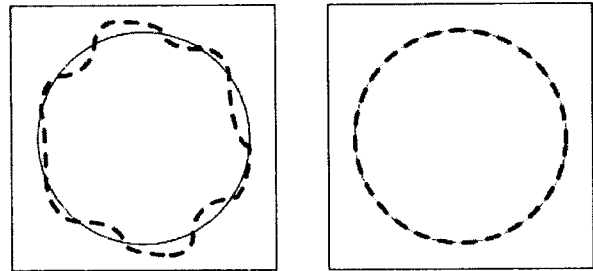


그림 3 불확실성 입력에 의한 에러 발생과 에러 패턴 분석 후의 제어 수행 결과

불확실성 입력에 따른 에러가 발생 되었으며 에러 패턴을 만들기 위해 1cycle동안 시스템을 작동시켜 에러 패턴을 만들어 낸다. 이 에러 패턴을 10kg기준 상태에서 이미 만들어진 에러 패턴과 비교함으로써 현재 12kg의 물체를 가지고 있다는 것을 알아낼 수 있다. 그래서 곧바로 올바른 작업을 수행한 것을 볼 수 있다.

또한 이미 항체를 가지고 있는 항원이 아닌 경우(11.5kg같은 경우) 대체로 기존 신경망 제어보다 빠르게 대처가 가능했지만 그렇지 못한 경우도 있어서 시뮬레이션 결과를 올릴 수가 없었다.

4. 결론 및 향후과제

시뮬레이션에서 보는 바와 같이 항원에 대한 항체가 immune bank에 존재하면 곧바로 올바른 제어가 가능하다. 하지만 항체가 없는 항원이 입력되었을 때는 아직 시간을 많이 소비한다. 이것은 시뮬레이션의 미완으로 인한 에러 패턴간의 연계성 부족으로 인해 유전 알고리즘에서 시간을

많이 소비할 뿐만 아니라, 유전 알고리즘에 의해 선택된 항체를 가지고 다시 신경망 학습을 수행해야 하기 때문에 신경망만을 가지고 재학습 시키는 것보다 느린 경우가 발생하는 것으로 추측된다. 그래서 이 부분을 좀 더 보완한다면 기존 제어기보다 빠르면서도 강건한 제어가 가능할 것이다.

5. 참고문헌

- [1] 최영길, “면역화된 귀환 신경망을 이용한 로봇 매니플레이터 제어 시스템 설계”, 1997년 12월
- [2] 원광재, “면역화된 귀환 신경망을 이용한 로봇 매니플레이터 제어 시스템 설계”, 1997년 12월
- [3] 서재용, “다층 신경망과 면역 알고리즘을 이용한 로봇 매니플레이터 제어 시스템 설계”, 1997년 12월
- [3] Craig, John J, “Introduction to Robotics : Mechanics & Control”, 1986년
- [4] Martin T.Hagan, Howard B.Demuth, Mark Beale, “Neural Network Design”, 2001
- [5] Valluru B.Rao, Hayagriva V.Rao, “C++ Neural Networks and Fuzzy Logic”