

FNN에 의한 선박의 제어

강창남
제주대학교

A ship control by fuzzy neutral network

Chang-Nam Kang
Cheju National University*

Abstract - Fuzzy neural ship controllers is used in ship steering control. It can make full use of the advantage of all kinds of intelligent algorithms. This provides an efficient way for this paper. An RBF neural network and GA optimization are employed in a fuzzy neural controller to deal with the nonlinearity, time varying and uncertain factors. Utilizing the designed network to substitute the conventional fuzzy inference, the rule base and membership functions can be auto-adjusted by GA optimization. The parameters of neural network can be decreased by using union-rule configuration in the hidden layer of the network. The ship control quality is effectively improved in case of appending additional sea state disturbance. The performance of controller is evaluated by the system simulation using simulink tools.

1. 서 론

선박제어는 자동조타시스템의 기본이다. 피지제어기는 제어대상이 정확한 모델링이 필요없고 전문가의 경험에 의한 룰베이스의 적용으로 시스템에 쉽게 적용 가능한 점이 있으나 전문가의 주관적인 견해를 판단하는 기준 척도가 애매하고 또한 선박의 운동은 예상치 못한 외란의 영향을 많이 받아 룰베이스를 항상 업데이트 해야된다. 이런 단점을 해소한 우수한 제어기를 구성하기 위하여 퍼지구조와 학습능력이 있는 신경망과 통합한 퍼지신경망선박제어기를 시도해 보고자 한다. 파라미터들은 유전자 알고리즘에 의하여 최적화한 퍼지 RBF 회로망을 이용하며 룰베이스 및 멤버십 함수는 자동조정된다.

2. 본 론

일반적으로 퍼지 선박 제어 시스템의 구조는 Fig.1처럼 합성 제어기를 가진 피드백 제어 시스템이다. 퍼지 제어기 모델은 3개의 기본적인 함수 성분을 가지는데 입력, 출력인터페이스 그리고 처리부이고 각각 퍼지화 비퍼지화 퍼지추론이다. 퍼지제어기는 주고 받는 데이터가 크리스프한 데이터 이므로 상호 바꿔줄수 있는 변환장치가 필요하게 되는데 퍼지화기, 비퍼지화기가 변환을 담당한다.

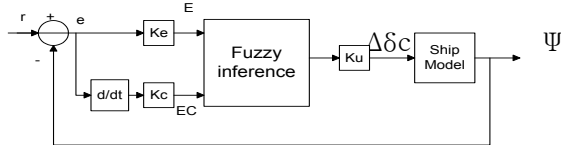


Fig.1 Basic scheme of fuzzy control system

제어기에서 선박의 코스에라 e 와 도함수 $\frac{de}{dt}$ 는 두 입력변수로 각각 계인 $\Delta\delta_c$ 는 출력변수이다. r 과 ψ 는 각각 기준 입력과 실제 출력이다. 퍼지 추론을 위하여 E 와 EC 는 e 와 $\frac{de}{dt}$ 가 논리영역에서 K 와 K_c 에 의하여 퍼지화된 입력 값이다. U 는 추론된 출력변수이다. 실제 각각 계인 $\Delta\delta_c$ 는 선박 모델 입력값으로 K_u 에 의하여 비퍼지화된다. 퍼지화 인터페이스에서 E 와 EC 는 연속적인 범위[-6,6]에서 균일분포이고, $\exp(-\frac{(x-m)^2}{\sigma^2})$ 로 묘사되는 가우시안 멤버십함수이다. m 은 퍼지집합을 나타내고 σ 는 그들의 전개이다. 비퍼지화 인터페이스에서 출력변수 U 는 싱글톤 이산 분포로 [-5 5]범위이다. 룰베이스는 항상 전문가의 실제적인 지식으로부터 요약되고, i 번째 룰은 다음처럼 묘사된다.

$R_i: IF E \text{ is } A_{1i} \text{ and } EC \text{ is } A_{2i} \text{ THEN } U \text{ is } B_i$ 여기서 R_i 는 i 번째 룰이다. A_{1i} and A_{2i} 는 멤버십함수 $\mu_{A_{1i}}(E)$ and $\mu_{A_{2i}}(EC)$ 에 의하여 정의되는 퍼지집합이다. B_i 는 싱글톤 출력 퍼지 집합이다. 거기

에는 기껏해야 전문가의 의견으로부터 룰베이스에 49개의 룰이 있다. 즉 두개의 입력변수 E 와 EC 각각 7개의 퍼지집합이다.

RBF 네트워크 은 3층의 피드-포워드 이다. 숨겨진 뉴런의 출력은 복사기저함수인 Gaussian 함수를 적용해서 유도한다. 이것은 입력벡터와 센터 사이의 차이이다. 선형 출력 뉴런은 숨겨진 뉴런 출력들의 합으로 계산한다.

$$g_i(\bar{x}) = g_i(x_1, x_2, x_3, \dots, x_n) = \exp\left[-\frac{\|\bar{x} - \bar{c}_i\|^2}{\sigma_i^2}\right], f(\bar{x}) = \frac{\sum g_i(\bar{x}) w_i}{\sum g_i(\bar{x})}$$

$x_1, x_2, x_3, \dots, x_n$ 과 $f(\bar{x})$ 는 네트워크의 입출력 변수이다. $w_1, w_2, w_3, \dots, w_n$ 은 숨겨진 층과 출력층 사이의 연결 하중치이다. $g_1, g_2, g_3, \dots, g_n$ 은 여러 가지 파라미터 \bar{c}_i 와 σ_i 가 있는 Gaussian 함수이다.

국부적인 근사능력은 국부 민감도에 의하여 결정되며 \bar{x} 와 \bar{c}_i 사이에 거리가 짧을수록 출력값은 더 커진다. 이것이 퍼지 룰 접근을 위하여 적절한 것이다. RBF 네트워크에서 숨겨진 가지들은 룰로서 맵핑되며 파라미터의 감소로 인하여 룰을 제거 하는 것과 같은 효과를 가져온다. 이 적용에서 RBF네트워크는 e 와 de/dt 를 가진 2차의 RBF 신경망 선박 모델을 구성하기 위하여 퍼지 추론을 대체하여 사용하였다.

$$g_i = \prod_{j=1}^n \mu_{A_j}(x_j) = \prod_{j=1}^n \exp\left[-\frac{(x_j - c_{ji})^2}{\sigma_j^2}\right] = \exp\left[-\frac{\|x - c\|^2}{\sigma^2}\right]$$

x_1 과 x_2 는 $\mu_{A_{1i}}(E)$ and $\mu_{A_{2i}}(EC)$ 이고 w_i 를 출력 $\mu_{B_i}(U)$ 이고 $f(\bar{x})$ 는 추론 U 의 결과이다. 변환요소들은 전체시스템에 중요하다

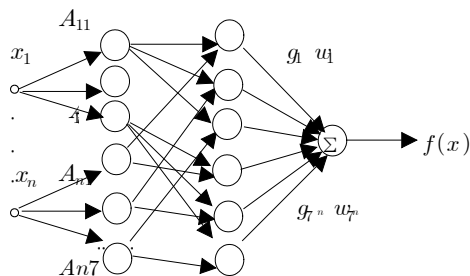


Fig.2 Structure of RBF Network based on union rule configuration

편차 e 와 도함수 de/dt 는 변환요소 K_c 와 K_e 에 의해 퍼지 제어기 영역으로 맵핑된다. U 는 RBF네트워크에 의하여 추론되며 K_u 요소에 의하여 각각 계인으로 변환된다. 시스템의 동적성능은 K_c 와 K_e 에 의하여 크게 영향을 받을 수 있다. 제어기 출력은 K_u 요소에 영향을 받게 된다. 동적응답은 K_u 가 작아지면서 증가하게 될 것이다. 이 값들을 조정하거나 자기적용 할 수 있는 요소에 대한 많은 제안이 있으나 너무 복잡하여 실현이 어렵고 다른 방법이 제안되는데, K_c 는 고정시키고 K_e 와 K_u 는 오차의 절대값으로 조정된다. 방위제어선박에 대하여 실제 입력 오차의 범위는 $[-30^\circ, 30^\circ]$ 이고 논리영역은 $[-6, 6]$ 이다. 오차의 변환요소는 $K_e = 6/30 = 0.2$ 이다. $\frac{de}{dt}$ 의 실제 범위는 $[-6/s, 6/s]$ 이다. K_c 는 다음으로 정리되고, 12보다는 커야 하며 오차의 절대치가 0.5보다 작을 때만 12이다.

$$K_c = \begin{cases} a - \frac{a-12}{0.5} |E|, & \text{if } |E| \leq 0.5 \\ 12, & \text{if } |E| > 0.5 \end{cases}$$

K_c 는 역시 자기 적응되며 타기의 허용 영역은 $[-35, 35]$ 이고 논리영역은 $[-5, 5]$ 이며 다음으로 정리된다. 최대값은 7보다 커야 하며 단지오차의 절대치가 0.5보다 약간 큰 경우만 7이다.

$$K_u = \begin{cases} b - \frac{7-b}{0.5} |E|, & \text{if } |E| \leq 0.5 \\ 7 & \text{if } |E| > 0.5 \end{cases}$$

여기서 a 는 12보다 큰 조정 가능한 파라미터이고, b 는 7보다 작은 조정 가능한 파라미터이다. $|E|$ 는 오차의 절대값이고, 전문가들의 대부분 주관적 의견들은 룰베이스 설계 및 멤버십 함수 선택에 달려있다. 최적화 상태를 만들 필요가 있는데 이는 다른 선박 또는 똑 같은 배의 다른 상태에서 최적의 상태를 찾는 데 있기 때문이다. 룰베이스와 RBF 네트워크 사이의 대응되는 맵핑은 방법과 전개, 출력연결을 포함하는 RBF 네트워크 파라미터를 조정하는 것과 같은 룰베이스의 최적화를 가져온다. 룰베이스를 조정하는 방법은 성능 판단을 기저로 한 자기 조정 퍼지 제어기의 방법보다도 훨씬 쉽다. GA는 국소적인 최적화를 피하기 위하여 파라미터를 최적화 하는데 채택된다. GA는 생명체의 자연적인 적응과정에서 기점을 둔 모델을 사용하는 무작위 최적화의 알고리즘이다. 생명체의 진화론으로 1859년에 다윈에 의하여 첫개념이 도입되었다. 진화론 뒤에 단지 4개의 기본적인 과정이 있다. 선택, 번식, 변종, 경쟁. 여러 저자들에 의해 유전 시스템을 모델링하는 과정들이 제안되었다. 자연적으로 가장 적합한 생존기구를 채택했으므로 자기 적응을 가졌고 최적화 문제에 널리 사용되어 왔다. 진화에 속도를 가하여 한 알의 씨앗이 인류 초기에 주어졌으며 무작위로 추출된 몇몇 개개인들은 전문가 경험으로부터 이들을 대체하였다. 일반적으로 최적화와 근사치 해결은 가장 적합한 파라미터가 선택될 때만 실제 근사치 해결이 가능하다. 파라미터 부호화는 GA 최적화의 첫 단계이다. 파라미터들은 부호 없는 2진수로 변환되며, 양적인 요소 파라미터와 네트워크 구조의 파라미터가 있다.

$k_1 = a$, $k_2 = b$ 로 가정하면 대응되는 이진수 코드는 각각 B_1 , B_2 이고 맵핑관계는 다음과 같다. $B_i = f \left[\frac{K_i - K_{i\min}}{K_{i\max} - K_{i\min}} (2^n - 1) \right]$ ($i = 1, 2, \dots$)

여기서 B_i 는 K_i 에 대응하는 n 비트 이진수, $K_i \in [K_{i\min}, K_{i\max}]$ f 는 10진수에서 2진수문자열로 변환함수. B_1 과 B_2 는 한 개체로 섞일 것이다. 최적 파라미터들은 앞에서 C_i , σ_i , w_i 를 포함하여 확인됐다. GA 최적화 필요성에 대하여 파라미터들은 다섯 그룹으로 된다. c_i 와 d_i 는 7에 대응되는데 $c_1 \dots c_7$ 를 의미하고 입력치 $x_1(E)$ 의 $\sigma_1 \dots \sigma_7$ 를 전개한다. e_i 와 f_i 는 다른 입력치 $x_2(EC)$ 의 똑같은 변수에 대응된다. 마지막 그룹 g 는 49개의 연결 하중치이다. 동일 σ 는 숨겨진 마디에 대응되는 것으로 간주한다. d 그룹과 f 그룹은 똑 같다. 거기에는 GA 최적화를 위하여 남겨진 4그룹만이 있다. 모든 파라미터는 십진수에서 16비트 이진 문자열로 변환될 것이다. 일반적으로 오버슈트, 과도시간, 정적편이 오차는 특정 인덱스로 간주되며, 바이어스 오차와 시간의 곱의 정수를 대상함수로 고려된다. 명백히 J 가 작아지면 각각 적응의 정도는 더 좋아진다. $J = \int_0^\infty t |e(t)| dt$

모의 실험대상은 대형 선박 운동 모델이다 이 선체의 운동에 관련된 모델들은 Abkowitz 모델, MMG 모델이 있는데 여기서는 Abkowitz 모델을 사용하였으며 선박, 보트 타등이 이음매 없는 하나의 유닛으로 고려되는 것이 특징이다. 수학적 식으로 다음과 같이 정리된다. 입력값 u_i 는 라다로 부터이며 출력치 Ψ 는 코스각이다

$$(m - X_u) \Delta \dot{u} = f_1(u, v, r, \delta)$$

$$(m - Y_v) \dot{v} + (m x_G - Y_r) \dot{r} = f_2(u, v, r, \delta)$$

$$(m x_G - N_v) \dot{v} + (I_Z - N_r) \dot{r} = f_3(u, v, r, \delta)$$

여기서 m 은 선박의 중량이고 x_G 는 좌표축의 원점에서 선체중심까지 거리, I_Z 는 ox 축의 배의 전후 방향으로 관성모멘트, u 는 전진 속도, v 는 롤링 속도, r 은 침로 이탈의 각속도, δ 는 라다각이며 x_u, y_v, z_r, N_v and N_r 는 유체 운동의 도함수, f_1, f_2, f_3 는 어떤 순서를 가진 u, v, r, δ 와 그들 도함수에 관계함수이다.

Matlab7에 있는 Simulink는 위에 언급한 방법을 개선하기 위하여 채택되었다. 이 퍼지 시스템 블록은 파라미터는 세대 크기를 20으로 하고, 변이율은 0.001, 교차율은 0.1, $K_c = 0.2$, $K_c = 12$, $K_u = 7$ 로 한 GA 최적화에 의하여 RBF 네트워크에 중점을 두었다. 통상적인 방법으로 모의 실험한 결과가 그림 3에 나타났다. 새로운 방법으로 모의 실험한 결과가 그림 4에 10 세대로 최적화된 후 나타내었다. 상승시간이 최적화 이후 짧아졌고, 응답속도는 단축되었다. 이 성능지표는 선박을 제어하는데 아주

중요하다. 퍼지 물들은 초기 룰베이스 설계한 것을 피하면서 GA 최적화를 통하여 직접 추출하였다. GA 최적화는 초기설계 때나 변화하는 대상의 특성의 것들을 수반하는 부당한 파라미터들을 조정할 수 있다. 다소 퍼지 제어를 설계하는 것이 더 쉬워진다.

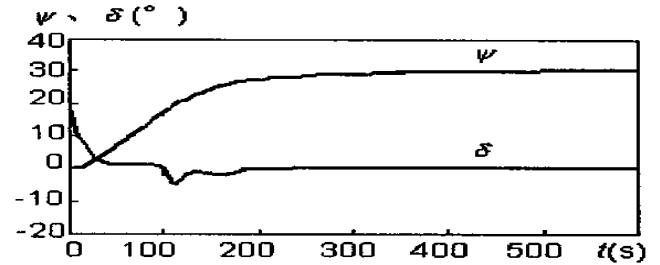


Fig.3 simulation result with conventional method

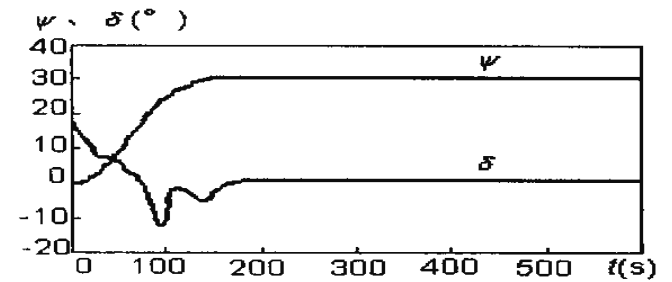


Fig.4 simulation result with new method

3. 결 론

퍼지 신경망제어기를 설계하여 선박의 제어 성능을 Simulink toolbox을 이용하여 모의 실험을 하였다. 퍼지신경망 선박제어기는 비선형, 시변, 불확실한 상황을 다루는데 RBF신경망을 사용하여 구축되며, 파라미터들은 GA에 의하여 최적화 됐다. 변환요소들은 여러 가지 단계의 과도과정을 통하여 적절히 자동조정됐다. 조정요소들은 GA Offline으로 최적화 됐다. 상승시간이 최적화 이후 짧아졌고 응답속도는 단축되었다. 다만 실선에서의 제어기를 장착후 실험 또는 결과 분석은 차후로 넘긴다.

[참고 문헌]

- [1] 이철영, "Optimal Design of Automatic Steering system of ship's at sea", J.of Institute of Navigation, Vol.2, pp.35-47, (1978)
- [2] A.Krogh, J.A.Hertz, "A simple weight decay can improve generalization", in Proc. Advances in Neural Processing Systems (NIPS), Vail, CO, pp.950-957, 1992.
- [3] Combs W E, Andrew J E., "Combinatorial rule explosion eliminated by a fuzzy configuration" IEEE Transaction on fuzzy systems, pp 1-11, 1998.
- [4] M Horigome Micro processor based AR Autopilot System for Ship's steering Ship Operation Automaton Isso479, (1979)
- [5] N.minorsky: Directional stability of automatically steered bodies, J. of ASNE, vol.34, 1922.