

CMAC을 이용한 구조물의 동적응답 예측

Prediction of Dynamic Response of Structures Using CMAC

김 동 현¹⁾ · 김 현 태²⁾ · 이 인 원³⁾

Kim, Dong Hyawn Kim, Hyon Taek Lee, In Won

요 약 : CMAC을 이용하여 구조물의 지진응답을 예측하였다. CMAC은 매우 빠른 학습능을 가지고 있는 것이 장점이며 구조물의 동적응답을 학습함에 있어서도 수 초 이내에 만족할 만한 정도로 학습을 끝낸다. 따라서 실시간 학습을 필요로 하는 분야에 매우 효과적으로 사용될 수 있다. 실시간 응답학습은 장기거동 등으로 역학적 특성이 변하거나 손상을 입은 구조물의 적응제어 등이 있다. 수치해석에서는 3층 전단건물의 지진응답을 CMAC을 통하여 학습하였으며 학습은 매우 빠르게 완수 되었다. 결론적으로 CMAC은 구조물의 진동제어 분야에서 매우 효과적으로 사용될 수 있는 인공지능의 하나이다.

ABSTRACT : Cerebellar model articulation controller (CMAC) is introduced and used for the identification of structural dynamic model. CMAC has fascinating features in learning speed. It can learn structural response within a few seconds. Therefore it is suitable for the real time identification structures. Real time identification is required in the control of structure which may be damaged or undergo severe change in mechanical properties due to shrinkage or relaxation etc. In numerical examples, it is shown that CMAC trained with the dynamic response of three-story building can predict responses under not trained earthquakes with allowable error. Finally, CMAC has great potential in structural and control engineering.

핵심용어 : CMAC, 응답예측, 학습, 동적 응답, 해쉬 매핑, 잡음.

KEYWORDS : CMAC, System Identification, Training, Dynamic Response, Hash Mapping, Noise.

1. 서론

1990년대에 들어서 인공신경망은 토목공학 분야에서 매우 성공적으로 적용되기 시작했다. 특히, 신경망은 구조물의 동적특성을 알 수 없는 경

우의 모델링이나 진동제어 분야에서 괄목할 만한 업적을 이루어 내는데 기여한 바가 크다. (J. Ghaboussi 등 1995; H. M. Chen 등 1995). 제어를 설계하기 위해서는 우선 제어하고자 하는 구조물의 수학적 모델을 구하는 작업이 선행되

1) 한국과학기술원 토목공학과, 박사후 연구원
2) 한국과학기술원 토목공학과 박사과정
3) 정회원, 한국과학기술원 토목공학과 교수, 공학박사

본 논문에 대한 토의를 2001년 4월 30일까지 학회로 보내 주시면 토의 회답을 게재하겠습니다.

어야 한다. 만일, 구조물의 수학적 모델이 불확실성으로 인해 오차를 포함하게 된다면 이를 근거로 설계된 제어기의 성능이 저하될 뿐만 아니라 심지어는 불안정한 특성을 보일 수도 있다. 그러나, 신경망을 이용한 진동제어에서는 이러한 수학적 모델이 필요 없다. 제어하고자 하는 구조물은 신경망에 의하여 식별하고 제어도 마찬가지로 신경망으로 수행한다. 이처럼 신경망은 불확실성을 내포한 구조물의 진동제어에 있어서 매우 획기적인 도구로서 자리잡고 있는 것이다.

그런데, 대부분의 신경망을 이용한 제어나 식별 문제에 있어서 구조의 단순성과 학습의 용이성으로 인해 거의 대부분 이른바 다층신경망(the multi-layer perceptron)이 사용되고 있다. 다층신경망은 하나의 외부신호의 학습을 위해서 사용되는 모든 가중치를 일괄적으로 학습한다. 다시 말해 어떤 학습자료에 의해 이미 학습이 되어진 가중치라 하더라도 다른 학습자료에 의해 다시 수정되어야 한다는 것이다. 이러한 방법의 일종의 전역학습법은 비교적 적은 학습자료로도 학습되지 않은 부분까지도 예측할 수 있는 일반화의 특성을 보이는 것이 장점이다. 그러나 이 때문에 학습에 소요되는 시간이 매우 길게 소요된다는 것이 피할 수 없는 단점이다. 따라서 이러한 방법의 학습으로는 실시간 응용의 필요성이 있는 문제에서는 적용이 불가능하게 된다. 실시간 제어나 식별문제의 적용을 위해서는 학습속도가 매우 빠른 다른 종류의 신경망이 필요하다.

구조물의 식별을 위한 신경망에 관한 연구자로는 우선 J. Ghaboussi(1995) 등과 H. M. Chen (1995) 등이 있다. 이들은 제어기 신경망을 학습시키기 위해 필요한 일종의 동적모델링 도구로서 emulator 라는 다층신경망을 사용하였다. 그들은 다층신경망이 충분히 우수한 결과를 보여준다고 발표하였다. 그러나 구조물의 동적응답을 비동기로 학습하는데 사용된 시간을 보면 59,200 에 포크이었다. 다시 말해 시간으로 계산하면 수십분 내지 수시간이 걸렸다는 말이 된다. J. Ghaboussi

등의 연구에서는 비동기 학습에 사용된 시간을 보여주지는 않았다. 이는 그들이 사용한 방법이 실시간으로 이루어지지 않는 것이기 때문에 소요시간이 큰 관심사가 되지 못했기 때문이다. 그들 이후로도 많은 연구자들이 이 분야의 연구결과를 발표하였다. (H. M. Chen et al. 1995; S. D. Snyder et al. 1995; Y. Tang 1996; K. Nikzad et al. 1996; K. Bani-Hani et al. 1998). 거의 모든 연구들은 다층신경망을 비동기로 학습한 것이고 실시간 학습을 위해서 적합하지 않은 것들이다.

본 연구에서는 이러한 문제점을 해결할 도구로서 신경망의 일종인 CMAC(cerebellar model articulation controller)을 소개하고자 한다. CMAC은 1975년 J. S. Albus에 의해 기계장치(manipulator)의 제어를 위해 처음 제안되었다. 그 이후 여러 연구자들에 의해 개선되었지만 CMAC이 가지고 있는 가장 큰 장점인 학습속도가 빠르다는 특성은 그대로 유지되고 있다. (S. H. Lane et al. 1992; J. Nie et al. 1993; C. T. Chiang et al. 1996; F. J. Gonzalez-Serrano et al. 1998; H. Hama et al. 1998). 이 때문에 CMAC이 주로 이용된 분야로는 기계나 전자장치의 진동제어 등 주로 매우 빠른 학습과 동작이 필요한 것들이었다. (M. Michael et al. 1994; K. M. Koo et al. 1994).

토목분야에서 CMAC을 처음 이용한 연구자로는 S. L. Hung (1999) 등이며 이들은 변형된 구조를 가진 Macro Structure CMAC (MS_CMAC)을 이용하여 강구조물의 설계변수를 학습하였다. 그들의 연구에서도 CMAC의 학습속도가 매우 빠름의 입증되었다.

본 연구에서는 CMAC을 이용하여 구조물의 동적 응답예측을 수행하였다. 한 시간단계 전의 응답과 외력정보를 이용해서 다음 시간단계에서의 구조물의 응답을 예측하도록 하였다. 이 때 구조물의 수학적 모델은 수치해석을 위해서만 사용하

고 응답을 예측함에 있어서는 전혀 사용하지 않는 것으로 하였다. 수치해석으로는 3층 전단 건물을 이용하였다. 임의의 지진기록을 이용하여 구조물의 응답을 학습한 후 학습에 사용되지 않은 지진 응답도 예측하였다. 예측된 응답은 거의 오차가 없었고 측정잡음을 가진 입력으로도 응답 예측에 있어서 큰 오차가 포함하지 않았다.

2. CMAC (Cerebellar Model Articulation Controller)

하나의 입력에 대해서 CMAC은 매우 국부적인 메모리만을 활성화시키고, 활성화된 메모리에 저장된 값을 이용하여 출력을 만든다. 저장된 가중치의 학습도 역시 국부적으로 이루어진다. 따라서, 매번 입력에 대해 전체 가중치를 모두 학습하는 다층신경망에 비해 학습속도가 매우 빠르며 국부적으로 매우 큰 비선형 특성도 학습이 가능하다는 장점이 있다. 물론 국부적인 가중치를 사용하므로 인해서 사용되는 가중치의 총 개수는 다층신경망에 비해 많다. 그래서 CMAC은 가중치를 줄이기 위해 해쉬사상(hash mapping) 이라 불리는 특별한 방법으로 입력과 실제 저장된 가중치 사이에 상호관계를 사용한다. 다음에서는 이러한 사상을 단계적으로 설명한다.

CMAC은 테이블 참조방식의 일종인 메모리 사상 기능을 가진 비선형 함수이다. J. S. Albus는 1975년 인간의 소뇌가 국부적인 학습기능을 수행하는 것에 착안해서 CMAC의 기본적인 구조를 제안하였다. 하나의 입력에 대해서 CMAC은 매우 국부적인 메모리만을 활성화시키고, 활성화된 메모리에 저장된 값을 이용하여 출력을 만든다. 또한 저장된 가중치의 학습 역시 국부적으로 이루어진다. 따라서, 매번 입력에 대해 전체 가중치를 모두 학습하는 다층신경망에 비해 학습속도가 매우 빠르며 국부적으로 매우 큰 비선형 특성도 학습이 가능하다는 장점이 있다. 물론 국부

적인 가중치를 사용하기 때문에 사용되는 가중치의 총 개수는 다층신경망에 비해 많다. 그래서 CMAC은 가중치를 줄이기 위해 해쉬사상(hash mapping) 이라 불리는 특별한 방법으로 입력과 실제 저장된 가중치 사이에 상호관계를 사용한다. 다음에서는 이러한 사상을 단계적으로 설명한다.

2.1 입력공간의 이동

메모리 활성화시 주소지정을 단순화하기 위해서 입력공간의 이동을 수행한다. 이는 단순히 모든 입력변수가 영보다 큰 범위에 존재하게 하기 위해서 하는 것으로 식(1)처럼 연산을 수행한다.

$$\overline{x}_i = x_i - x_{i, \min}, \quad i = 1, 2, \dots, n \quad (1)$$

여기서 $x_{i, \min}$ 은 x_i 의 최소값을 의미하며 n 은 입력공간의 차원을 나타낸다. 이렇게 하면 모든 변수는 영보다 크고 양의 공간에만 존재하게 된다.

2.2 관련 메모리의 활성화

설명을 간단히 하기 위해 입력변수가 단 2개인 CMAC을 생각하자. 각 입력변수 \overline{x}_1 와 \overline{x}_2 에 해당하는 공간은 블록(block)이라 불리는 영역으로 그림 1의 (가)와 같이 정량화한다. 각 블록의 크기는 같은 변수에서는 일정하며 q_i 로 표시하기로 한다. 따라서 블록의 수는 각각 4개씩이 된다. 그러면 입력변수 공간은 하이퍼큐브(hypercube)라 불리는 여러 개의 영역으로 구분이 된다. 각각의 하이퍼큐브는 컴퓨터상의 메모리의 위치와 일대일 대응한다. 이러한 방법으로 입력공간을 구분한 것을 요소(element)라 하며 또다른 요소를 만들기 위해 (가)를 일정한 방향으로 이동(shift)시켜 (나)와 같은 새로운 요소를 얻는다. 이 때의 이동방향을 $S_2 = [s_{21} \ s_{22}]^T$ 라 하여 표시 할 수 있는데 s

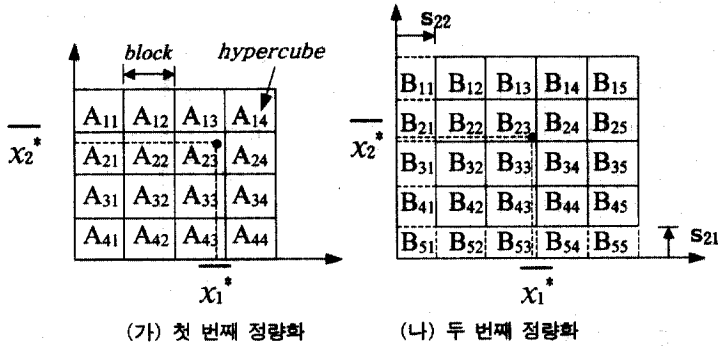


그림 1. 주소활성화를 위한 정량화

의 앞첨자 2는 2번째 요소의 정의를 위한 것이라는 의미이고 뒤첨자는 각각 변수의 방향을 의미한다. 즉, s_{21} 은 두 번째 요소의 정의를 위해서 첫 번째 변수가 이동해야 할 양을 의미한다. 첫 번째 요소에서는 요소의 이동이 없으므로 $S_1 = [0 \ 0]^T$ 이다. 이렇게 요소의 정의를 얼마나 많이 하는냐는 CMAC의 일반화 특성을 결정짓는데 사용된 요소의 총 수를 일반화 폭이라 한다. 이와 같이 정의된 CMAC에서 가령 입력 $\bar{X}^* = [\bar{x}_1^* \ \bar{x}_2^*]^T$ 에 대하여 출력을 어떻게 만드는지를 알아보면, 우선 그림 1의 (가)에서는 A_{23} 이, (나)에서는 B_{23} 이 각각 활성화 된다. 이 때 각 활성화된 하이퍼큐브는 하나의 가중치가 기억된 메모리를 의미하므로 이에 해당하는 가중치를 이용해서 출력을 만든다. 일반적으로 요소의 수(일반화 폭)가 N_g , 입력변수가 n 일 때 k 번째 요소에서 활성화된 가중치의 주소는 식(2)와 같이 표현할 수 있다.

$$a_k = \text{ceil}\left(\frac{\bar{x}_1 - s_{k1}}{q_1}\right) + \sum_{i=2}^n \left[\text{ceil}\left(\frac{\bar{x}_i - s_{ki}}{q_i}\right) \cdot \prod_{j=1}^{i-1} (b_j + 1) \right] + a_{k0} \quad (2)$$

$k = 1, 2, \dots, N_g$

여기서 $\text{ceil}(y)$: y 보다 작지 않은 최대 정수,
 s_{ki} : k 번째 요소의 정의에 쓰이는 i 번

째 변수의 이동량,
 q_i : i 번째 변수의 블록 크기,
 b_j : j 번째 변수공간의 총 블록 수

$$a_{k0} = \begin{cases} a_{k-1,0} + \prod_{i=1}^n (b_i + 1), & (k \geq 2) \\ 1 & (k = 1) \end{cases} \quad (3)$$

2.3 출력계산과 학습

활성화된 메모리의 가중치를 이용한 출력의 계산은 식(4)와 같다.

$$u = \sum_{k=1}^{N_g} W(a_k) \quad (4)$$

여기서 W 는 가중치를 나타내며 사용된 총 가중치의 개수 (N_w)는 식(5)와 같다.

$$N_w = N_g \cdot \prod_{i=1}^n (b_i + 1) \quad (5)$$

주어진 입력에 대하여 적절한 출력을 얻기 위해서는 우선 학습을 통해서 최적의 가중치를 찾는 과정이 필요하다. 이러한 가중치의 학습은 식(6)과 같은 규칙에 의해서 이루어지는데 이것은 오차의 최소화를 위한 최적화기법에 근거한 식이다.

$$W^{new} = W^{old} + \frac{\eta}{N_g} (d - u) \quad (6)$$

여기서 d 는 목표 출력값이고, η 는 학습률을 나타낸다.

2.4 필요한 메모리의 크기

CMAC은 기본적으로 표참조 기법(table look-up)에 근거해서 정보를 저장하기 때문에 필요한 메모리의 크기가 클 수밖에 없다. 따라서 현실적으로 가능한 메모리를 사용하기 위해서 앞에서 설명한 방법으로 사상을 함으로써 메모리의 크기를 줄인다. 만일 입력이 n 개이고 서로 다른 Ω 개의 정보를 저장하기 위해서는 Ω^n 개의 메모리가 필요하게 되지만 앞서 설명한 사상기법을 도입하면 그 크기를 $N_g(\Omega/N_g)^n$ 개로 줄일 수 있게 된다. 그 비를 따져보면 N_g^{1-n} 이 되는데, 예를 들어 입력이 세개 일반화 폭이 50이라면 그 비는 $50^{-2} = 4 \times 10^{-4}$ 가 되기 때문에 CMAC이 현실적으로 사용 가능하게 만드는 사상기법인 것이다.

3. 동적응답 예측

그림 2는 구조물의 동적 응답을 학습하기 위한 개념도를 보여주고 있다. 외부하중과 구조물의 상태가 CMAC의 입력으로 주어지면 구조물의 동적 응답을 출력으로 내보낸다. 이 때 예측된 동적응답은 실제 구조물의 응답과 비교하여 오차가 줄어

들 수 있도록 가중치를 수정해 간다. 이러한 반복적인 가중치의 보정작업을 학습이라 한다. 만일 이러한 학습이 충분히 수행된다면 CMAC은 구조물의 동적응답을 예측하게 되며 구조물의 동적응답 예측(식별)이 가능하게 된다. 학습을 위한 동적응답은 실험을 통하여 구하거나 이것이 불가능할 경우 수치적으로 구할 수 있다.

학습을 수행함에 있어서 그 달성정도를 평가하기 위해서는 우선 오차의 척도를 정의해야 한다. 널리 사용되어 온 것 중의 하나가 바로 SSE (Sum Squared Error)이며 때때로 이 값을 정규화하여 사용하기도 한다. 그러나 이것들은 예측신호의 오차를 누적하기 때문에 예측하는 응답의 개수에 영향을 받기 때문에 이러한 영향을 배제하기 위해서 식(7)과 같은 오차의 척도를 이용하고자 한다. 이것은 예측하고자 하는 응답의 최대값으로 정규화하고 예측응답의 총 개수에 대하여 평균을 취하였기 때문에 Square-root of Averaged Error Norm(SAEN)이라 정의하였고 개개의 응답예측의 정확성을 이 값으로 평가하면 용이하다.

$$SAEN = \sqrt{\frac{1}{N_T} \sum_{i=1}^{N_T} \left(\frac{d_i - u_i}{d_{max}} \right)^2} \quad (7)$$

여기서 N_T 는 예측신호의 총 개수를 의미한다. 학습은 SAEN 값이 미리설정된 값 이하로 작아질 때까지 계속되며 또한 이 값은 미학습 신호의 예측시 성능평가의 척도로서 사용된다.

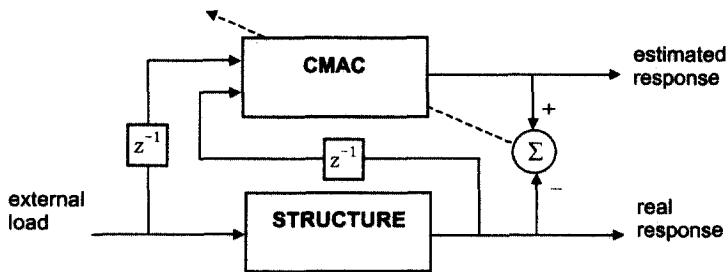
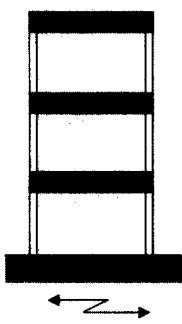


그림 2. 동적응답의 학습개념도

4. 수치해석

4.1 모델 구조물

일반적으로 토목구조물은 다자유도 시스템이지만 지진응답은 소수의 저차 모드 성분으로 대표할 수 있는 경우가 대부분이다. 따라서 본 제안방법의 검증에 위해 3자유도 구조물을 예제로 채택하였다. 물론 고차 모드의 성분이 응답에 많은 영향을 미치는 구조물이라 할지라도 특정자유도의 응답 예측을 위해 그 위치의 응답자료만 얻을 수 있다면 본 방법을 국부적으로 적용할 수 있다.



$$M = \begin{bmatrix} 9.806 & 0.0 & 0.0 \\ 0.0 & 9.806 & 0.0 \\ 0.0 & 0.0 & 9.806 \end{bmatrix} (N \cdot \text{sec}^2/m)$$

$$C = \begin{bmatrix} 3.826 & -0.573 & 0.616 \\ -0.573 & 4.567 & -0.026 \\ 0.616 & -0.026 & 4.372 \end{bmatrix} (N \cdot \text{sec}/m)$$

$$K = \begin{bmatrix} 2.740 & -1.641 & 0.369 \\ -1.641 & 3.021 & -1.624 \\ 0.369 & -1.624 & 1.333 \end{bmatrix} \times 10^4 (N/m)$$

그림 3. 모델구조물

T. T. Soong (1990)이 사용한 그림 3의 3층 전단건물을 본 연구의 해석 모델로 사용하였다. 이 건물은 Soong의 연구에서 능동 텐던에 의한 진동제어에 사용된 실제모델이다. 이 모델의 질량(M), 감쇠(C) 및 강성행렬(K)은 그림 3에 있는 바와 같다. 구조물의 고유진동수는 각각 2.24, 6.83 및 11.53Hz이며 모드별 감쇠는 각각 1.62, 0.39 및 0.36 % 이다.

이 모델이 지진하중 하에 있을 때 최상층의 변위와 속도를 측정하고 이 데이터가 CMAC으로 입력된다. CMAC은 이 데이터를 이용하여 바로 다음 샘플링 단계에서의 변위를 예측하게 된다.

4.2 지반 운동

표 1에 나열한 10개의 중요지진을 이용하여 구조물을 가진시켰다. 이 지진가속도는 다른 연구들에서 주로 사용해 온 역사적 지진 중에서 임의로 추출한 것들이다. 처음의 3개는 CMAC의 학습을 위해 사용하였으며 나머지 7개는 학습된 CMAC의 검증을 위해 사용하였다. 모든 지진기록은 샘플 시간간격이 0.02초이며 20초씩의 지속시간을 갖는다. 따라서 각 지진은 1000개씩의 가속도 데

표 1. 해석에 사용한 지진

기 호	발 생 지 역	발생년도	성 분	PGA (m/s ²)	Scaling (%)	scaling 후의 PGA (m/s ²)
EC	El Centro	1940	S00E	3.417	100	3.417
NO	Northridge	1994	S65W	3.246	100	3.246
KC	Kern County	1952	N21E	1.527	100	1.527
SF	San Fernando	1971	S16E	9.567	30	2.870
SJ	San Jose	1955	N31W	1.002	200	2.004
BM	Borrego Mt.	1968	S00W	1.278	250	3.195
LB	Long Beach	1933	S08W	1.306	100	1.306
LP	Loma Prieta	1989	N90E	3.258	100	3.258
WW	Western Washington	1949	N86E	2.746	100	2.746
ML	Mammoth Lake	1980	S90W	3.040	100	3.040

이터로 구성되었다. 최대지진가속도(peak ground acceleration: PGA)가 너무 작거나 큰 것은 중소규모의 지진이 되도록 그 크기를 조정하였다. 예를 들어, San Fernando(SF) 지진은 원래 PGA가 $9.567(m/s^2)$ 이었지만 전체적으로 30%로 감축시켜 PGA가 $2.87(m/s^2)$ 이 되었다.

4.3 CMAC의 구조 및 학습

CMAC의 입력은 전단계의 변위, 속도 및 지반가속도이며 출력은 다음단계의 변위이다. 각 입력에 대한 정량화 간격은 변위가 0.015(m), 속도가 $0.367(m/s)$ 이며 지반가속도는 $2.33(m/s^2)$ 이다. 그리고 각 입력공간은 6개, 3개 및 3개의 블록으로 구성되었다. 따라서 일반화 폭을 50개로 하면 CMAC이 가지는 총 기억장소의 크기는 $50 \times (6+1) \times (3+1) \times (3+1) = 5600$ 이다. 학습자료로는 3개의 지진 즉 EC, NO 및 KC에 의한 지진응답으로 구성되었으므로 총 3000가지 경우의 입력-출력 짝이 된다. 학습률을 0.9로 하여 SAEN 값이 0.01이하가 될 때까지 계속하였다. Pentium 166MHz 성능의 컴퓨터를 이용한 학습결과를 표 2에 요약하였다. 그림 4는 학습에 따른 SAEN 값의 감소를 보여주고 있다. 표와 그림에서 보는 바와 같이 CMAC의 학습속도는 매우 빠름을 알 수 있다.

표 2. 학습 소요시간

Epochs	CPU Time
4	3.406 sec

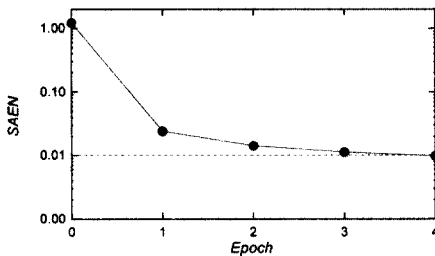


그림 4. 학습에 따른 SAEN

4.4 잡음이 포함되지 않은 경우의 예측

학습한 CMAC의 성능을 분석하기 위하여 학습에 사용하지 않은 나머지 7개의 지진이 가해졌을 때의 응답예측을 수행하였다. 그림 5는 학습한 지진뿐만 아니라 학습하지 않은 지진에서도 뛰어난 응답예측 성능을 가짐을 보여주고 있다. 표 3에 정리한 SAEN 값의 비교에서도 거의 대부분 1.5 이하의 값을 나타내고 있다.

4.5 잡음이 포함된 경우의 예측

측정신호에 잡음이 포함되었을 경우 CMAC이 가지는 예측성능을 알아보기 위하여 임의로 발생한 랜덤 신호를 CMAC의 입력신호에 더하였다. 랜덤 신호의 최대 크기는 입력신호의 최대 값의 3%로 가정하였다.

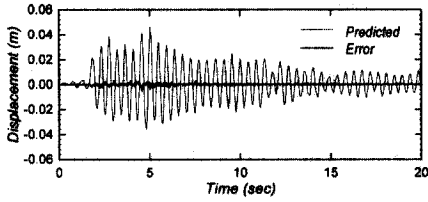
그림 6은 잡음이 포함된 신호를 가지고 예측한 결과를 보여주고 있다. 비록 CMAC의 입력신호에 잡음이 포함됐어도 응답 예측시 심각한 오차는 보이지 않았다. 표 4에 나타난 SAEN 값을 보더라도 대부분 큰 증가를 보이지 않았으며 LB지진의 경우가 가장 크고 나머지는 모두 4.0 이하의 값을 가진다.

표 3. 잡음이 포함되지 않은 경우의 SAEN ($\times 10^{-2}$)

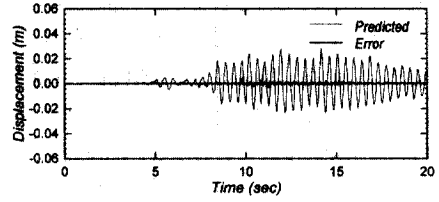
지진	EC	NO	KC	SF	SJ
SAEN	1.26	1.47	1.15	0.79	0.91
지진	BM	LB	LP	WW	ML
SAEN	1.07	1.04	0.99	1.27	1.38

표 4. 잡음이 포함된 경우의 SAEN ($\times 10^{-2}$)

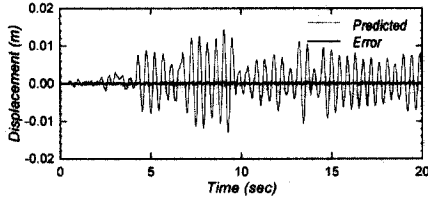
지진	EC	NO	KC	SF	SJ
SAEN	2.24	2.69	3.63	2.46	3.15
지진	BM	LB	LP	WW	ML
SAEN	2.26	4.17	2.25	2.96	3.85



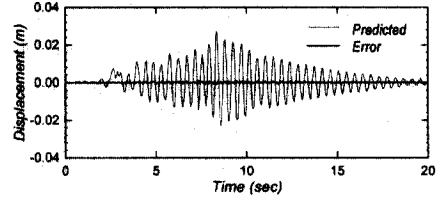
(가) EC



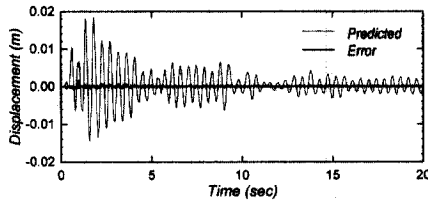
(나) NO



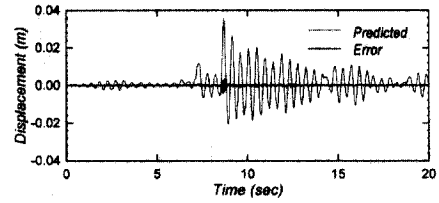
(다) KC



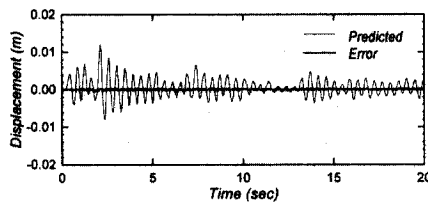
(라) SF



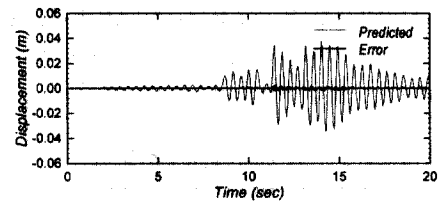
(마) SJ



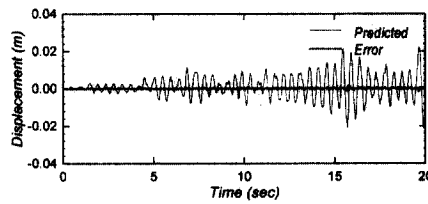
(바) BM



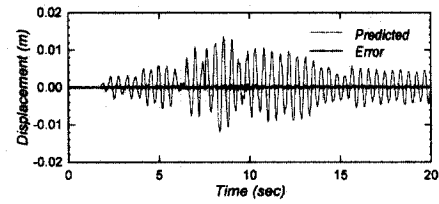
(사) LB



(아) LP

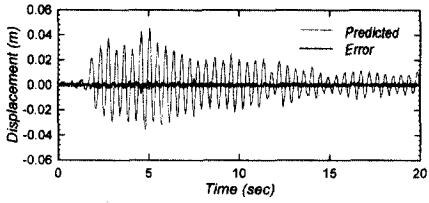


(자) WW

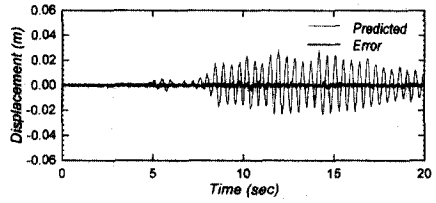


(차) ML

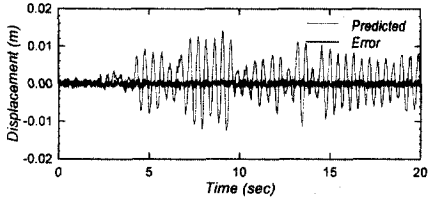
그림 5. 잡음이 포함되지 않은 경우의 응답 예측



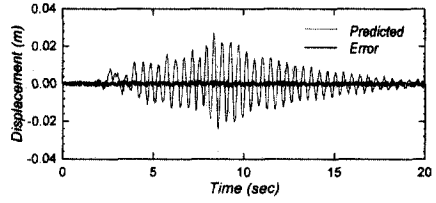
(가) EC



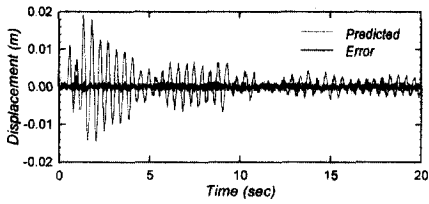
(나) NO



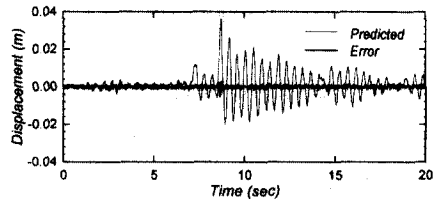
(다) KC



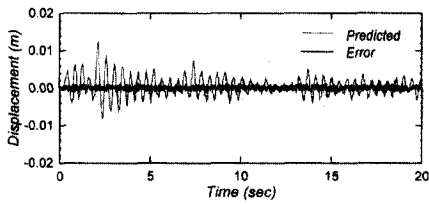
(라) SF



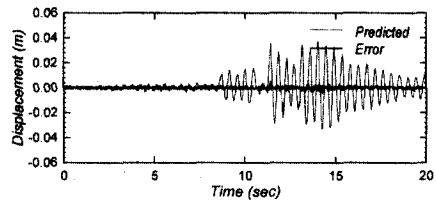
(마) SJ



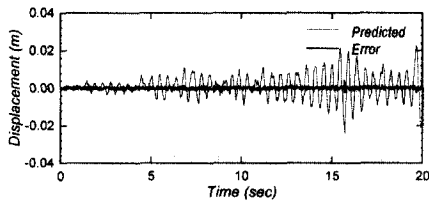
(바) BM



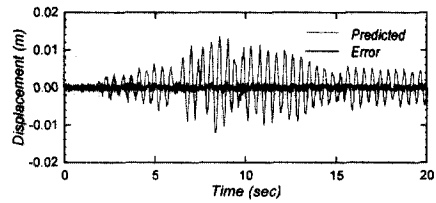
(사) LB



(자) LP



(차) WW



(차) ML

그림 6. 잡음이 포함된 경우의 응답 예측

5. 결론

본 연구에서는 입력공간의 이동, 주소할당 및 학습 등 CMAC의 전반적인 과정을 체계적으로 정리하였다. 그리고 구조물의 동적응답 예측을 위한 CMAC을 구성하여 매우 우수한 성능을 보였다. 수치해석에서는 동적응답의 학습시 그 속도가 매우 빠름을 보였다. 또한 학습한 지진뿐만 아니라 학습하지 않은 지진에 대해서도 그 성능이 크게 나빠지지 않음을 보였으며 잡음을 포함한 신호를 입력했을 때도 예측의 오차가 크게 증가하지 않고 안정적으로 작동함을 보였다. 이처럼 CMAC은 구조공학분야에서도 매우 효과적으로 적용될 수 있다는 결론을 얻었다.

감사의 글

본 연구는 한국과학재단의 특정연구개발사업의 사업비 지원으로 수행된 것으로서 한국과학재단의 연구비 지원에 감사드립니다.

(계약번호: KOSEF(특정) 97-0601-0301-3)

참고 문헌

1. J. S. Albus, A new approach to manipulator controller: the cerebellar model articulation controller (CMAC), *J. Dyn. Sys., Measurement & Control, Trans. ASME*, Vol. 97, No. 3, pp. 220-227, 1975.
2. K. Bani-Hani, and J. Ghaboussi, Nonlinear structural control using neural networks, *J. Engrg. Mech., ASCE*, Vol. 124, No. 3, pp. 319-327, 1998.
3. D. H. Kim, J. S. Jo, and I. W. Lee, Application of CMAC neural network to real time identification of nonlinear structures, *J. Engrg. Mech., ASCE*, 2000. (to be published).
4. A. G. Chassiakos, and S. F. Masri, Modelling unknown structural systems through the use of neural networks, *Earthquake Engrg. and Struct. Dynamics*, Vol. 25, pp. 117-128, 1996.
5. Chassiakos, A. G., Masri, S. F., Smyth, A. W. and Caughey T. K. (1998). "On-line identification of hysteretic systems." *J. Appl. Mech., Trans. ASME*, Vol. 65, March, pp. 194-203.
6. H. M. Chen, , G. Z. Qi, J. C. S. Yang, and F. Amini, Neural network for structural dynamic model identification, *J. Engrg. Mech., ASCE*, Vol. 121, No. 12, pp. 1377-1381, 1995.
7. H. M. Chen, K. H. Tsai, G. Z. Qi, and J. C. S. Yang, Neural network for structural control, *J. Computing in Civ. Engrg., ASCE*, Vol. 9, No. 2, pp. 168-176, 1995.
8. C. T. Chiang, and C. S. Lin, CMAC with general basis functions, *Neural Networks*, Vol. 9, No. 7, pp. 1199-1211, 1996.
9. J. G. Francisco, R. F. Anibal, and A. Antonio, Generalizing CMAC architecture and training, *IEEE Trans. Neural Networks*, Vol. 9, No. 6, 1509-1514, 1998.
10. J. Ghaboussi, and A. Joghataie, Active control of structures using neural networks, *J. Engrg. Mech., ASCE*, Vol. 121, No. 4, pp. 555-567, 1995.
11. H. Hama, C. Xing, and Z. Liu, New high order association memory system based on Newtons forward interpolation. *IEICE Trans. Fundamentals*, E81-A(12), pp. 2688-2693, 1998.

12. Y. He, and J. Wu, Control of structural seismic response by self-recurrent neural network (SRNN), *Earthquake Engrg. and Struct. Dynamics*, Vol. 27, pp. 641-648, 1998.
13. S. L. Hung, and J. C. Jan, MS_CMAC neural network learning model in structural engineering, *J. Computing in Civ. Engrg.*, ASCE, Vol. 13, No. 1, pp. 1-11, 1999.
14. J. T. Kim, H. J. Jung, and I. W. Lee, An optimal control using neural networks, *J. Engrg. Mech.*, ASCE, Vol. 126, No. 2, pp. 201-205, 1999.
15. K. M. Koo, and J. H. Kim, Robust control of robot manipulators with parametric uncertainty, *IEEE Trans. Automatic Control*, Vol. 39, No. 6, pp. 1230-1233, 1994.
16. M. Michael, S. James, and D. I. Cho, Neural network control of automotive fuel-injection systems, *IEEE Control Systems*, June, pp. 31-36, 1994.
17. K. Nikzad, J. Ghaboussi, and S. L. Paul, Actuator dynamics and delay compensation using neurocontrollers, *J. Engrg. Mech.*, ASCE, Vol. 122, No. 10, pp. 966-975, 1996. [18] T. T. Soong, *Active structural control: theory and practice*. John Wiley & Sons, New York, N. Y., 1990.
18. H. L. Stephen, A. H. David, and J. G. Jack, Theory and development of higher-order CMAC neural networks, *IEEE Control Systems*, April, pp. 23-30, 1992.

(접수일자 : 2000년 2월 18일)