

두개의 Extended Kalman Filter를 이용한 Recurrent Neural Network 학습 알고리즘

A Learning Algorithm for a REKFN Neural Network Base on Dual Extended Kalman Filter

*송명근, **김상희, ***박원우

Myung Geun Song, Sang Hee Kim, Won Woo Park

Abstract - The classical dynamic backpropagation learning algorithm has the problems of learning speed and the determine of learning parameter. The Extend Kalman Filter(EKF)is used effectively for a state estimation method for a non linear dynamic system. This paper presents a learning algorithm using Dual Extended Kalman Filter(DEKF) for Fully Recurrent Neural Network(FRNN). This DEKF learning algorithm gives the minimum variance estimate of the weights and the hidden outputs. The proposed DEKF learning algorithm is applied to the system identification of a nonlinear SISO system and compared with dynamic backpropagation learning algorithm.

Key Words : Extended Kalman Filter, Fully Recurrent Neural Networks

1. 서론

인공 신경망은 디지털 신호처리, 시스템 동정화(Identification) 및 제어, 패턴 분류 및 인식 등에 널리 이용되고 있다. 신경망은 비선형 시스템에 대한 적응성 등 많은 장점을 가지고 있으나 기존의 역전파 방식은 학습 속도가 느리고 학습률 등의 학습 요소의 값들을 임의로 정해야하는 어려움을 가지고 있다. 신경망의 효과적인 학습 알고리즘의 개발을 위하여 비선형 이산 시스템의 상태를 효과적으로 예측할 수 있는 Extended Kalman Filter(EKF)를 이용하였다. Dual Extended Kalman Filter(DEKF) 알고리즘을 이용하여 Fully Recurrent Neural Network(FRNN)의 은닉층 및 출력층의 각 연결 강도와 은닉층의 상태를 예측하여 학습시켰다. [1]

본 논문은 이를 비선형 시스템 동정화에 적용시키고 기존의 역전파 방식의 FRNN과 비교하였다.

2. Fully Recurrent Neural Network 학습 알고리즘

2.1 Extended Kalman Filter

Standard Kalman Filter는 선형시스템의 상태예측에 재귀적 해를 제공한다. 이를 비선형 시스템으로 확장하여 시스템을 임의적으로 선형화한 후 오차 분산과 평균을 구해 Kalman filter에 적용한 것이 EKF이다.

비선형 시스템 상태 방정식이 다음과 같이 주어지면

$$\vec{x}(t+1) = f(\vec{x}(t)) + \vec{w}(t) \quad (1)$$

$$\vec{y}(t) = h(\vec{x}(t)) + \vec{v}(t) \quad (2)$$

여기서 $\vec{x}(t)$ 는 $(n \times 1)$ 상태 벡터이고 $\vec{y}(t)$ 는 $(m \times 1)$ 관측 벡터이다. f 와 h 는 비선형 시변 함수이다. $\vec{w}(t)$ 와 $\vec{v}(t)$ 는 가우시안 백색 잡음이고 잡음의 분산은 $Q(t)$ 와 $R(t)$ 로 주어지고 다음 식과 같이 나타낸다.

$$Q(t) = R(t) = \eta I \quad (3)$$

여기서 I 는 단위행렬이고 η 는 임의의 작은 상수이다. EKF의 상태예측은 다음 식과 같이 계산된다.

$$\hat{\vec{x}}(t|t) = \hat{\vec{x}}(t|t-1) + K(t)[\vec{y}(t) - h(\hat{\vec{x}}(t|t-1))] \quad (4)$$

$$\hat{\vec{x}}(t+1|t) = f(\hat{\vec{x}}(t|t)) \quad (5)$$

$$K(t) = P(t|t-1)H(t)^T \cdot [R(t) + H(t)P(t|t-1)H(t)^T]^{-1} \quad (6)$$

$$P(t|t) = P(t|t-1) - K(t)H(t)P(t|t-1) \quad (7)$$

$$P(t+1|t) = F(t)P(t|t)F(t)^T + Q(t) \quad (8)$$

여기서 $(n \times n)$ 행렬 $P(t)$ 는 오차분산이고 $(m \times n)$ 행렬 $K(t)$ 는 Kalman Gain이라 부른다. 비선형 시스템 방정식을 선형화한 자코비안 $(n \times n)$ 행렬 $F(t)$ 와 자코비안 $(m \times n)$ 행렬 $H(t)$ 그리고 오차 분산 $P(t)$ 는 다음과 같이 정의 된다. [3]

저자 소개

* 準 會 員 : 金 烏 工 科 大 學 電 子 學 科 碩 士 課 程

** 正 會 員 : 金 烏 工 科 大 學 電 子 學 科 教 授 · 工 博

*** 正 會 員 : 金 烏 工 科 大 學 電 子 學 科 教 授 · 工 博

$$F(t) = \left. \frac{\partial f(x)}{\partial x} \right|_{x=\hat{x}(t)} \quad (9)$$

$$H(t) = \left. \frac{\partial h(x)}{\partial x} \right|_{x=\hat{x}(t-1)} \quad (10)$$

$$P(t) = E\{(\hat{x}(t) - \hat{x}(t|t-1)) \cdot (\hat{x}(t) - \hat{x}(t|t-1))^T\} \quad (11)$$

EKF의 예측과정은 크게 Time Update와 Measurement Update과정으로 나눌 수 있다. Time Update에서는 $\hat{x}(t+1|t)$ 와 $P(t+1|t)$ 를 계산하고 Measurement Update에서는 $K(t)$ 과 $\hat{x}(t|t)$ 그리고 $P(t|t)$ 를 계산한다. 각 과정은 그림1과 같은 관계를 가진다. [1, 2, 3]

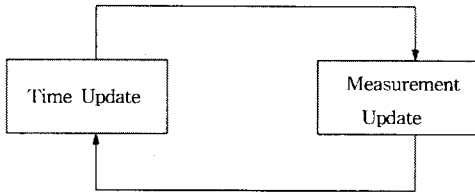


그림 1. EKF 동작도

2.2 Fully Recurrent Neural Network

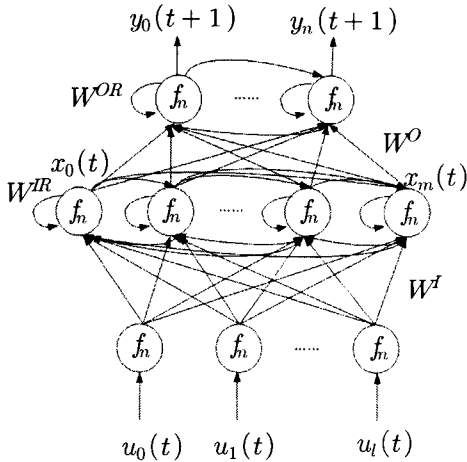


그림 2. Fully Recurrent Neural Network 구조

FRNN의 구조는 그림 1과 같으며 그림에서 \vec{u} 는 (1×1) 의 입력벡터이고 $\vec{x}(t)$ 는 $(m \times 1)$ 의 은닉층의 출력 벡터이다. 그리고 $\vec{y}(t)$ 는 $(n \times 1)$ 의 출력층의 출력 벡터이다. W^I 는 입력층과 은닉층 사이의 연결강도이고 W^O 는 은닉층과 출력층 사이의 연결강도이다. W^m 은 은닉층의 각 뉴런들 사이의 연결강도이고 W^{OR} 은 출력층의 각 뉴런들 사이의 연결강도이다. 입력에 대한 각층의 출력은 아래의 식과 같이 정리된다.

$$\vec{x}(t) = f_n(W^m \vec{u}(t) + W^{OR} \vec{x}(t-1)) \quad (12)$$

$$\vec{y}(t) = f_n(W^O \vec{x}(t) + W^{OR} \vec{y}(t-1)) \quad (13)$$

여기서 함수 $f_n(\cdot)$ 는 Sigmoid 함수이고 다음과 같이 정의된다.

$$f_n(x) = 1 + \tanh(x/U_0) \quad (14)$$

식(14)에서 U_0 는 Sigmoid 함수의 기울기이고 함수 $f_n(\cdot)$ 의 미분은 다음과 같이 계산된다.

$$f'_n(x) = f_n(x)(1 - f_n(x))/U_0 \quad (15)$$

2.3 Dual Extend Kalman Filter Algorithm

DEKF 알고리즘은 두개의 EKF를 사용하여 은닉층의 출력과 각 연결강도의 다음 상태를 예측함으로써 신경망을 학습시킨다. 은닉층의 출력을 \hat{x} 그리고 각 연결강도를 $\vec{W} = \{W^I, W^m, W^O, W^{OR}\}$ 와 같이 정의할 때 DEKF 알고리즘의 동작 과정은 다음 그림 3과 같다.

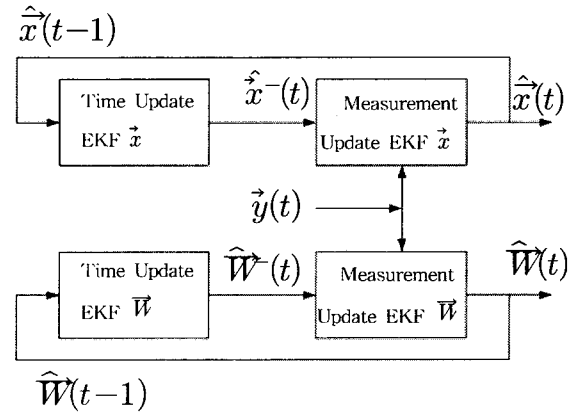


그림 3. DEKF 동작도

은닉층의 출력의 예측은 식 (3)에서 식(7)과 같이 구해지며 연결강도의 예측은 다음 식과 같이 구해진다.

$$H^m(t) = \left. \frac{\partial \vec{y}}{\partial \vec{W}} \right|_{\vec{W}=\vec{W}(t-1)} \quad (16)$$

$$K^m(t) = P^m(t-1)H^m(t) \cdot (I + H^m(t)^T P^m(t) H^m(t))^{-1} \quad (17)$$

$$\vec{W}(t) = \vec{W}(t-1) + K^m(t)(\vec{y}(t) - \vec{y}(t)) \quad (18)$$

$$P^m(t) = P^m(t-1) - K^m(t)H^m(t)^T P^m(t-1) \quad (19)$$

여기서 \vec{y} 는 신경망의 교사 출력벡터이고 \vec{y} 는 신경망의 현재 출력벡터이다. [2]

3. 실험 및 결과

아래의 식 (20)와 같은 SISO시스템에 대하여 식 (21)과 같은 입력을 주었을 때

$$y(t+1) = \frac{y(t)}{1+y(t)^2} + u(t)^3 \quad (20)$$

$$u(t) = \sin(2\pi t/25) \quad (21)$$

DEKF 알고리즘과 역전파 방식인 Dynamic Recurrent Neural Network(DRNN)의 500회 학습후의 시스템 동조화 결과와 그림 4와 같이 나타난다.

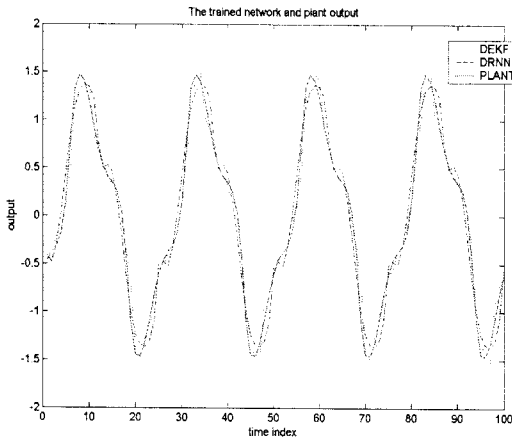


그림 4. DEKF와 DRNN의 시스템 동조화 결과

DRNN은 학습률 0.4로 학습 시켰으며 DEKF는 오차분산의 초기값을 10으로 하고 잡음 상수 η 는 0.001로 하였다. 두 경우 모두 Sigmoid함수의 기울기는 0.4로 하였다.

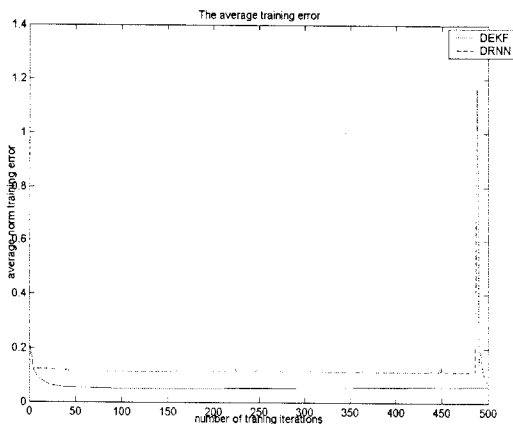


그림 5. DEKF와 DRNN의 평균 오차

그림 5는 학습 횟수에 따른 DEKF와 DRNN의 평균 에러의 변화를 비교한 그래프이다.

4. 결론 및 향후 연구과제

기존의 역전파 방식에 비해 DEKF 학습 알고리즘은 학습 속도가 빠르고 학습률 등을 조정하지 않아도 되는 장점이 있다. 그러나 계산량이 많아 총 학습시간이 느리고 비선형 시스템의 동특성을 충분히 학습 하지 못하는 단점을 가지고 있다. 향후에는 이러한 문제점을 해결하기 위하여 Unscented Kalman Filter를 이용한 방법을 연구해 보고자 한다.

본 연구는 한국과학재단 지정 금오공과대학교 전자부품 산업 고품질화 및 공정자동화연구센터의 지원에 의한 것입니다.

참 고 문 헌

- [1] Simon Haykin, "Kalman Filtering and Neural Networks", Wiley-Interscience.
- [2] Chi-Sing Leung, Lai-Wan Chan., "Dual extended Kalman filtering in recurrent neural networks", Elsevier Science Ltd. Neural Network, pp. 223-239, 2003.
- [3] Gintaras V.Puskorius, Lee A. Feldkamp, "Neurocontrol of Nonlinear Dynamical Systems with Kalman Filter Trained Recurrent Networks", IEEE Trans, Neural Networks, VOL 5, No 2, March 1994.
- [4] Kumpatis, Narendra, "Identification and Control of Dynamical Systems Using Neural Networks", IEEE Trans, Neural Networks, VOL 1, No 1, March 1990.