

효율적인 신경회로망 학습을 이용한 \bar{x} 관리도의
이상패턴 인식에 관한 연구
- \bar{x} Control Chart Pattern Identification
Through Efficient Neural Network Training -

김기영*
Kim, Ki Young
유정현*
Yoo, Jeong Hyun
윤덕균**
Yun, Deok Kyun

Abstract

Control Chart is a powerful tool to detect that process is in control or out of control. CIM can have real effect when CIM involve automated quality control. A neural network approach is used for unnatural pattern detecting of control chart. The previous moving window method uses all unnatural pattern that is detected as moving time window. Therefore, It trains a large number of unnatural pattern and takes training time long. In this paper, the proposed method tests a small number of training unnatural pattern which modifies test data without repeating time. We shows that the proposed method has differences in training time and identification rate on the previous moving windows method. As results, we reduced training time and obtain the same identification rate.

1. 서론

컴퓨터와 정보 공학의 발전으로 인하여 요즘의 생산관리 형태는 컴퓨터 통합생산(Computer Integrated Manufacturing)시스템이 되는 추세이다. CIM 시스템은 생산 부문에서 영업부문까지의 모든 정보를 컴퓨터 네트워크로 연결하여 제품의 설계, 제조 및 생산관리에 관한 데이터베이스를 공유함으로써 수주에서 출하, 고객관리에 이르기까지 종합적으로 물자와 정보의 흐름을 자동화하여 경영의 효율화를 이루기 위한 유연생산시스템이다[2].

CIM에서 품질관리는 피드백(Feed-Back)과 관리의 기능을 갖는다. 품질에 대한 정보를 적시에 제공하고 이 정보를 분석하여 설계, 제조 및 공정관리와 계획, 영업 등 경영진의 의사결정에 기초를 제공한다. 품질 정보의 즉각적인 제공, 공정의 실시간 관리등 품질 관리가 자동화되어 CIM의 일부분으로 통합되어야만 CIM의 진정한 효과를 얻을 수 있다[15].

SPC(Statistical Process Control)에서의 주된 도구는 관리도이다. 관리도는 자연스러운 패턴과 부자연스러운 패턴의 2가지 범주로 나뉘어 질 수 있는데 이러한 패턴들의 형태는 Random과 Nonrandom으로 각각 짝지어 질 수 있다[11]. 이것은 곧 가피원인과 불가피원인으로 규정될 수 있는데, 관리도상의 패턴을 통해 가피원인이 있다는 것을 확인하기 위한 테스트와 Run에 대한 규칙들이 많은 실무자들에 의해서 제안되어왔다. 일반적으로 이러한 규칙들은 공정에서의

*한양대학교 산업공학과 박사과정

**한양대학교 산업공학과 교수

문제점들을 감지하기 위해서 생산현장에서 많이 사용되어왔다. 하지만, 이러한 규칙들이 항상 명쾌하게 관리도상에서 공정이 보여주는 특정한 형태의 패턴을 감지할 수는 없었다. 그래서 여러 가지 이상패턴에 대한 감지를 위해서 신경망(Neural Network)을 이용한 패턴인식 방법과 전문가 시스템을 이용한 방법이 적용되어졌다.

관리도는 가피원인(Assignable Causes)을 일으키는 문제들을 바로 잡기 위해서 작업자에게 공정 특성치들(Process Characteristics)의 시각적인 표현을 전해준다. 예를 들어, 샘플링 검사를 한 하나의 관측치가 관리한계를 벗어나거나, 관리한계내에서 관측치가 주기적인 변동을 나타낸다든지 하는 것을 가피원인에 대한 표시가 될 것이며 공정에 대한 올바른 조치를 취하는 것에 대한 유용한 정보를 제공할 것이다[7]. 관측치들의 변동은 자연스러운 패턴과 부자연스러운 패턴의 2가지 범주로 나뉘어 질 수 있는데 이러한 패턴들의 형태는 Random과 Nonrandom으로 각각 짝지어 질 수 있다[11]. 이것은 곧 가피원인과 불가피원인으로 규정될 수 있는데, 관리도상의 패턴을 통해 가피원인이 있다는 것을 확인하기 위한 테스트와 Run에 대한 규칙들이 많은 실무자들에 의해서 제안되어왔다. 일반적으로 이러한 규칙들은 공정에서의 문제점들을 감지하기 위해서 생산현장에서 많이 사용되어왔다. 하지만, 이러한 규칙들이 항상 명쾌하게 관리도상에서 공정이 보여주는 특정한 형태의 패턴을 감지할 수는 없었다. 그래서 여러 가지 이상패턴에 대한 감지를 위해서 신경망(Neural Network)을 이용한 패턴인식 방법과 전문가 시스템을 이용한 방법이 적용되어졌다.

이것은 CIM환경에서 관리도상에 표시되는 공정상의 이상형태의 패턴에 대한 정보를 전달해주는 것을 가능하게 만들 수 있다. 이것은 CIM환경에서 발생하는 무수한 품질 데이터를 효율적으로 관리할 수 있도록 만들 것이며, 품질향상에 필요한 정확한 정보를 적시에 제공받을 수 있게 만들 것이다[2].

본 논문에서의 목적은 이러한 이상패턴에 대해서 자동적으로 인식을 할 수 있는 패턴인식기를 개발하여서 이상패턴에 대한 인식을 사람의 손을 거치지 않고 인식을 할 수 있는 효율적인 인식기를 개발하고자 하는 것이다. 본 논문에서 인식을 하고자 하는 이상패턴의 종류는 아래의 <그림1>과 같고 그 이상패턴은 옆에 있는 분포를 내재하고 있다. 이러한 이상패턴은 관리도상에서는 관리한계내에 있지만 공정상의 이상을 포함하고 있는 것이고 불가피 원인 때문에 생기는 것이 아니라 가피원인(Assignable Causes)을 포함하고 있는 것이다[6].

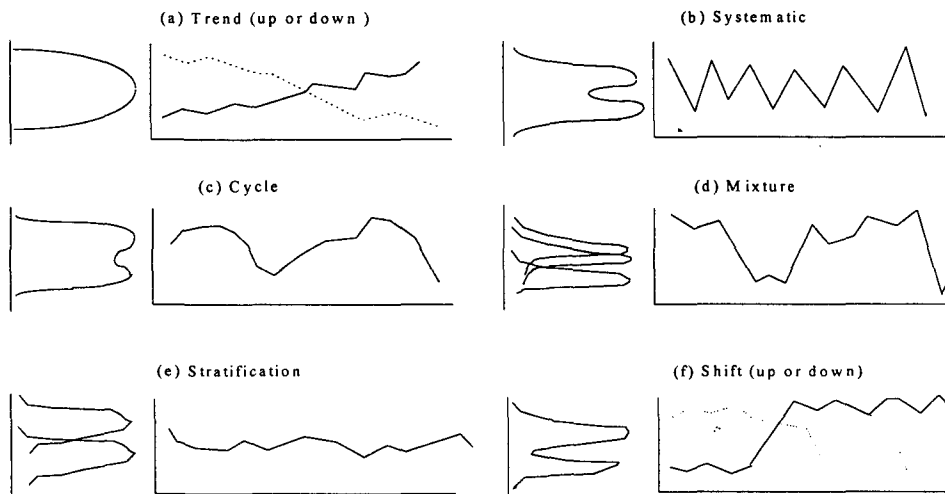
본 논문에서는 이상 패턴에 대해서 학습을 시킬 경우 Window창을 움직여서 모든 경우에 대해서 학습을 시키지 않고 기본적인 패턴들에 대해서 학습을 시켜서, 총 패턴의 수를 106개로 줄였다. 학습을 시킨 후 Test data에 대해서 Window창을 움직여서 학습시킨 패턴과 일치하는지를 검사하도록 하였다. 본 논문의 특성은 관리도 이상 패턴 인식이 실제 생산현장에서 적용될 때 학습에 필요한 계산량을 크게 줄여서 공정변수의 변화가 있을 때 빠르게 재학습을 시킬 수 있도록 하였다.

2. 관리도의 이상패턴 인식을 위한 신경망의 적용

2.1 관리도에서의 이상요인과 Pattern의 종류

관리도에서의 이상요인은 기본적으로 3σ 를 벗어나는 이상점에 대해서 공정에 대한 이상유무를 판단할 수 있다. 하지만 관리도상의 점들이 모두 관리한계내에 들어 있다고 해도 공정이 관리상태에 있는 것은 아니다. 왜냐하면 제2종 오류를 일으키고 있는 점들이 있을지도 모르기 때문이다. 상이한 모집단에 속해 있는 점이 있으면 그것은 다분히 점의 배열이 랜덤하게 되어야만 한다. 만약 관리도 상의 점들에서 랜덤성이 결여되어 있다면 그것은 관리상태가 아니라 관리 이탈 상태(Out-of-control)를 일으키고 있다고 보아야 한다.

특별한 원인에 대한 관리도상에서의 테스트에 대한 연구는 Western Electric과 Deming, Nelson등에 의해서 이루어졌다[12][13][14]. 이러한 테스트방법에 따르면 지나치게 중심선에 모여있거나, 변동이 주기적으로 일어나는 경우, 관리하한이나 관리상한에 모여있는 경우, 급격스러운 변동, 지속적인 상승등이 그것이다. 이러한 이상패턴들에 관한 연구는 지속적으로 계속되고 있으며 생산현장에서 실질적으로 나타난 현상에 대해서 지속적인 이상패턴에 대한 보고가 있을 것으로 기대된다.



<그림 1> 이상패턴과 내재되어 있는 분포

관리도에서 이상 패턴을 인식하기 위하여 많은 연구가 수행되어 왔다. 이 방법들은 크게 규칙에 의한 인식과 신경망(Neural Network)에 의한 인식의 두 가지로 나뉘어질 수 있다. 과거의 연구는 대부분 규칙을 이용한 인식 방법으로서, 대표적인 것으로는 Western Electric Company의 규칙과 Nelson의 규칙이 있다[12][13].

신경망에 의한 관리도의 패턴 인식에 대한 연구가 Hwang and Hubele [7][8][9]에 의하여 연구되었다. 이들은 패턴 인식에서 가장 많이 사용되는 Back Propagation방법[1][5]을 사용하였고, 학습을 위한 각각의 패턴을 패턴 생성기에 의하여 발생시켰고, 수렴 속도가 빠른 Momentum 방법[5]에 의하여 학습을 시켰다. 관리도의 구역을 m개의 구역(zone)으로 나누고 측정된 수치에 대해서 정규화를 한 후 m개의 구역 중 정규화된 수치에 해당하는 구역에 1을 입력하고 나머지 구역에는 0을 입력하여서 하나의 Data Set을 형성하도록 하였다. 이들은 실험 결과 은닉층(Hidden Layer)의 node 수는 30, 구역수는 7이 가장 적합하다는 결론을 내렸다 [9].

2.2 Pattern 인식에 대한 신경망의 적용

신경망은 원래부터 병렬 처리의 장점을 가지고 있으며, 패턴 인식 과정에서 노이즈(Noise)나 형태의 변형에 강한 장점을 가지고 있다. 또한 패턴 인식에 있어서는 명확하게 구별되는 정보 처리보다는 다소 부정확한 정보를 처리해야 하는 경우가 훨씬 더 많다. 이런 상황을 고려해 본다면 신경망을 통한 패턴 인식에의 접근은 매우 훌륭한 방법이라고 할 수 있다[1].

신경망의 구조는 특정한 학습에 매우 우수한 성능을 나타내며, 대부분의 신경망 모델들이 패턴 인식에 매우 훌륭한 결과를 보여 주고 있다. 패턴 인식은 신경망의 성능을 극대화할 수 있

는 분야로 볼 수 있다. 네트워크들은 이전에 학습된 것들의 형상을 찾아내거나 주어진 입력에 대하여 해당되는 패턴을 선택하거나 생성시킨다. 이것은 정적인 패턴들에게만 국한된 것이 아니고 시간과 공간에 따라 변하는 복잡하고 동적인 패턴들에게도 적용될 수 있다[1].

2.3 Momentum 방법을 이용한 Back Propagation Model

다층 퍼셉트론은 입력층과 출력층 사이에 하나 이상의 중간층이 존재하는 신경망이다. 이 때 입력층과 출력층 사이의 중간층을 은닉층(Hidden layer)이라 부른다. 네트워크는 입력층, 은닉층, 출력층 방향으로 연결되어 있으며, 각 층내의 연결과 출력층에서 입력층으로의 직접적인 연결은 존재하지 않는 전방향(Feedforward)의 네트워크이다[1].

이런 다층 퍼셉트론 모델에서 시그모이드(Sigmoid) 비선형 활성화함수를 적용할 경우 결정 영역이 통상 직선이 아닌 완만한 곡선으로 경계를 이루기 때문에 행위의 분석이 약간 복잡하지만 미분이 가능하므로 오히려 은닉층을 학습할 수 있는 Back Propagation 학습 알고리즘을 수행할 수 있다. 대부분의 다층 퍼셉트론은 Back Propagation 학습 알고리즘을 사용하여 학습시킬 수 있다.

본 논문에서는 수렴속도가 빠른 Momentum 수정방법[10]을 사용하고자 하는데, 이전의 수정량도 고려하여 수정량을 결정하는 방법이다. 이 방법은 엄밀히 말하자면 최급하강법은 아니지만 오차로부터 구해진 수정량에 의한 오버 시프트(Over Shift)를 억제하여 학습의 고속화에 유효한 것으로 알려져 있어 일반적으로 많이 사용되고 있다.

Momentum 방법은 신경망이 여러가지 패턴들에 대해서 학습의 수정이 일반적으로 같은 방향일 때는 상당히 큰 연결강도의 변화를 가져오고, 상대적으로 적은 학습계수를 선택함으로써 어떤 하나의 패턴에 대해서 큰 에러가 발생하는 것을 막는다. 이러한 것들은 신경망이 지역최소값(Local Minimum)에 빠지는 것을 막는데 큰 도움을 준다. Momentum 방법은 신경망이 경사(Gradient)의 방향이 아니라 현재의 경사(Gradient)와 전의 연결강도를 조정한 방향의 결합된 방향으로 나아가게 한다.

3. 제안된 신경망 학습방법

3.1 이상패턴의 생성

이상패턴의 발생을 하기 위해서 (식1)과 같은 이상 패턴 발생기를 사용하여서 이상 패턴을 발생시킨다.

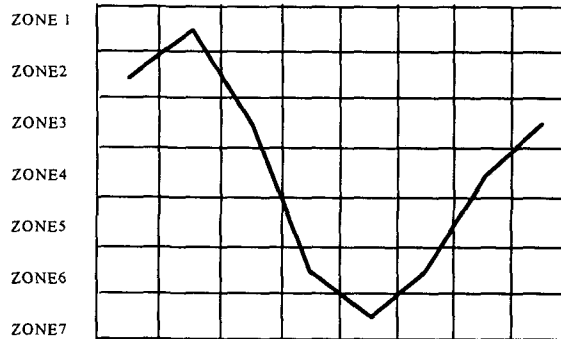
$$x(t) = \mu + \gamma(t) + d(t) \quad (1)$$

$x(t)$ 는 시간의 변화에 따른 관측값을 나타내고, μ 는 공정의 평균을 나타내며, $\gamma(t)$ 는 불가피 원인 때문에 생기는 Random Normal Noise를 나타내며 본 논문에서는 일양분포를 convolution하여서 구하였다[3]. $d(t)$ 는 가파원인에 기인한 특별한 분포를 나타내게 된다. 예를 들어, 아래와 같은 식을 가지는 발생기는 주기적인 변동에 대해서 발생시키는 발생기가 된다.

$$d(t) = \sin[2\pi(t-t_0)/\Omega]k\sigma \quad (2)$$

여기에서 σ 는 프로세스가 안정상태일때의 공정의 표준편차이고, k 는 주기의 진폭을 결정하는 상수이다. 그리고 Ω 는 주기적 변동의 주기를 나타내며 t_0 는 참고시점이 된다. 관리도 패턴들에 대한 패턴의 파라미터값의 선택은 학습에 있어서 중요하다.

이 패턴발생기는 발생해야 하는 이상패턴의 특정한 모양을 결정하기 위한 여러가지의 모수값들을 가지고 6가지의 이상패턴을 발생한다. 이러한 모수들의 모든 가능한 값을 가지고 있는 집합은 H.Brian Hwang 과 Norma Faris Hubele의 논문에서 다룬 모수 값들의 집합을 그대로 사용하였다[8].



<그림 2> 정규화된 Data Set

<그림 2>에서 보는 것과 같이 관리도의 구역을 7개의 구역으로 나누고 패턴 발생기로 부터 발생된 데이터 수치를 정규화의 과정을 거쳐 해당되는 구역에 입력했다. 그리고 값이 들어 있는 구역은 1로 표시를 하고 나머지 구역에 대해서는 0으로 표시를 하였다. 그리고 Type I error와 Type II error를 줄이기 위해서 Window 창의 크기를 8개로 하였다. 그러므로 하나의 데이터 집합은 총 56개의 이진코드로 부호화 되어서 네트워크에 입력이 되게된다.

3.2 수정된 학습 패턴의 선택

기존의 연구에서는 반복적인 이상 패턴 즉 경향, 주기적 변동, 증화, 시스템적인 변동, 혼합과 반복적이지 않은 패턴 즉 수준급변동에 대해서 반복적인 이상 패턴에 대해서는 Moving Windows를 이용하여서 시간의 흐름에 따른 모든 이상 패턴에 대해서 학습을 시키고자 하였고 비반복적인 패턴에 대해서는 한 가지의 패턴에 대해서 학습을 시켰다. 그리고 이상 패턴마다 각각의 모수값 수치들이 있어서 발생해야 하는 패턴의 수가 증가하였다. 이런 경우 공정변수의 값이 변한다면 Back Propagation 모델은 추가적인 재학습이 불가능하기 때문에 다시 모든 이상 패턴을 발생시켜서 다시 학습을 시켜야 했다.

본 논문에서 제안하는 학습 패턴의 선택은 각각의 패턴에 대해서 Moving Window방법을 사용하지 않고 Noise만이 첨가된 패턴에 대해서 학습을 시킨다.

3.3 Test Data에 대한 Modification

Moving Window에 의해서 학습된 신경망은 주기적인 변동에서의 경우 각각의 다른 형태의 주기적 변동에 대해서 같은 반응을 보이도록 학습을 시켰기 때문에 학습의 패턴이 많았고 신경망 학습에서 수렴하지 못하는 경우가 있어서 기존의 논문에서는 Data Refinement과정을 통해서 이미 학습된 신경망에 다시 패턴들을 테스트하여서 수렴하지 못하는 Pattern에 대해서는 삭제할 함으로서 학습시키고자 하는 Pattern의 수를 줄였다. 하지만 여전히 재학습시킬 경우, 많은 Pattern 들을 학습을 시켜야 한다.

본 논문에서는 Test Data에 대해서 Moving Window 변형을 통해서 올바른 Output을 얻어 내고자 하였다. 신경망 학습을 위해서 전처리 과정을 거쳐서 정해진 구역에 1의 형태로 <그림 3>표시되어 있다. 이제 Test Data에 대해서 Moving Window방법을 적용하여서 8개의 새로운 Data Set을 <그림 4>에서 만들어 낸다.

테스트 데이터로부터 발생된 테스트 데이터에 대해서 판별을 하고 수렴을 하지 않는 데이터에 대해서 위와 같은 방법으로 테스트 데이터에 대해서 변형을 가하여 테스트를 실시한다. 그리고 그 기록을 파일에 기록하여서 분석을 하는 과정을 거친다.

ZONE 1	0	1	0	0	0	0	0	0
ZONE 2	1	0	0	0	0	0	0	0
ZONE 3	0	0	1	0	0	0	0	1
ZONE 4	0	0	0	0	0	0	1	0
ZONE 5	0	0	0	0	0	0	0	0
ZONE 6	0	0	0	1	0	1	0	0
ZONE 7	0	0	0	0	1	0	0	0
	t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8

<그림 3> 이진코드로 표현된 테스트 데이터

t=2	t=3	t=4	t=5	t=6	t=7	t=8	t=1
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1
0	1	0	0	0	0	1	0
0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0
0	0	0	1	0	0	0	0

t=6	t=7	t=8	t=1	t=2	t=3	t=4	t=5
0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0
0	0	1	0	0	1	0	0
0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0
0	0	0	0	0	0	0	1

t=3	t=4	t=5	t=6	t=7	t=8	t=1	t=2
0	0	0	0	0	0	0	1
0	0	0	0	0	0	1	0
1	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0
0	1	0	1	0	0	0	0
0	0	1	0	0	0	0	0

t=7	t=8	t=1	t=2	t=3	t=4	t=5	t=6
0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0
0	1	0	0	1	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	1
0	0	0	0	0	0	1	0

t=4	t=5	t=6	t=7	t=8	t=1	t=2	t=3
0	0	0	0	0	0	1	0
0	0	0	0	0	1	0	0
0	0	0	0	1	0	0	1
0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0

t=8	t=1	t=2	t=3	t=4	t=5	t=6	t=7
0	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0
1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0
0	0	0	0	1	0	1	0
0	0	1	0	0	1	0	0

t=5	t=6	t=7	t=8	t=1	t=2	t=3	t=4
0	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0
0	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	1
1	0	0	0	0	0	0	0

t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8
0	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1
0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0
0	0	0	0	1	0	0	0

<그림 4> 변형된 테스트 데이터

4. 수행도의 분석 및 평가

본 연구의 목적은 기존의 인식률을 유지하면서 학습패턴 발생시간과 학습시간을 줄이는 것이다. 그러므로 일단 인식률이 기존의 연구의 인식률에 도달하는지 여부를 알아봐야 한다. 수행도의 평가는 ROT(Rate of Target)[8] 즉, 주어진 패턴을 얼마만큼 제대로 인식을 했느냐와 RONT(Rate of Non Target)[8] 즉, 주어진 패턴을 다른 패턴으로 인식했느냐의 2가지로 평가하기로 하였다. ROT와 RONT의 합이 1이 되지 않는 이유는 신경망이 수렴을 하지 못하고 판정을 내리는데 실패했기 때문이다. 이 Test에서는 0.95의 임계값을 사용하였다.

테스트의 방법들은 패턴생성기에서 생성된 모의 데이터에 의해서 이루어진다. 반복적인 이상패턴에 해당하는 독립적인 패턴 100개가 생성이 된다. 그리고 비반복적인 패턴에 대해서는 1000개의 데이터를 발생시켜서 기존의 논문에서 사용한 방법에 대하여 통계적 수치들을 취합하고, 본 논문에서 제안한 방법에 대하여 취합했다.

<표 1> 주기 8인 순환에서의 수행도 평가

모수		기존의 연구		제안된 연구	
진폭	잡음	ROT	RONT	ROT	RONT
1.50	0.10	1.00	0.00	1.00	0.00
2.00	0.10	1.00	0.00	1.00	0.00
2.50	0.10	1.00	0.00	1.00	0.00
1.50	0.30	1.00	0.00	1.00	0.00
2.00	0.30	1.00	0.00	1.00	0.00
2.50	0.30	1.00	0.00	1.00	0.00
1.50	0.50	0.99	0.00	1.00	0.00
2.00	0.50	1.00	0.00	0.99	0.00
2.50	0.50	1.00	0.00	1.00	0.00

<표 2> 시스템적 변동에서의 수행도 평가

모수		기존의 연구		제안된 연구	
진폭	잡음	ROT	RONT	ROT	RONT
1.50	0.10	1.00	0.00	0.99	0.00
2.00	0.10	1.00	0.00	0.99	0.00
2.50	0.10	1.00	0.00	0.98	0.00
1.50	0.30	0.98	0.00	0.95	0.00
2.00	0.30	0.87	0.00	0.86	0.00
2.50	0.30	0.67	0.00	0.63	0.00
1.50	0.50	0.95	0.00	0.93	0.00
2.00	0.50	0.65	0.00	0.61	0.00
2.50	0.50	0.36	0.00	0.33	0.00

<표 3> 총화에 대한 수행도 평가

모수		기존의 연구		제안된 연구	
치우친 값	잡음	ROT	RONT	ROT	RONT
-0.20	0.10	1.00	0.00	1.00	0.00
0.00	0.10	1.00	0.00	1.00	0.00
0.20	0.10	1.00	0.00	1.00	0.00
-0.20	0.20	1.00	0.00	1.00	0.00
0.00	0.20	1.00	0.00	1.00	0.00
0.20	0.20	1.00	0.00	1.00	0.00
-0.20	0.30	1.00	0.00	1.00	0.00
0.00	0.30	1.00	0.00	1.00	0.00
0.20	0.30	1.00	0.00	1.00	0.00

<표 4> 혼합에 대한 수행도 평가

모수		기존의 연구		제안된 연구	
치우친 값	잡음	ROT	RONT	ROT	RONT
1.75	0.30	1.00	0.00	1.00	0.00
2.00	0.30	1.00	0.00	1.00	0.00
2.25	0.30	0.98	0.02	0.98	0.01
1.75	0.50	0.99	0.01	0.99	0.01
2.00	0.50	0.98	0.02	0.98	0.02
2.25	0.50	0.99	0.01	0.98	0.01
1.75	0.70	0.99	0.01	0.99	0.01
2.00	0.70	0.99	0.01	0.98	0.01
2.25	0.70	0.99	0.01	0.99	0.01

<표 5> 상승에 대한 수행도 평가

모수		기존의 연구		제안된 연구	
기울기	잡음	ROT	RONT	ROT	RONT
0.15	0.1	1.00	0.00	1.00	0.00
0.15	0.3	0.85	0.00	0.84	0.00
0.15	0.5	0.56	0.02	0.49	0.00
0.25	0.1	1.00	0.00	1.00	0.00
0.25	0.3	0.93	0.00	0.93	0.00
0.25	0.5	0.54	0.04	0.48	0.00
0.35	0.1	0.69	0.00	0.61	0.01
0.35	0.3	0.43	0.01	0.40	0.00
0.35	0.5	0.32	0.03	0.29	0.01

<표 6> 수준 급변에 대한 수행도 평가

모수		기존의 연구		제안된 연구	
변화폭	잡음	ROT	RONT	ROT	RONT
0.15	0.3	0.4666	0.005	0.461	0.001
2.00	0.3	0.771	0.001	0.751	0.000
2.50	0.3	0.921	0.000	0.910	0.000
1.50	0.5	0.582	0.020	0.533	0.014
2.00	0.5	0.746	0.012	0.734	0.009
2.50	0.5	0.242	0.010	0.211	0.004
1.50	0.7	0.119	0.024	0.132	0.032
2.00	0.7	0.260	0.026	0.239	0.021
2.50	0.7	0.390	0.015	0.383	0.011

결과에서 보듯이 기존의 연구와 제안된 방법을 사용했을 때의 인식률에는 차이가 거의 없음을 알 수 있다. 이제 같은 학습패턴의 생성과 신경망 학습에 걸리는 시간을 분석해 보자. 시간의 측정은 신경망 학습 프로그램의 시작과 끝에 시간을 측정할 수 있는 함수를 삽입함으로써 얻었다. 본 논문에서는 계산량을 측정하는 것이 아니라 프로그램의 전체적인 실행시간을 측정하고 하였다.

실행시킨 결과는 기존의 연구의 방법을 사용한 결과, 10번의 실행을 거쳐서 평균 250.68초의 결과를 얻을 수 있었고 본 논문에서 제안된 방법을 사용한 결과, 10회 실행하여 평균 56.80초의 결과를 얻을 수 있었다. 각 시행마다 약간의 학습시간의 차이가 있었는데 이것은 초기 연결 강도의 설정에 기인한 것이다.

위의 결과와 같이 제안된 방법을 사용하면 거의 같은 인식률을 가지면서도 학습시간을 줄일 수 있다는 결과를 볼 수 있다.

5. 결론

관리도의 이상패턴에 대한 신경망을 적용하여서 이상패턴을 인식하려는 시도가 최근에 있어 왔다. 그리고, 그 연구를 통하여 기존의 규칙으로는 감지하기 어려웠던 이상패턴에 대해서 상당한 수준까지 사람의 손을 거치지 않고 인식을 할 수 있었다. 하지만 기존의 모델에서는 공정 변수의 변화가 있을 경우, Back Propagation Model의 특성상 추가적인 학습이 불가능하여서 전면적인 재학습이 필요로 하였다.

본 논문에서는 재학습이 필요할 경우에 대비하여서 학습을 시킬 패턴의 수를 줄이고 Test Data에 대한 변환을 통해서 학습패턴의 수를 줄인 것을 보상하고자 하였다. 그래서, 각각의 패턴에 대해서 Simulation을 한 결과 인식률에 있어서는 기존의 연구와 거의 같은 결과를 얻을 수 있었고, 패턴생성과 학습에 있어서는 시간을 많이 줄일 수 있었다. 이것은 변화하는 공정에 대해서 그만큼 신속하게 신경망이 변화할 수 있는 환경을 마련한 것이라고 결론지을 수 있다.

참고문헌

- [1] 김대수, 신경망 이론과 응용, 하이테크 정보, 1994
- [2] 김성인, 조남길, 한정희, 품질관리 자동화 - 공정의 이상 패턴 인식을 위한 법칙, 산업공학, 제8권, 제3호, 1993, pp.249-257.
- [3] 지영준외 2인 공저, C로 구현한 수치해석, 높이 깊이, 1995.
- [4] Adam Blum, *Neural Networks In C++*, Wiley.
- [5] Fausett L., *Fundamentals of Neural Networks*, Prentice Hall International, INC.
- [6] H. Brian Hwang, "A Study of Learning of Three-layer Backpropagation Pattern Recognizers for Statistical Control Charts", *International Joint Conference on Neural Networks*, Seattle, Washington, 1991, July, 8-12, II, A-905.
- [7] H. Brian Hwang and Norma Faris Hubele, "X-bar Chart Pattern Recognition, Using Neural Nets", *ASQC Quality Congress Transactions*, Milwaukee, WI, 1991, pp 884-889.
- [8] H. Brian Hwang and Norma Faris Hubele, "Back-propagation pattern recognizers for \bar{x} -control charts : Methodology and Performance", *Computers and Industrial Engineering*, Vol. 24, No. 2, 1993, pp219-235.
- [9] H. Brian Hwang and Norma Faris Hubele, " \bar{x} -control chart pattern identification through efficient off-line neural network training", *IIE Transactions*. Vol. 25, No. 3, 1993, pp. 27-40.
- [10] Jacobs, R. A., "Increased Rates of Convergence Through Learning Rates Adaptation", *Neural Networks*, Vol 1, No 3, 1988, pp295-307.
- [11] Montgomery D. C., *Statistical Quality Control*, 2 nd edit.
- [12] Nelson, L. S., "The Shewhart control chart-tests for special causes", *Journal of Quality Technology*, Vol. 16, No. 4, 1984, pp. 237-239.
- [13] Nelson, L. S., "Interpreting Shewhart \bar{x} -Control Charts", *Journal of Quality Technology*, Vol 17, 1985, pp114-116.

374 김 기 영 · 유 정 현 · 윤 덕 균 효율적인 신경회로망 학습을 이용한 \bar{X} 관리도의 이상패턴 인식에 관한 연구

- [14] Nelson, L. S., "Standardization of Shewart control chart", *Journal of Quality Technonolgy*, Vol 21, 1989, pp287-288.
- [15] Ranky, P. G. "Real time quality control loops in CIM environment", a *Proceedings of 8th International Conference on Automated Inspection and product Control*, IFS Publication, 1987, pp45-62.