

오차 자기순환 신경회로망 기반 반능동 현가시스템 제어기 개발

The Development of Semi-active Suspension Controller based on Error Self Recurrent Neural Networks

이 창 구, 송 광 현

(Chang Goo Lee and Kwang Hyun Song)

Abstract : In this paper, a new neural networks and neural network based sliding mode controller are proposed. The new neural networks are an error self-recurrent neural networks which use a recursive least squares method for the fast on-line learning. The error self-recurrent neural networks converge considerably fast than the back-propagation algorithm and have advantage of being less affected by the poor initial weights and learning rate. The controller for suspension system is designed according to sliding mode technique based on new proposed neural networks. In order to adapt sliding mode control method, each frame distance between ground and vehicle body is estimated and controller is designed according to estimated neural model. The neural networks based sliding mode controller approves good performance through computer simulations.

Keywords : ESR(Error Self Recurrent) neural networks, degree of freedom, semi-active suspension

I. 서론

최근 들어 신경회로망을 이용하여 정적, 동적인 비선형시스템의 시스템 인식과 제어를 위한 여러 방법들이 활발히 연구되고 있다. 신경회로망은 병렬처리가 가능하고 적응능력이 있으며 특히 학습능력이 뛰어나기 때문에 비선형 시스템을 표현할 수 있고[1]-[2] 제어기를 신경회로망으로 구성할 수도 있다. 본 논문에서는 신경회로망을 이용하여 비선형 동적 시스템을 실시간 인식하고 인식된 모델을 바탕으로 제어기를 설계하는데 중점을 두었다. 이를 위해서는 신경회로망이 초기 연결강도(weight), 학습계수(learning rate)등에 민감하지 않아야 하고 무엇보다도 구조가 간단하고 빠르게 수렴하여야 한다. 그러나 뉴로 제어기에 사용되는 신경회로망은 보통 역전파 학습 알고리즘[3]을 사용하고 있어 지역최소(local minima)에 빠지기 쉽고 초기 연결강도값이나 학습계수에 민감할 뿐만 아니라, 특히 수렴속도가 매우 느리기 때문에 오프라인으로 시스템을 인식하거나 제어기를 학습한 후에 적용하고 있다. 따라서 시스템이 바뀌거나 학습되지 아니한 입력에 대해서는 응답특성이 좋지 못하며 이러한 제어기는 적응제어기라 할 수 없다.

이와 같은 단점을 개선하기 위하여 Scalero와 Tepedelnioglu, Lou와 Perez는 각각 신경회로망을 선형과 비선형 구간으로 구분하고 칼만 필터기법을 도입하여 수렴속도를 빠르게 하고 초기 연결강도값에 크게 영향을 받지 않도록 개선하였으나 여전히 학습계수에 민감한 단점을 가지고 있다[4]-[5]. 본 논문에서는 학습속도

가 빠르고 여러 파라미터에 덜 민감한 신경회로망으로 알려진 오차 자기순환 신경회로망(Error Self Recurrent Neural networks · ESRN)[6]을 도입하여 온라인으로 7 자유도 차량의 현가시스템을 인식하고 인식된 모델을 기반으로 예측제어 기법[7]과 슬라이딩 모드 제어기법[8]에 의하여 반능동 현가시스템 제어기를 설계하였다. 제안한 방법은 기존 뉴로 제어기와는 달리 온라인으로 시스템 인식과 제어가 이루어지므로 시스템 변화에 따른 재학습이 필요치 않으며 변화에 적응하는 적응제어 기능을 갖는다. 여러 도로 조건과 차량의 파라미터 변화에 따른 실험을 통하여 제안한 제어기가 반능동 현가시스템 제어에 효과적임을 보였다.

II. ESR 신경회로망과 학습알고리즘

신경회로망을 이용하여 비선형 동적 시스템을 실시간으로 인식하고 인식된 모델을 기초로 하여 적응제어기를 설계하려면, 먼저 신경회로망이 초기 연결강도, 학습계수 등에 민감하지 않아야 하고 무엇보다도 구조가 간단하고 빠르게 수렴하여야 한다. 본 장에서는 이러한 특성을 갖는 것으로 알려진 오차 자기순환 신경회로망에 대하여 간략히 기술하고자 한다.

1. ESR 신경회로망의 구조

일반적인 신경모델과 오차 자기 순환(ESR) 신경모델의 구조는 각각 그림 1, 그림 2와 같다[6]. 기존 신경모델과 ESR 신경모델의 차이는 그림에서와 같이 바이어스 입력에 있다. 바이어스는 각 활성화함수의 합출력을 증가하거나 감소시켜 주는 기능을 가지고 있다. 그림 1에서 알 수 있는 바와 같이 기존의 신경모델은 바이어스 입력값으로 +1 또는 -1을 제공한다. 그러나 시스템에 따라 이 값을 어떻게 선정하는지에 대한 명확한 이론적인 근거가 없다.

접수일자 1998. 2. 9., 수정완료 1999. 7. 26

이창구, 송광현 · 전북대학교 전자정보공학부

※본 연구는 과기부·전라북도 지정 지역협력센터인 메카트로닉스연구센터의 지원하에 이루어 졌습니다.

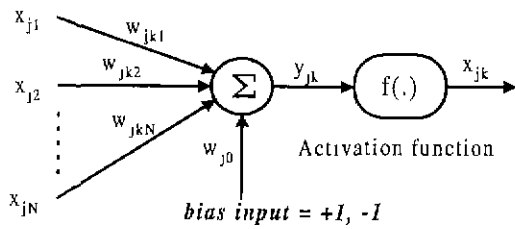


그림 1 기존 신경 모델.

Fig. 1. Conventional neuron model.

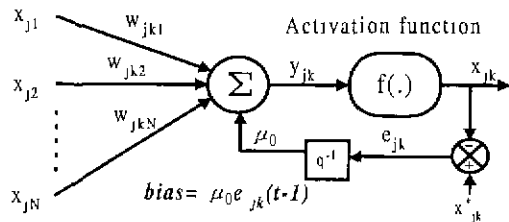


그림 2 오차 자기 순환 신경 모델.

Fig. 2. ESR neuron model.

ESR 신경 모델은 그림 2와 같이 바이어스 입력 값으로 한 단계 이전의 추정오차($e_{jk}(t-1)$)를 자기 순환시키고 이에 대한 연결강도(w_{j0})는 양의 상수값(μ_0)으로 한정한다. 이러한 구조는 net 입력을 증감시키는데 있어 과거 오차를 의미 있는 정보로 활용함으로써 자동적으로 바이어스의 크기와 기호를 설정할 뿐만 아니라 신경회로의 출력이 기준 목표치에 빠르게 수렴하는 기능을 갖도록 한다. 즉 바이어스는 한 단계 이전의 추정 오차가 양수 값이면 합출력(y_{jk})을 증가시키고 음수 값이면 합출력(y_{jk})을 감소시킴으로써 수렴속도를 빠르게 하는 기능을 한다.

2. 학습 알고리즘

ESRN의 학습 알고리즘은 신경모델을 그림 3과 같이 선형과 비선형 구간으로 나누고 선형구간에서의 연결강도는 RLS(Recursive Least Square)방법을 이용하여 갱신하고, 은닉층의 목표값(x_{jk}^*)은 역전파 학습 알고리즘 대신에 최적 방법을 이용하여 계산함으로써 학습계수에 대한 민감도를 가감할 뿐만 아니라 수렴속도를 빠르게 하였다[6].

설명을 간략화하기 위하여 하나의 은닉층을 갖는 다층 신경회로망을 생각하여 보자. 입력, 은닉, 출력층의 노드수가 각각 N, P, M이라 하고 성능지수를 (2.1)과 같이 표시하도록 하자. 여기에서 $o_{jk}, d_{Lk} = f^{-1}(o_{jk})$ 는 각각 출력층 k 노드의 목표값과 선형목표값이고, $w_{Lk} = [w_{Lk0}, w_{Lk1}, \dots, w_{LkP}]^T$ 는 은닉층과 출력층의 k번째 노드 사이의 연결강도 벡터이다. 또한, $x_i = [x_{i1}, x_{i2}, \dots, x_{iP}]$ 는 은닉층의 출력벡터이다.

$$E = \frac{1}{2} \sum_{k=0}^M (d_{Lk} - x_i^T w_{Lk})^2 \quad (1)$$

여기에서 우리는 출력층의 연결강도 matrix가 최적의 값

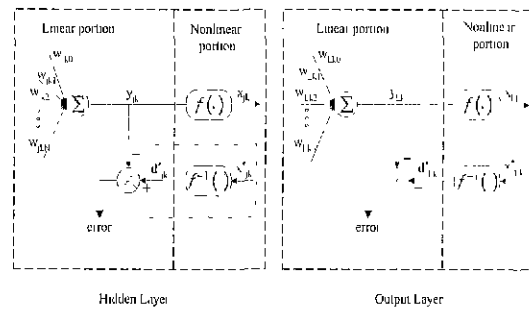


그림 3. 은닉층과 출력층에서 신경소자의 구분.

Fig. 3 Linear and nonlinear portion of the neurons in the hidden and output layers.

으로 학습되어 졌다고 보고 은닉층의 출력에 대한 (1)을 최소화하는 최적의 출력값을 구하여 이를 은닉층의 목표값으로 정한다. x_j 에 대한 함수 E의 최소화 결과는 (2)과 같이 표시할 수 있으며 이를 matrix형태로 다시 쓰면 (3)과 같다.

$$\frac{\partial E}{\partial x_i} = 0 \Rightarrow \sum_{k=0}^M d_{Lk} w_{Lk} = \sum_{k=0}^M w_{Lk} w_{Lk}^T \quad (2)$$

$$W_L d_L = W_L W_L^T x_i \quad (3)$$

(3)에서 $W_L = [w_{L0}, \dots, w_{LM}]$ 는 $P \times M$ 차원의 출력층 연결강도 matrix이고 $d_L = [d_{L1}, \dots, d_{LM}]^T$ 는 출력층의 선형 목표값으로 구성된 $M \times 1$ 벡터이다 즉, (3)은 M개의 방정식으로 P개의 알려지지 않은 값을 구하는 문제이며, 은닉층에서의 최적 목표값($x_i^* = [x_{i1}^*, \dots, x_{iP}^*]$)을 유도하는 데는 다음과 같은 두 가지 방법이 있다.

1) $P \geq M$ 인 경우, (3)를 만족하는 해는 하나 이상일 수 있으며 minimum norm method를 이용하여 (4)와 같이 구할 수 있다.

$$x_i^* = W_L (W_L^T W_L)^{-1} d_L \quad (4)$$

2) $P < M$ 인 경우, (3)를 만족하는 정확한 해는 존재하지 않으며 least-square approximate method를 이용하여 (5)와 같이 구할 수 있다.

$$x_i^* = (W_L W_L^T)^{-1} W_L d_L \quad (5)$$

다만, (4)와 (5)에서 $W_L W_L^T$ 는 nonsingular 매트릭스가 되어야 하며 만약 singular인 경우에는 pseudoinverse 개념을 적용하여 계산한다. (4), (5)로 구한 은닉층의 목표값으로부터 은닉층 k 노드의 선형 목표값은 (6)와 같이 구한다.

$$d_{jk}^* = f(x_{jk}^*)^{-1} \quad (6)$$

따라서, 새로운 학습알고리즘은 은닉층의 선형 목표값을 $d_{jk} = y_{jk} + \mu \delta_{jk}$ 대신에 (6)에 의하여 출력층부터 역방향으로 구한다. Bias 입력에 대한 연결강도($w_{j0} = \mu_0$)가 고정되었으므로, 이를 제외한 입력벡터에 대하여 RLS방법에 의하여 이득과 covariance 매트릭스를 (7)과 (8)로 계산한다.

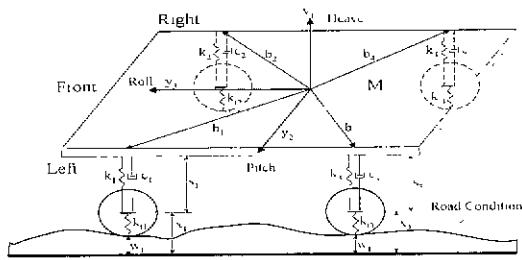


그림 4. 7 자유도 차량 모델.
Fig. 4. 7-degree of freedom vehicle model.

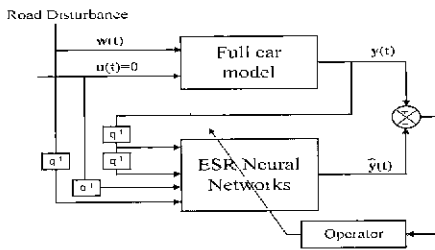


그림 5. Full car 모델 인식을 위한 방법.
Fig. 5. The method for estimating a full car.

$$k_j = P_j x_{j-1} / (\lambda + x_{j-1}^T P_j x_{j-1}) \quad (7)$$

$$P_j = (I - k_j x_{j-1}^T) P_j / \lambda \quad (8)$$

따라서 bias 입력에 해당하는 연결강도를 제외한 연결강도들은 (9)와 (10)에 의하여 갱신한다.

$$w_{Lk} = w_{Lk} + k_L (d_{Lk}^* - y_{Lk} + \mu_0 e_{Lk}(t-1)) \quad (9)$$

$$w_{jk} = w_{jk} + k_j (d_{jk}^* - y_{jk} + \mu_0 e_{jk}(t-1)) \quad (10)$$

III. 7 자유도 차량 모델의 인식

현가시스템 제어기를 설계하기 위한 신경회로망 모델은 실제차량 대신에 그림 4와 같은 구조의 7 자유도 차량 모델로부터 실시간으로 구한다. 그림 4는 도로장애에 의한 차륜 4개의 움직임과 상하(heave), 경사(pitch) 그리고 회전(roll) 운동을 고려한 반응동 현가시스템 모델이다[9]. 현가시스템 제어란 승차자가 편안함을 느끼고 차량의 주행성능이 안정하도록 \$c_i\$를 조절함으로써 상하, 경사, 회전 운동을 억제하는 것이다.

그림 4에서 기호의 의미는 \$k_i\$: 스프링 계수, \$c_i\$: 댐핑 계수, \$s_i\$: 서스펜션 스트로크, \$w_i\$: 도로 장애, \$x_i\$: 평행위치에서부터 타이어의 중심까지의 거리, \$M\$: 차량의 무게를 나타내며 \$y_1\$: 상하 운동, \$y_2\$: 경사, \$y_3\$: 회전운동을 나타낸다.

본 논문에서는 제어기 설계 방법에 따라 상하, 경사, 회전 운동과 차체 위치(frame distance, 그림 4에서 \$s_i + x_i\$를 지시)를 인식한다. 모델인식은 그림 5와 같은 Series-parallel 방법을 사용한다.

그림 5에서 full car 모델에서의 출력벡터는 \$y(t) = [y_1 \ y_2 \ y_3]^T\$를 나타내고 각 원소의 의미는 상하, 경사, 회전 운동을 각각 의미하고 있다. 또한, 입력벡터 \$u(t) =

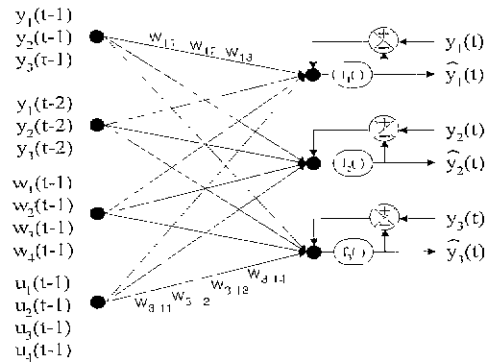


그림 6. Full Car 모델인식을 위한 신경회로망 구조.
Fig. 6. Neural networks structure for estimation of full car model.

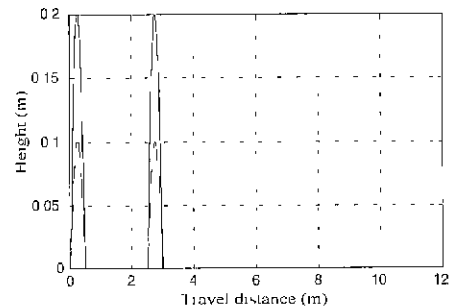


그림 7. 도로장애 #1 (좌측:0.2, 우측:0.1).
Fig. 7. Road disturbance #1 (Left:0.2, Right:0.1)

\$[u_1 \ u_2 \ u_3 \ u_4]^T\$. 왜란으로 발생하는 도로장애 \$w(t) = [w_1 \ w_2 \ w_3 \ w_4]^T\$이고 \$u_i\$는 \$c_i\$를 조절하는 제어값이며 \$w_i\$는 각각의 차륜에 가해지는 도로장애이다. 실제 차량은 각 입력과 출력관계가 2차 시스템으로 묘사된다고 간주하고 그 특성을 표현하기 위하여 신경회로망 입력으로 현재와 한 스텝 이전의 과거값을 인가한다. 이를 위한 신경회로망 구조는 그림 6과 같다.

모델 인식을 위한 신경회로망의 입력은 그림 6에서 나타낸바와 같이 full car 모델 출력 \$y(t-1), y(t-2)\$의 순환값과 제어입력 \$u(t-1)\$, 도로장애 \$w(t-1)\$이며 출력은 \$\hat{y}(t)\$이다. 신경회로망은 출력 \$\hat{y}(t)\$가 목표값 \$y(t)\$를 인식하도록 학습됨으로써 full car 모델을 인식하게 된다. 시뮬레이션의 초기 연결강도는 0~0.5로 선택하고 바이어스 입력은 \$e_{jk}(t-1)\$이며, \$w_{jk0} = \mu_0 = 0.05\$로 하였다. 도로장애는 그림 7에서와 같이 차량이 20Km/h로 진행하고, 좌측 전·후 차륜에는 0.2m 높이의 시그모이드 형태의 장애물이, 우측 전·후 차륜에는 0.1m 높이의 장애물이 통과하는 것으로 하였다. 좌우의 높이를 다르게 설정한 이유는 신경회로망이 차량의 회전 운동까지도 인식이 가능하도록 고려하기 위함이다.

그림 8은 학습오차의 감소형태를 나타내는 한 예로서 상하운동의 오차를 나타내었으며 시뮬레이션 결과 도로 굴곡을 지날 때 오차가 약간 발생하며 그 외의 구간에 대하여는 오차가 0인 것을 알 수 있다. 이러한 인식 성능

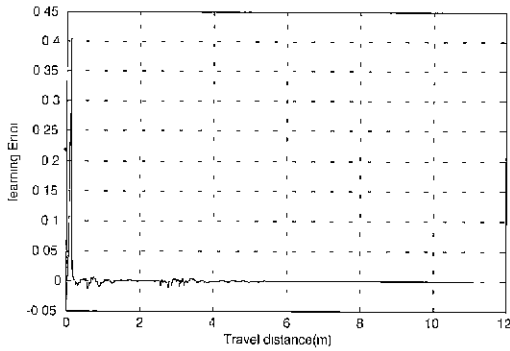


그림 8. 상하 운동의 학습 오차.
Fig. 8. Learning error of heave motion.

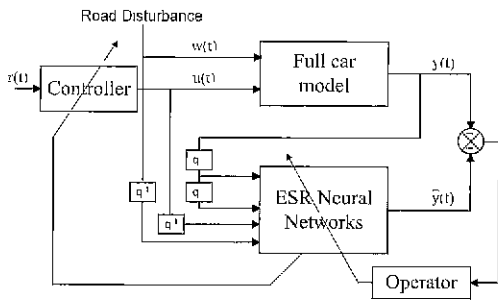


그림 9. 실시간 차량 모델 인식과 제어 방법도.
Fig. 9. The identification of vehicle in real time and control.

은 신경회로망을 기반으로 한 제어기 설계를 가능하게 한다.

IV. 신경회로망 기반 현가시스템 제어기

본 장에서는 참고문헌[6]에 제시된 오차 자기순환 신경회로망과 그 학습 방법을 이용하여 실시간 인식되어진 신경회로망 모델에 기반 하여 제어기를 설계하였다. 제어기를 설계하는 방법은 그림 9와 같은 방법을 사용하고 반능동 현가시스템 예측 적응제어기와 슬라이딩 모드 적응제어기를 구성한다.

1. 예측 적응제어기

예측 적응제어기는 미래의 출력값을 미리 예측하여 제어값을 출력하는 시스템의 변화에 적응성을 갖는 제어기이다. 제어기 구성을 위해 7자유도 차량 모델의 상하, 경사, 회전 운동을 추정하고, 추정된 모델을 동작점에서 선형화한 후 예측제어 기법을 이용하여 제어기를 구성한다

1) 인식된 신경회로망 차량 모델

모델의 상하, 경사, 회전운동을 추정한 신경회로망 차량 모델은 (11)과 같이 나타낼 수 있다.

$$\hat{y}(t+1) = f(\theta, \Phi) = f(\hat{A}y(t) + \hat{B}y(t-1) + \hat{C}u(t) + \hat{D}w(t) + 0.05e(t)) \quad (11)$$

여기에서 함수 f 는 신경회로망의 활성화함수를 나타내며 양극성(bipolar) 시그모이드 함수를 사용하고, $\Phi = [y(t), y(t-1), u(t), w(t)]^T$, $\theta = [\hat{A} \hat{B} \hat{C} \hat{D}]$ 이며, \hat{A} , \hat{B} , \hat{C} , \hat{D} 는 연결강도로 구성된 매트릭스로 다음과 같다

$$\hat{A} = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix}, \quad \hat{B} = \begin{bmatrix} w_{14} & w_{15} & w_{16} \\ w_{24} & w_{25} & w_{26} \\ w_{34} & w_{35} & w_{36} \end{bmatrix}$$

$$\hat{C} = \begin{bmatrix} w_{17} & w_{18} & w_{19} & w_{110} \\ w_{27} & w_{28} & w_{29} & w_{210} \\ w_{37} & w_{38} & w_{39} & w_{310} \end{bmatrix}, \quad \hat{D} = \begin{bmatrix} w_{111} & w_{112} & w_{113} & w_{114} \\ w_{211} & w_{212} & w_{213} & w_{214} \\ w_{311} & w_{312} & w_{313} & w_{314} \end{bmatrix}$$

(11)은 현재의 값으로 한 단계 이후의 예측된 출력값, 즉 미래값을 출력하게 된다. 따라서 예측 제어기를 설계할 수 있는 모델이 된다.

2) 예측 적응 제어기 설계

예측제어기 설계는 $u(t)$ 에 대하여 J 가 최소가 되도록 설계를 한다.

$$J(t+1) = \frac{1}{2} E[\|\hat{y}(t+1)\|_Q^2 + \|u(t)\|_R^2] \quad (12)$$

Q 와 R 은 각각 상태 가중치, 제어 가중치를 나타낸다. Φ_0 를 동작점으로 하고 신경회로망 모델 (11)를 테일러 급수(Taylor series)로 전개하면 (13)과 같이 나타낼 수 있다

$$\hat{y}(t+1) = f(\theta, \Phi_0) + \delta B(u(t) - u(t-1)) \quad (13)$$

단, $\Phi_0 = [y(t), y(t-1), u(t-1), w(t)]^T$

$$\delta B = \frac{\partial f(\theta, \Phi(t))}{\partial u(t)} \Big|_{\theta = \theta_0}$$

$$= \frac{\partial}{\partial u(t)} f(\hat{A}y(t) + \hat{B}y(t-1) + \hat{C}u(t) + \hat{D}w(t) + \hat{E}e(t)) \Big|_{u(t) = u(t-1)} \quad (14)$$

(14)를 자코비안(Jacobian)을 이용하여 자세히 전개하면 (15)와 같다

$$\delta B = \frac{\partial f(\theta, \Phi(t))}{\partial u(t)} \Big|_{\theta = \theta_0} \quad (15)$$

$$= \begin{pmatrix} \frac{\partial f_1}{\partial u_1(t)} & \frac{\partial f_1}{\partial u_2(t)} & \frac{\partial f_1}{\partial u_3(t)} & \frac{\partial f_1}{\partial u_4(t)} \\ \frac{\partial f_2}{\partial u_1(t)} & \frac{\partial f_2}{\partial u_2(t)} & \frac{\partial f_2}{\partial u_3(t)} & \frac{\partial f_2}{\partial u_4(t)} \\ \frac{\partial f_3}{\partial u_1(t)} & \frac{\partial f_3}{\partial u_2(t)} & \frac{\partial f_3}{\partial u_3(t)} & \frac{\partial f_3}{\partial u_4(t)} \end{pmatrix}$$

(11)은 3×1의 행렬로써 각각 상하, 경사, 회전 운동 값을 나타낸다 여기서 함수 f_1, f_2, f_3 는 출력 상하, 경사, 회전 운동 전단의 활성화함수를 나타내며, 예를 들어 f_1 의 의미를 나타내면

$$f_1(w_{11}y_1(t) + w_{11+3}y_1(t-1) + w_{11+6}u_1(t) + w_{11+10}w_1(t) + 0.05e_1(t)), \quad i=1\sim 3, j=1\sim 4$$

이다. (15) 각각의 원소들을 예를 들어 나타내면 (16)과 같다.

$$\frac{\partial f_1}{\partial u_1(t)} = \hat{C}(1,1)f_1(\hat{A}_1y(t) + \hat{B}_1y(t-1) + \hat{C}_1u(t) - \hat{D}_1w(t) + 0.05e_1(t))$$

$$\frac{\partial f_2}{\partial u_2(t)} = \hat{C}(2,2)f_2(\hat{A}_2y(t) + \hat{B}_2y(t-1) + \hat{C}_2u(t) + \hat{D}_2w(t) + 0.05e_2(t)) \quad (16)$$

$$\frac{\partial f_3}{\partial u_3(t)} = \hat{C}(3,3) \hat{f}_3(\hat{A}_3)x(t) + \hat{B}_3y(t-1) + \hat{C}_3u(t) + \hat{D}_3w(t) + 0.05e_3(t)$$

여기에서 각 행렬의 첨자는 그 행렬의 1행을 의미한다. $u(t)$ 를 구하기 위해 (12)를 편미분을 하면 (17)과 같다.

$$\frac{\partial K(t+1)}{\partial u(t)} = \left(\frac{\partial K(t+1)}{\partial u(t)} \right)^T Q \hat{y}(t+1) + Ru(t) \quad (17)$$

제어기는 (17)를 0으로 하는 $u(t)$ 로써 전개하여 정리하면 (18)과 같이 표현할 수 있다.

$$u(t) = -(R + \delta B^T Q \delta B)^{-1} \delta B^T Q (K(t+1) \theta_0 - \delta B u(t-1)) \quad (18)$$

(18)은 차량의 상하, 경사, 회전 운동을 제어하기 위한 4개의 현가시스템 제어출력을 만들어낸다.

3) 예측 적응제어기의 실험결과

설계한 적응 예측제어기의 성능을 평가하기 위한 신경회로망의 초기 연결강도는 0~0.5로 선택하고, 바이어스 입력은 $e_{jk}(t-1)$, $w_{j0} = \mu_0 = 0.05$ 로 하였다. 그림 7과 같은 도로장애 #1에 의한 상하, 경사, 회전 운동을 실시간 추정하고 예측 적응제어기를 적용하였을 때의 제어결과를 그림 10, 11, 12에 나타내었다. 점선은 제어기가 없는 수동 현가시스템의 경우를, 실선은 예측 적응제어기를 적용한 반능동 현가시스템의 제어결과이다.

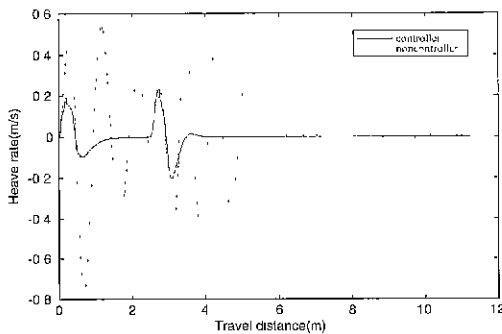


그림 10. 예측 적응제어에 의한 상하 움직임 (도로장애 #1).

Fig. 10. Heave motion of predictive adaptive control (Road disturbance #1)

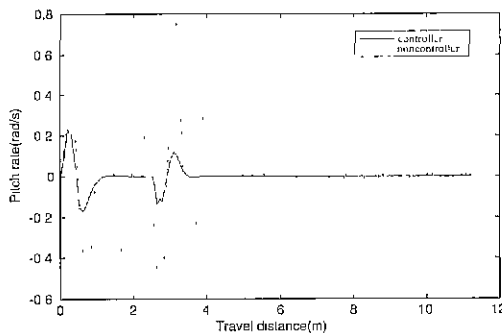


그림 11. 예측 적응제어에 의한 경사 움직임 (도로장애 #1).

Fig. 11. Pitch motion of predictive adaptive control (Road disturbance #1).

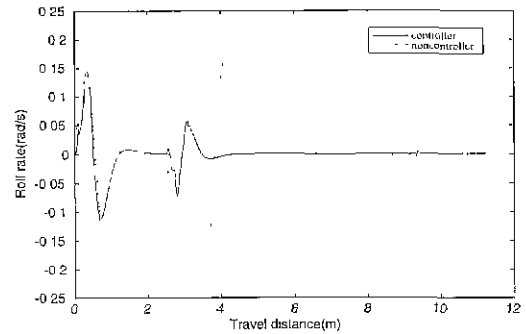


그림 12. 예측 적응제어에 의한 회전 움직임 (도로장애 #1).

Fig. 12. Roll motion of predictive adaptive control (Road disturbance #1).

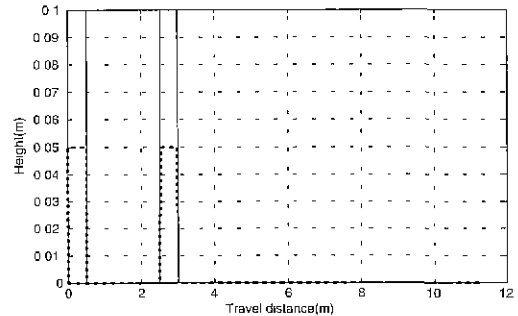


그림 13. 도로장애 #2 (좌측:0.1, 우측:0.05(m)).

Fig. 13. Road disturbance #2 (Left:0.1, Right:0.05(m)).

학습이 진행함에 따라 도로 굴곡을 지나갈 때의 차체의 진동이 처음에 비해 현저히 줄어들었다는 것을 알 수 있다. 그림 13과 같은 도로장애 #2에 대하여도 같은 실험을 하였다 그 결과를 주시하면 매우 많은 채터링 현상을 볼 수가 있는데 이는 제어기가 없는, 즉 수동 현가시스템의 경우에 대한 응답에서도 나타나는 현상으로 사각 파형의 굴곡을 지나갈 때 매우 빠르게 차량이 움직이기 때문에 발생하는 것으로 생각할 수 있다.

2. 적응 슬라이딩 모드 제어기

슬라이딩 모드 제어 기법은 강인 제어의 한 분야로서 임의의 모델 부정확성(uncertainty)에도 불구하고 좋은

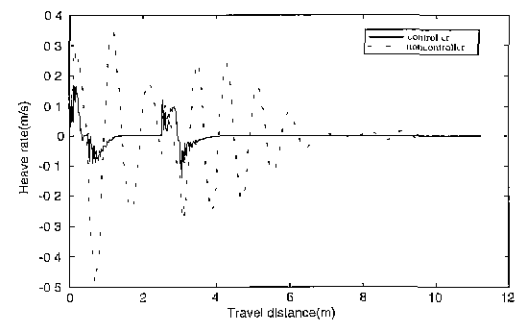


그림 14. 예측 적응제어에 의한 상하 움직임 (도로장애 #2)

Fig. 14. Heave motion of predictive adaptive control (Road disturbance #2).

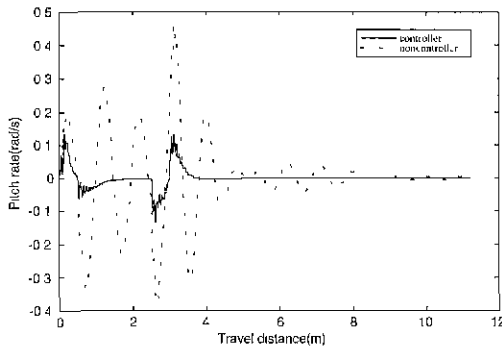


그림 15. 예측 적응제어에 의한 경사 움직임 (도로장애 #2).

Fig. 15. Pitch motion of predictive adaptive control (Road disturbance #2).

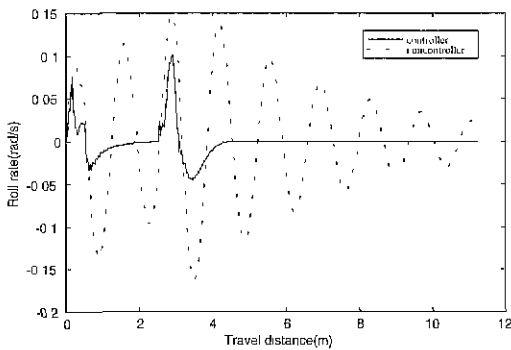


그림 16. 예측 적응제어에 의한 회전 움직임 (도로장애 #2).

Fig. 16. Roll motion of predictive adaptive control (Road disturbance #2).

제어 성능이 보장되며 빠른 응답 특성과 제어기 구조식 안에 비선형 항을 포함할 수 있다는 장점을 지니고 있다 [8]. 또한 높은 차수의 시스템을 1차의 시스템으로 묘사할 수 있는 장점이 있어 다루기가 쉽다.

슬라이딩 모드 제어는 단일입력 시스템에 대해 정의 되어 있기 때문에 본 연구에서는 full car 모델의 차체 위치를 각각 추정하고 이를 기초로 슬라이딩 모드 제어 기법을 응용하여 제어기를 구성하였다. 이용된 신경회로망의 구조와 모델 인식 방법은 상하, 경사, 회전 운동의 인식 방법과 동일하다. 그러나 슬라이딩 모드에 의하여 제어기를 구성하기 위해서는 입출력수가 동일하여야 하기 때문에 신경회로망의 출력과 모델의 출력을 각 차체 위치인 4개가 성립한다.

1) 인식된 신경회로망 차량 모델

슬라이딩 제어기법은 단일 입력을 가진 시스템, $\dot{x}^{(n)} = f(x) + b(x)u$ 와 같은 운동방정식에 대하여 정의되었다. 여기에서 x 는 상태벡터를 나타낸다. 슬라이딩 제어기법을 다입력·다출력 현가시스템에 적용하기 위해서는 시스템을 묘사하는 신경회로망 모델에서 입력벡터 $u(t)$ 의 원소로 제어되는 각각의 신경회로망 모델이 존재하여야 한다. 예를 들면 입력 $u(t) = [u_1 \ u_2 \ u_3 \ u_4]^T$ 에 대해 $\hat{y} =$

$[y_1 \ y_2 \ y_3 \ y_4]^T$ 이 존재해야 한다는 것이다 이를 위해, 예측 적응제어기에서 full car 모델로부터 상하, 경사, 회전 운동을 인식한 것과는 달리 슬라이딩 모드 제어기를 위해서는 4개의 출력 변수, 즉 차체 위치(frame distance, 그림 4에서 $s_i + x_i$ 를 지시함)를 인식한다. 시스템을 묘사하는 신경회로망 출력은 (19)와 같다.

$$\hat{y}(t) = f[\hat{A}y(t-1) + \hat{B}y(t-2) + \hat{C}u(t-1) + \hat{D}w(t-1) + \hat{E}e(t-1)] \tag{19}$$

여기에서의 각 파라미터는 1)절에 나타낸 것과 같고 다만, $y(t) = [s_1 + x_1 \ s_2 + x_2 \ s_3 + x_3 \ s_4 + x_4]^T$, \hat{y} 은 4×1의 차원을 갖는다.

2) 적응 슬라이딩 제어기 설계

적응 슬라이딩 제어기 설계를 위해 먼저 일반적인 단일 입력 동적 시스템을 (20)과 같이 나타낼 수 있다.

$$\dot{x}^{(n)} = f(x) + b(x)u \tag{20}$$

여기에서 좌측 전 차체의 현가시스템을 제어한다고 가정 하면 상태 x 는 신경회로망의 출력 \hat{y}_1 으로 나타낼 수 있으며 이는 앞에서 언급한 것과 같이 왼쪽 앞의 차체 위치($s_1 + x_1$)이다. 슬라이딩 제어는 시변(lime varying) 상태, $x_d = [x_d \ \dot{x}_d \ \ddot{x}_d \ \dots \ x_d^{(n-1)}]^T$ 를 상태 x 가 따라가도록 하는데 목적이 있다. 즉, (21)의 \tilde{x} 가 0이 되도록 하는 것이다. 슬라이딩 제어에 대해 기술하면 다음과 같은 식들로 나타낼 수 있다.

$$\tilde{x} = x - x_d = [\tilde{x} \ \dot{\tilde{x}} \ \ddot{\tilde{x}} \ \dots \ \tilde{x}^{(n-1)}]^T \tag{21}$$

(21)을 추정 오차 벡터라 하고 $s(x; t) = 0$ 이 되는 시변 평면 $S(t)$ 를 정의한다.

$$s(x; t) = \left(\frac{d}{dt} + \lambda \right)^{n-1} \tilde{x} \tag{22}$$

여기에서 λ 는 strictly positive 상수이다.

예를 들어 n=3이면 s 는 단순히 위치와 속도 오차의 합이 된다.

$$s = \tilde{x} + 2\lambda \dot{\tilde{x}} + \lambda^2 \tilde{x} \tag{23}$$

이렇게 되면 $x = x_d$ 가 되는 문제는 시변 평면 $S(t)$ 위에 시스템 궤적이 위치하도록 하는 문제로 귀결된다.

이때 $\tilde{x} = \frac{1}{p+\lambda} \times \frac{1}{p+\lambda} \times s$ 가 되므로 3차의 시스템이 1차의 모델로 간소화된다.

또한 s 를 0으로 유지하는 1차 문제는 (20)에서 제어 법칙 u 를 다음과 같은 조건에 맞게 설정하므로써 이를 수 있다.

$$\frac{1}{2} \frac{d}{dt} s^2 \leq -\eta |s| \tag{24}$$

(24)를 슬라이딩 조건(sliding condition)이라 하며 평면 $S(t)$ 와 시스템 궤적까지의 제어거리가 모든 시스템 궤적을 따라 감소한다는 것을 의미한다. 즉, 시스템 궤적이

평면 $S(t)$ 로 이동해서 그 평면 위에 시스템 궤적이 위치하게 되면 시스템 궤적들은 평면 위에 계속해서 남아있게 된다는 것을 의미한다. 여기에서 η 는 strictly positive constant이다.

7 자유도 full car 모델은 각 서스펜션을 2차 시스템으로 생각할 수 있으므로 2차 미분방정식으로 생각할 수 있다. 따라서 슬라이딩 제어를 설계하기 위한 식으로 (22)에 따라 (25)와 (26)과 같이 표현할 수 있다.

$$s = \left(\frac{d}{dt} + \lambda \right) \hat{x} = \dot{\hat{x}} + \lambda \hat{x} \quad (25)$$

$$\dot{s} = \dot{x} - \ddot{x}_d + \lambda \dot{\hat{x}} = f + u - \dot{x}_d + \lambda \dot{\hat{x}} \quad (26)$$

$s=0$ 으로 하는, 즉 오차들의 변화량을 0으로 하는 최적의 추정값 \hat{u} 를 구하면 (27)과 같다.

$$\hat{u} = -\dot{f} + \dot{x}_d - \lambda \dot{\hat{x}} \quad (27)$$

그러나 실제의 u 값과는 모델링 차이가 있으므로 제어법칙 u 는 (28)과 같다

$$u = \hat{u} - k \cdot \text{sgn}(s) \quad (28)$$

단,
$$\text{sgn}(s) = -1, \text{ if } s > 0$$

$$\text{sgn}(s) = 1, \text{ if } s < 0$$

(26)에서 $k = F + \eta$, $|f - \dot{f}| \leq F$ 로 정의되며 f 는 이미 알고 있는 함수 $F = F(x, \dot{x})$ 으로 제한된다. 본 논문에서는 F 를 정확하게 정의할 수 없으므로 최적의 제어 성능을 위하여 (30)의 파라미터와 같이 선택하였다.

(28)을 스위칭 제어 법칙(switching control law)라 하는데 이 스위칭 제어법칙으로 제어할 경우에는 $k \cdot \text{sgn}(s)$ 의 영향으로 인하여 채터링 현상이 발생한다 채터링 현상을 없애주기 위하여 경계층(boundary layer)을 설정해 주면 제어기 u 는 (29)가 된다.

$$u = -\dot{f} + \dot{x}_d - \lambda \dot{\hat{x}} - k \cdot \text{sat}\left(\frac{s}{\phi}\right) \quad (29)$$

(29)는 단일 입·출력 시스템에 대한 제어기이므로 이를 다입력·다출력 현가시스템에 적용하여 다시 기술하면 다음과 같이 기술 할 수 있다.

$$u(t) = -\dot{f} + \dot{x}_d - \lambda \dot{\hat{x}} - k \cdot \text{sat}\left(\frac{s}{\phi}\right) \quad (30)$$

여기에서

$$\hat{f} = \hat{y}(t+1) = \hat{A}\hat{y}(t) + \hat{B}y(t-1) + \hat{D}u(t) + \hat{E}e(t)$$

$$\ddot{x}_d = \dot{x}_d = 0, \lambda = 50$$

$$k = F + \eta = 10 \times |\hat{y}(t)|^2 + 0.1, \phi = 40 \text{ 이다.}$$

$$\text{sat}\left(\frac{s}{\phi}\right) = \frac{s}{\phi}, \text{ if } \left|\frac{s}{\phi}\right| \leq 1$$

$$\text{sat}\left(\frac{s}{\phi}\right) = \text{sgn}\left(\frac{s}{\phi}\right), \text{ if } \left|\frac{s}{\phi}\right| > 1 \text{ 이다.}$$

\hat{x} 는 $y(t+1)$ 의 직접 측정이 불가능하므로 인식되어진 신경회로망 모델로 대체하여 (31)로 구하게 된다. (31)은 $\Delta y/\Delta t$ 의 의미다.

$$\hat{x}(t+1) = \frac{y(t+1) - y(t)}{h} = \frac{\hat{y}(t+1) - y(t)}{h}$$

$$\hat{y}(t+1) = \hat{A}\hat{y}(t) + \hat{B}y(t-1) + \hat{C}u(t) + \hat{D}w(t) + \hat{E}e(t) \quad (31)$$

(30)과 (31)에서 \hat{f} 와 $\hat{y}(t+1)$ 은 한 단계 이후의 출력값, 즉 시스템의 미래 출력값을 나타낸다. 따라서 제어기는 한 단계 이후의 출력을 미리 예측하고 제어출력을 만들어 내는 예측제어의 기능도 가지게 된다.

3) 적응 슬라이딩 제어기의 실험결과

시뮬레이션은 현가시스템을 매 샘플 순간마다 인식하고 인식된 신경회로망 모델을 기반으로 하여 식 (29)에 따라 제어입력을 계산한다. 따라서 기존 뉴로 제어기(neuro controller)와는 달리 인식과 제어가 동시에 실시간으로 이루어지게 된다. 시뮬레이션 과정으로 각 차체 위치를 신경회로망으로 추정하고 그에 따른 제어 출력을 발생하였다. 제어 성능을 평가하기 위하여 각 차체 위치의 움직임 대신에 상하, 경사, 회전 운동을 출력하였다. 그림 17, 18, 19는 그림 7과 같은 도로장애 #1에 의한 제어 결과를 보여준다. 또한 실험적으로 제어기의 적응능력을 평가하기 위하여 주행중 차량의 속도를 증가하여 실험하였다.

다음은 도로장애 #2에 대한 적응슬라이딩 제어기의 실험결과이다.

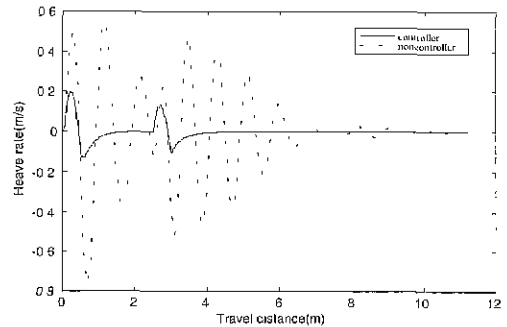


그림 17. 적응 슬라이딩 제어에 의한 상하 움직임 (도로장애 #1)

Fig. 17. Heave motion of adaptive sliding control (Road disturbance #1).

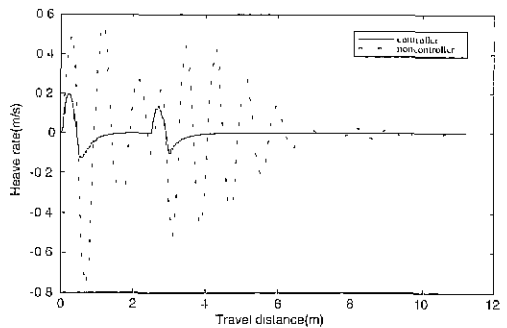


그림 18. 적응 슬라이딩 제어에 의한 상하 움직임 (도로장애 #1).

Fig. 18. Heave motion of adaptive sliding control (Road disturbance #1).

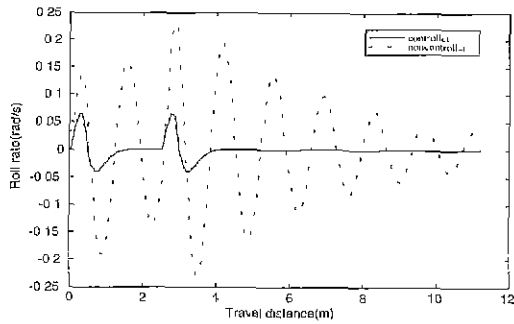


그림 19. 적응 슬라이딩 제어 의한 회전 움직임 (도로 장애 #1).
Fig. 19. Roll motion of adaptive sliding control (Road disturbance #1).

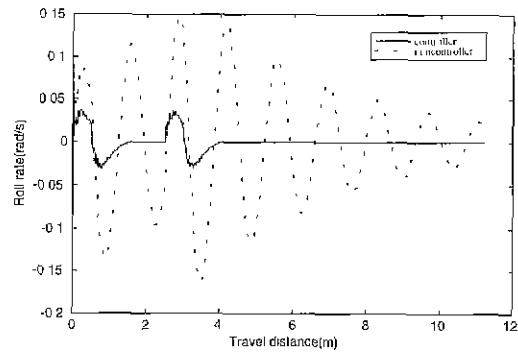


그림 22. 적응 슬라이딩 제어에 의한 회전 움직임 (도로장애 #2).
Fig. 22. Roll motion of adaptive sliding control (Road disturbance #2).

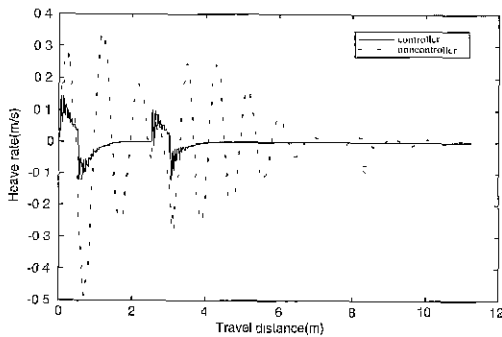


그림 20. 적응 슬라이딩 제어에 의한 상하 움직임 (도로장애 #2).
Fig. 20. Heave motion of adaptive sliding control (Road disturbance #2).

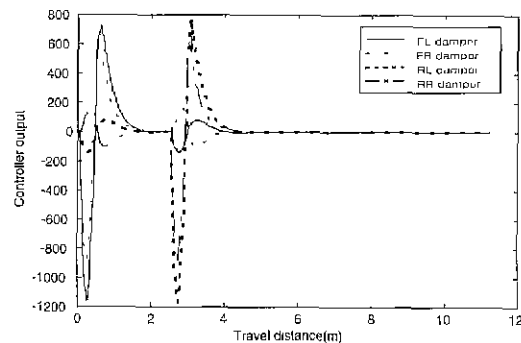


그림 23. 적응 슬라이딩 제어기의 제어출력 (도로장애 #1).
Fig. 23. Control output of adaptive sliding controller (Road disturbance #1).

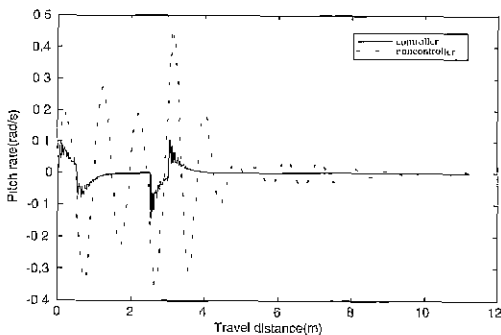


그림 21. 적응 슬라이딩 제어에 의한 경사 움직임 (도로장애 #2).
Fig. 21. Pitch motion of adaptive sliding control (Road disturbance #2).

그림 23은 도로장애 #1에 대한 적응 슬라이딩 제어기의 제어신호의 모양을 보여주고 있다. 제어신호는 차량의 각 움직임이 목표값이 되도록 서스펜션을 조절하도록 발생된다. 제어신호 형태는 각 도로조건에 따라 다양한 유형이 될 것이며, 신경회로망을 이용한 시스템 인식에 충분한 여자신호로 작용됨을 알 수 있다

시뮬레이션 결과 전체적으로 적응 슬라이딩 제어기의 성능이 좋음을 알 수 있다. 또한 사각 파형의 도로 장애에 대한 결과에 있어서는 모두다 체터링 현상을 보이고

있다. 앞서서도 언급한 것과 같이 사각 파형의 도로장애에 대하여는 수동 현가시스템보다 많은 체터링 현상을 보이고 있다. 이러한 이유는 제어기가 차체의 각 움직임을 최소화 하기 위하여 제어를 수행하기 때문에 일어나는 것으로 생각 할 수 있다.

V. 결론

본 논문에서는 7 자유도 차량 현가시스템 제어를 위해 신경회로망 기반 적응 반응동 현가시스템 제어기를 제안하였다. 제안한 방법은 오차 자기 순환 신경회로망을 이용하여 온라인으로 시스템을 인식하고 인식된 신경회로망 모델을 기반으로 제어기를 설계한다. 제안한 방법은 기존 뉴로 제어기와는 달리 온라인으로 시스템인식과 제어가 이루어지므로 시스템의 변화에 따른 재학습이 필요하지 않으며 곧바로 변화에 적응할 수 있고 시간이 지날수록 정확한 시스템 인식으로 인하여 제어기가 더욱 안정되며 여러 다른 제어기법을 적용할 수 있다는 장점을 가지고 있다. 제어기 구성을 위해 예측 제어 기법과 슬라이딩 모드 제어기법을 응용하여 적용하였고 시그마이드와 사각 파형의 도로장애에 대한 시뮬레이션을 통하여 제안한 제어기가 반응동 현가시스템 제어에 효과적임을 밝혔다.

또한 차량의 파라미터 변화에 따른 적응성이 있음을

보였다. 제안한 제어기 설계방법은 신경회로망 모델을 기반으로 하기 때문에 모델의 변경에 따른 적응성은 가지고 있으나 선형 또는 비선형 제어기법을 이용하기 때문에 제어기 자체에는 학습능력이 없다고 볼 수 있다. 따라서 제어기 구성시 신경회로망을 부가하여 학습능력을 갖도록 구성한다면 더 나은 제어 결과를 가질 수 있을 것이라 생각된다. 또한 신경회로망의 파라미터에 많이 의존하는 예측 적응 슬라이딩 제어기의 경우에도 심각한 이형의 도로장애에 대하여 민감한 특성을 보인다. 따라서 이러한 문제점도 신경회로망을 부가하여 해결할 수 있을 것으로 생각된다.

참고문헌

[1] K. S. Narendra, K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Networks*, vol. 2, no. 1, March 1990.
 [2] H. Wabgaonkar, and A. Stubberud, "A system identification technique based on neural networks," *Proc. IEEE Int. Conf. Systems Engineering*, Kobe, Japan, pp. 352-356, Sept. 1992.

[3] D. E. Rumelhart, and J. L. McClelland, "Parallel distributed processing," vol 1, *MIT Press*, Cambridge, MA, 1986.
 [4] R. S. Scalerio, and N. Tepedelenliouglu, "A fast new algorithm for training feedforward neural networks," *IEEE Trans. Signal Processing*, vol. 40, pp. 202-210, 1992.
 [5] S. Haykin, *Neural Networks : A Comprehensive foundation*, Macmillan College Publishing Company Inc., NY, 1994.
 [6] 이창구, "오차순환 신경회로망을 이용한 빠른 시변 비선형 시스템의 인식과 제어," *대한전기학회 논문지*, vol. 46, no 12, pp. 1793-1799, 1997.
 [7] X. Ma and N K. Loh, "One-step-ahead Controller design using neural networks." 1992 ACC/WP2, pp. 958-962, 1992
 [8] Jean-Jacques E. Slotine Weiping Li, *Applied Nonlinear Control*. Prentice Hall, New Jersey, 1991.
 [9] 노태수, 정길도, 홍동표, "7DOF 차량 모델을 이용한 자동차 현가장치 동역학 해석 및 시뮬레이션에 관한 연구," *한국자동차공학회 96년 춘계학술대회 논문집*, pp. 212-216, 1996.



이 창 구

1958년 12월 25일생. 1981년 전북대 전기공학과 졸업(공학사). 1991년 동대학원 전기공학과 졸업(공학박사). 1983년-1992년 한국전자통신연구소 선임연구원. 1996년 1월-12월 Alberta대학방문교수. 현재 전북대학교 전자정보공학부 부교수. 관심분야는 지능제어, 공정제어, 마이크로프로세서 응용.



송 광 현

1974년 3월 27일생. 1996년 전북대 제어계측공학과 졸업(공학사). 1998년 전북대 전기공학과 졸업(공학석사). 관심분야는 지능제어, 비선형시스템 제어.