

단 신

## 공정자료만을 이용한 모델링 및 최적화에서 Data Reconciliation과 Gross Error Detection

엄태일 · 김인원\*<sup>†</sup> · 박선원\*\*

선경건설(주)

\*건국대학교 화학공학과

\*\*한국과학기술원 화학공학과

(1994년 9월 22일 접수, 1995년 6월 13일 채택)

## Data Reconciliation and Gross Error Detection for Data Intensive Modeling and Optimization

Tae-Il Eom, In-Won Kim\*<sup>†</sup> and Sun Won Park\*\*

*Sunkyung Engineering and Construction Ltd.*

*\*Department of Chemical Engineering, Kon Kuk University*

*\*\*Department of Chemical Engineering, KAIST*

(Received 22 September 1994; accepted 13 June 1995)

### 요 약

공장에서 측정되는 데이터들은 random error 및 gross error와 같은 측정오차를 포함하고 있으며, 이 데이터들은 공정을 설명하는 물질수지식이나 에너지수지식을 만족시키지 못한다. 이와 같은 데이터를 이용하여 공정 해석 및 최적화를 하기 위하여는 측정치들을 조정하고(data reconciliation), gross error를 제거해야 하는 문제(gross error detection)가 선행되어야 한다. 이러한 자료 처리 방법으로는 기존에는 수학적, 통계학적 방법들을 많이 이용하고 있으나, 최근에는 인공신경망을 이용하기도 한다. 본 연구에서는 autoassociative neural network(AAN)과 AAN에 건실성을 부여한 Robust AAN을 이용하여 gross error detection과 data reconciliation(GED/DR)을 동시에 수행할 수 있었다. AAN과 RAAN을 CSTR에 적용하여 비교하고, GED/DR에의 적용 가능성을 검증하였다. AAN을 이용한 GED/DR보다 RAAN을 이용하였을 경우의 GED/DR이 더욱 우수한 결과를 보였다. 공정에 대한 수학적 모델이 없이 공정자료만으로 모델링과 최적화를 행하는 data intensive 모델링과 최적화기법에서 위의 방법으로 GED/DR이 가능함을 알 수 있었다.

**Abstract**—Measured process data are usually containing random errors and gross errors. These measured data do not satisfy process constraints such as the mass and energy balances that describe a process. For the use of these error-contained data in process analysis and optimization, the preprocessing steps such as gross error identification and elimination, and data reconciliation(data rectification) are prerequisite. The existing methods are based on mathematical and statistical techniques, but recently neural networks were investigated for data rectification. In this study, autoassociative neural networks(AAN) and robust ANN(RAAN) were applied for the data rectification of process data of CSTR. The performance of RAAN proved to be superior to that of AAN in the data rectification. We conclude that the use of AAN and RAAN appears to be a promising tool for data rectification.

## 1. 서 론

센서 기술의 발달과 컴퓨터의 가격 대 성능 비의 향상으로 인하여 많은 공정에 자료획득시스템(data acquisition system)과 제어시스템이 도입됨에 따라 공정에서 발생하는 많은 데이터를 기록 저장하게 되어 공정운전 데이터는 매우 풍부하게 되었다. 그러나 이러한 풍부한 데이터는 공정의 해석이나 모델링에 별로 이용되지 못한 채 보관되고만 있는 실정에 있다. 1980년대에 접어들어 인공신경망이론(Artificial Neural Networks)에 대한 연구가 활발해지면서 이용하지 못하고 보관하고 있던 많은 공정데이터를 이용하여 공정에 대한 정보를 얻을 수 있는 가능성이 제시되기 시작하였다. 이러한 방법은 인공신경망의 학습 능력에 의해 가능한 것으로, 인공신경망은 입력 데이터와 출력 데이터만을 이용하여 입출력 데이터의 상호관계를 밝혀 낼 수 있는 강력한 회귀 알고리즘이라는 개념에 바탕을 두고 있다. 이와 같이 인공신경망은 처리하고자 하는 대상 공정에 대한 구체적인 정보가 없어도 학습이라는 과정을 통해 대상에 대한 모델링이 가능하고 비교적 적용이 용이하다는 장점이 있어서, 화학공정과 같이 비선형성이 심하고 공정이 복잡하여 공정모델을 구하기 어려운 공정에 대해서도 적용 가능성이 검토되고 있다[1].

따라서 인공신경망 이론을 이용하면 공정에 대한 모델링, 해석 및 최적화도 공정운전 데이터로부터 신속하고 쉽게 수행할 수 있게 된다. 이와 같이 수학적 관계식을 사용하지 않고 공정 운전 데이터에만 기초하여 공정을 모델링하고 해석하고 최적화하는 것을 Data-Intensive 모델링 및 최적화[2]라고 하며 이와 같은 기능을 할 수 있는 도구로 인공신경망 이론이 많이 이용된다.

그런데 이러한 Data-Intensive 모델링 및 최적화에 있어서 데이터의 신뢰성은 매우 중요한 영향을 미치게 된다. 데이터의 신뢰성이란 공정의 물질수지와 에너지수지 및 변수의 물리적 제약조건을 만족하는가 하는 것이다. 그러나 일반적으로 측정 데이터는 이러한 제약조건(물질 수지, 에너지 수지 등)을 만족하지 못하는데 이러한 이유는 공정 데이터에는 항상 random error 및 gross error가 존재할 수 있기 때문이다. Random error는 평균이 0인 정규 분포를 갖는다고 가정하는 오차이며, gross error란 공정의 누출이나 측정기기의 영점 보정 불량, 측정기기의 잘못된 작동 등 예상할 수 없는 사건에 의해 발생하는 오차이다. 따라서 이러한 불일치 및 부정확한 오차를 가진 측정치를 그대로 공정 규명에 이용하게 되면 공정을 잘못 판단하게 됨으로써 잘못된 공정 모델을 구하게 되어 공정을 최적화하려 했던 목

표를 달성하지 못하게 된다. 이러한 오차를 제거하는 문제 이외에 발생하는 또 한가지 문제점은 비용의 고려 측면이나 기술적인 난점으로 인하여 모든 변수를 측정하지 못한다는 점이다. 따라서 측정치를 조정함과 동시에 측정하지 못한 변수를 추정하여 주어진 제약 조건을 만족하도록 하여야 한다. 이러한 방법을 data reconciliation이라 한다.

그러나 측정치에는 앞에서 기술한 gross error가 존재할 수 있다. 이러한 gross error가 측정 데이터에 존재하게 되면 random error를 제거하기 위해 수행되는 data reconciliation 결과가 오히려 측정 데이터보다도 신뢰성이 떨어지게 된다. 따라서 이러한 gross error를 data reconciliation을 적용시키기 전에 반드시 찾아내어 제거하는 단계가 선행되어야 하는데 이러한 방법을 gross error detection이라고 한다.

본 연구에서는 비선형성이 매우 큰 CSTR 시스템에서 얻은 측정 데이터만을 이용하여 Autoassociative 인공신경망(AAN)을 사용하여 data reconciliation 및 gross error detection에 응용 가능성을 입증하였다. 또한 AAN에 견실성을 부여한 RAAN(Robust AAN)을 도입하여 AAN을 이용한 data reconciliation 및 gross error detection과 비교하였다.

## 2. 이론적 배경

### 2-1. Data Reconciliation and Gross Error Detection

과거 30년간 data reconciliation은 상당한 관심을 받아 왔었다. Kuehn과 Davidson[3]은 변수가 모두 측정되었을 때의 문제와 측정되지 않은 변수가 있을 경우의 문제를 Lagrange multiplier를 사용하여 최적의 추정치 즉, 참 값에 근사한 값을 구하였다. Hlavacek[4]과 Mah[5]는 이 알고리즘을 실제 공정에 이용하였다. 또한 Crowe[6, 7] 등은 matrix projection을 이용하여 해를 구하는 방법을 제시하였다. Knepper와 Gorman[8]은 비선형 제약 조건을 연속적으로 선형화하여 비선형 data reconciliation 문제를 해결하였으며 Kim 등[9]은 비선형 프로그래밍 기법을 사용하여 비선형 data reconciliation 문제를 해결하였다. 지금까지 제시된 data reconciliation 알고리즘의 일반적인 가정을 기술하면 다음과 같다.

(1) 공정은 정상상태에서 작동하고 있다.

(2) 측정 오차는 정규분포를 따르며, 이 정규분포의 분산은 알려져 있다고 가정한다. 따라서 측정치에는 gross error가 존재하지 않는다고 가정한다.

선형제약조건을 갖는 데이터에서 gross error를 구

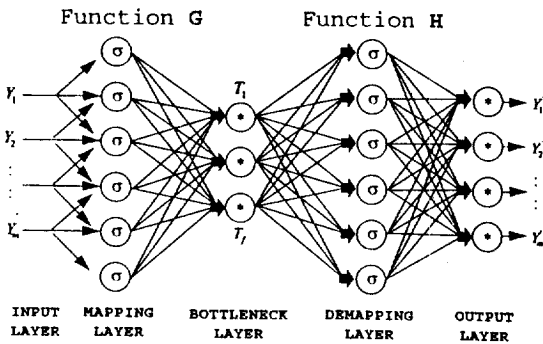


Fig. 1. Architecture of autoassociative neural networks.

명하는 방법에는 여러 가지가 있는데, 대부분 데이터의 random error는 정규분포를 갖는다는 가정을 기본으로 한 통계학적 검증을 이용하고 있다. 이 중 가장 간단한 방법은 최소자승법으로부터 구한 잔차를 검증한 후 이 중에서 검증에 실패하는 측정치를 gross error라고 간주하는 방법이다. 이 방법들은 Mah와 Thamhan[10], Crowe 등[6, 7], Stephenson과 Schewchun[11]에 의해 개발되어 왔으며, 각 알고리즘의 성능은 Iordach 등[11] 및 Serth와 Heenan[13]이 여러 가지 문제에 적용하여 연구하여 왔다. Serth와 Heenan은 outlier에 대한 검증을 반복적으로 하면 성능이 월등히 좋아지는 것을 제시하였다. 매 반복 단계에서 가장 나쁜 outlier라고 의심되는 측정치만을 gross error라고 규명한 후 데이터 중에서 제거한다. 그리고 나머지 잔차가 outlier에 대한 검증을 만족하였을 경우에만 검증은 끝나게 된다. 여러 가지 gross error 규명 방법을 비교하였을 때 이 방법(MIMT)은 계산속도나 효과면에서 가장 우수한 알고리즘으로 판명되었다.

## 2-2. Autoassociative Neural Networks

AAN(Autoassociative Neural Networks)은 역전파법 등과 같은 인공신경망의 학습방법에 의해 신경망의 입력과 출력간의 identity mapping을 위한 앞먹임 구조의 회로망으로서, 입력과 출력 사이의 bottleneck에 의해 identity mapping을 할 수 있도록 구조화된 것이 특징이다.

일반적인 AAN의 구조는 Fig. 1과 같다. 공정에서 측정된 데이터가 회로망의 입력이 되므로 입력층은 입력 데이터와 같은 수의 노드로 구성되고, 출력층은 입력 데이터가 회로망에 전파되었을 때의 결과를 출력하며, 노드의 수는 입력층과 같다. 보통 AAN의 은닉층의 개수는 3개로 구성되어 1개의 입력층과 1개의 출력층으로 AAN의 구조가 결정된다. 이 중 첫번째 은닉층을 map-

ping layer, 2번째 은닉층을 bottleneck layer, 3번째 은닉층을 demapping layer라 한다. 이 중 bottleneck layer는 회로망에서 입력층의 노드의 수보다 작게 구성되어야 하는데, 이 bottleneck에 의해 회로망의 identity mapping이 제약 받는 것을 특징으로 한다.

따라서 AAN은 1개의 은닉층으로 구성된 두 개의 회로망이 직렬로 연결된 것으로도 해석이 가능하다. Bottleneck layer를 중심으로 앞쪽을 함수 G로 표시하고, 뒷부분을 H라 표시하면, AAN은 다음과 같이 나타낼 수 있다.

$$T_i = G_i(Y), \quad i=1, \dots, f \quad (1)$$

$$Y'_j = H_j(T), \quad j=1, \dots, m \quad (2)$$

함수 G의 역할은 입력 데이터 Y를 feature 또는 factor space라는 더 작은 차원으로 project하는 기능을 하며, 함수 H는 다시 bottleneck layer의 출력인 feature로부터 입력자료를 근사하는 데이터를 만들어 내는 역할을 수행한다. 즉 데이터의 압축과 이 압축된 데이터를 푸는 과정으로 해석할 수 있으며, 이 과정에서 데이터에 포함되어 있던 오차의 제거가 가능해진다.

그러나 mapping layer와 demapping layer의 노드의 수를 결정하는 정량적인 방법은 아직 제시되어 있지 않다. 일반적으로 많은 수의 노드를 사용하지만 overfitting을 방지하기 위해서 다음과 같은 관계식에 의해 노드의 수를 결정한다[14].

$$M_1 + M_2 \leq m(n-f)/(m+f+1) \quad (3)$$

여기서  $M_1$ 과  $M_2$ 는 각각 mapping layer와 demapping layer의 노드의 수이고,  $m$ 과  $n$ 은 각각 입력 노드의 수와 데이터 세트의 수이고,  $f$ 는 bottleneck layer의 노드의 수이다.

AAN의 학습은 입력과 출력간의 sum of squared error를 최소화하는 연결 강도를 구하는 최적화 과정으로 설명되며, 다음과 같은 목적함수의 최적화 문제로 표현할 수 있다.

$$E = \sum_{p=1}^n \sum_{i=1}^m (Y_i - Y'_i)^2 \quad (4)$$

이 식 (4)의 목적함수를 만족하는 회로망의 연결강도를 구하는 방법으로는 인공신경망의 학습에 많이 이용되는 오류 역전파법(error backpropagation), conjugate gradient 알고리즘을 이용한 오류 역전파법[15]과 회로망의 연결강도를 직접 비선형 최적화 방법에 의해 구하는 최적화 기법 등이 이용된다.

### 2-3. Robust Autoassociative Neural Networks

일반적으로 효과적인 data reconciliation을 위해서는 우선 측정된 데이터에 포함되어 있을 수 있는 gross error를 제거하는 과정이 수반되어야 한다. Gross error를 제거하기 위해서는 앞에서 언급한 통계학적 방법 등에 의해 detect-identify-replace의 과정이 필요하다. 그러나 현재 많이 이용하고 있는 전통적인 방법은 gross error detection과 correction 과정이 각 분리되어 진행되어 오랜 시간이 걸리고, 항상 처리 대상에 대한 모델을 필요로 한다.

그러나 이 과정을 인공지능망 이론을 이용하여 수행하면, 대상 공정에 대한 구체적인 정보나 대상 공정에 대한 모델이 없어도 데이터에만 의존하여 모든 과정을 한번에 수행하게 되어 매우 효율적인 gross error detection과 correction을 수행할 수 있게 된다. 이처럼 gross error를 detect-identify-replace하는 과정을 한번에 진행하는 신경망으로 Robust AAN(RAAN)[16]이 있다.

RAAN은 측정치에 gross error가 없을 경우에는 일반적인 AAN의 출력과 같으나 측정치에 gross error가 존재하면 gross error가 제거된 rectified value를 신경망의 출력으로 내게 된다.

이러한 RAAN의 견실한 특성을 나타내기 위해서는 학습에 사용할 자료를 두 가지로 만들어야 한다. 우선 입력자료로서  $Y + \delta I_j$ ,  $I_j$ 는 identity matrix의  $j$ 번째 컬럼이다. 출력자료로는  $Y$ 로 하여, 신경망의 출력이  $Y$ 가 되도록 학습을 진행하여야 한다. 이를 학습의 제 1단계라 한다. 그 다음엔 RAAN에 견실성을 부여하기 위해 학습자료는  $Z = Y + \delta I_j$ 에 대해  $Y$ 를 출력하도록 구성하여 학습을 진행한다. 이 단계를 제 2학습 단계라 한다. 제 2단계 학습에서는의 범위를 제 1단계 학습 단계에서 사용한 학습 자료와 중복되지 않도록  $\pm 2\sigma \sim 0$ 이외의 값을 사용한다. 결국 정리해 보면 RAAN의 학습과정은 원래의 측정치로 구성된 일반적인 AAN을 이용하여 학습한 후, bottleneck layer 이후의 신경망의 연결강도를 고정시킨 후, 측정치에 gross error를 포함시킨 학습 자료를 mapping layer의 뉴론의 수를 증가시켜 오차를 제거할 수 있는 능력을 배양시킨 후, 입력층과 mapping layer, mapping layer와 bottleneck layer 사이의 연결강도를 최적화하는 과정으로 진행된다.

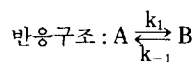
## 3. 예 제

예제로 사용된 시스템은 CSTR 시스템으로써 6개의 측정변수와 3개의 대수 제약조건으로 구성되어 있다. 측정변수는 입력농도  $A_0$ ,  $B_0$  및 입력 온도  $T_0$ 와 출력 농도  $A$ 와  $B$  및 출력온도  $T$ 이다. 반응구조는 다음과 같다.

Table 1. Physical data and uncorrupted measurement values for CSTR example

$\tau = 60$ s	$R = 1.987$ cal/(mol K)
$C_1 = 5,000$ s <sup>-1</sup>	$\Delta H_r = -5,000$ cal/mol
$C_{-1} = 1,000,000$ s <sup>-1</sup>	$\rho = 1.0$ g/L
$Q_1 = 10,000$ cal/mol	$C_p = 1,000$ cal/(g K)
$Q_{-1} = 15,000$ cal/mol	$A_0 = 0.9$ mol/L
$B_0 = 0.1$ mol/L	$T_0 = 427$ K
$A = 0.474$ mol/L	$B = 0.526$ mol/L
$T = 429.128$ K	

1차 가역반응이다.



이 시스템을 모델링하여 구한 상미분 방정식을 정상 상태로 가정하면 다음과 같은 세 개의 대수 방정식이 유도된다.

$$\frac{dA}{dt} = \frac{1}{\tau} (A_0 - A) - k_1 A + k_{-1} B = 0 \quad (5)$$

$$\frac{dB}{dt} = \frac{1}{\tau} (B_0 - B) + k_1 A - k_{-1} B = 0 \quad (6)$$

$$\frac{dT}{dt} = \frac{1}{\tau} (T_0 - T) + \frac{-\Delta H_r}{\rho C_p} (k_1 A - k_{-1} B) = 0 \quad (7)$$

여기서,  $A_0$ =입력 반응물의 농도

$A$ =출력 반응물의 농도

$B_0$ =입력 생성물의 농도

$B$ =출력 생성물의 농도

$T_0$ =입력 온도

$T$ =출력 온도

$k_1$ =정반응 속도 상수,  $C_1 \exp\left(\frac{-Q_1}{RT}\right)$

$k_{-1}$ =역반응 속도 상수,  $C_{-1} \exp\left(\frac{-Q_{-1}}{RT}\right)$

이 시스템의 물리적 데이터와 대수 방정식을 만족하는 참값은 Table 1에 나타내었다.

본 모사 실험에서는 100개 세트의 측정치를 만든 다음, 각각의 알고리즘을 적용하여 결과를 구하였다. 각 측정치는 다음과 같은 방법에 의해 구하였다.

$$z_m = x + e + d \quad (8)$$

여기서  $x$ 는 Table 1에 나타난 참 값이며,  $e$ 는 random error,  $d$ 는 gross error를 나타낸다. Gaussian random

number generator를 사용하여 random error를 만들었다. 농도의 표준편차는 0.025 mol/L, 온도에 대한 표준편차는 3 K로 정하였다. 또한 uniform pseudo random number generator를 사용하여 gross error의 개수, 위치, 크기 및 부호(+ 또는 -)를 정하였다. 0이 아닌 gross error의 크기는 다음과 같은 조건으로 구하였다.

$$0.1(z_{0j} + \varepsilon_j) \leq |\delta_j| \leq z_{0j} + \varepsilon_j \quad (9)$$

즉, gross error 크기는 참 값의 10%에서 100%를 갖게 되어 있다. 식 (9)에 따라 가장 작은 측정치는 0으로써 이 경우는 측정 불가인 경우라 할 수 있을 것이다.

CSTR에 AAN을 적용하여 data reconciliation과 gross error detection을 수행한 본 연구는 크게 1) AAN을 이용한 data reconciliation, 2) AAN을 이용한 data reconciliation 및 gross error detection, 그리고 3) Robust AAN을 이용한 data reconciliation 및 gross error detection의 3개의 부분으로 나뉘어 수행되었다. 우선 AAN을 이용한 data reconciliation의 경우에는 참값에 random error만을 추가한 측정치를 입력자료와 출력자료로 하여, 100개의 세트에 대해서 여러 가지 신경망 구조를 이용하여 학습을 실시하였으며, 이 중에서 학습과 재현단계에서 가장 좋은 성능을 보인 구조를 선택하여 사용하였다.

AAN과 Robust AAN을 이용한 data reconciliation 및 gross error detection의 경우에는 random error만 포함된 자료를 100개의 세트로, gross error도 포함된 데이터를 200개의 세트로 구성하여 모두 300개의 세트로 구성된 학습자료를 이용하였으며, 재현에는 100개의 자료 세트를 이용하였다. 이 경우에는 입력 자료에 대한 원하는 출력은 각 입력자료를 만들 때의 random error만 포함된 학습 자료로 출력 자료를 구성하였다. 역시 여러 가지 신경망 구조로 학습과 재현을 반복하여 가장 좋은 성능을 보인 구조를 선택하여 사용하였다.

각 AAN의 성능은 전체 오차의 감소를 퍼센트로 나타낸 다음과 같은 수식으로 계산하였다[17].

$$\% \text{ 전체 오차 감소} = \frac{E_1 - E_2}{E_1} \times 100 \quad (10)$$

$$E_1 = \sqrt{\sum_{j=1}^6 (z_{mj} - z_{0j})^2}$$

$$E_2 = \sqrt{\sum_{j=1}^6 (\hat{x}_j^* - z_{0j})^2}$$

여기서  $E_1$  및  $E_2$ 는 초기와 최종 root mean square 오차이며  $z_m$ ,  $z_0$ ,  $\hat{x}^*$ 은 각각 측정치, 참값, 최종 최적 추정치이다. 밑첨자 s는 각 변수를 스케일한 것을 나타

**Table 2. Performance results of data reconciliation by AAN with the data containing no gross errors**

	Learning stage	Recall stage
Sum of squared error	11.80	12.84
Average error reduction	63.73	64.99
Standard deviation	17.94	18.77
Max. value	92.81	93.22
Min. value	16.98	23.69

\*Architecture of autoassociative neural networks: 6-12-1-12-6

**Table 3. Performance results of gross error detection by AAN**

Number of gross error	Learning stage	Recall stage		
		0	1	2
Sum of squared error	30.04	9.94	11.05	10.72
Average error reduction	83.35	82.12	88.61	91.65
Standard deviation	17.50	18.12	15.86	12.95
Max. value	99.83	99.83	99.86	99.87
Min. value	10.88	29.65	40.10	42.69

\*Architecture of autoassociative neural networks: 6-12-1-12-6

낸다.

## 4. 결과 및 고찰

각 AAN의 성능은 CSTR 시스템에 100개의 무작위 추출한 데이터를 적용하여 비교하였다. 측정치는 위에서 제시하였던 random error 발생 방법으로 구하였다. 각각의 신경망의 구조에서, 재현단계에 사용한 데이터는 똑같은 random number seed를 사용하여 같은 측정치를 발생시켰다.

먼저, gross error가 포함되지 않은 데이터의 data reconciliation의 결과는 Table 2와 같다. 6-12-1-12-6의 신경망 구조를 이용한 학습과 재현단계의 SSE가 각각 11.80과 12.84로 비슷하여 학습이 제대로 이루어졌음을 알 수 있다. 오차 감소의 경우에 학습 단계보다 재현 단계가 조금 더 우수한 결과를 보이고 있다. 그러나 gross error를 포함하고 있는 측정치가 입력되었을 경우에는 이에 대처하지 못하는 맹점을 가지고 있다. 따라서 gross error를 포함하고 있는 측정치에 대해서도 능동적으로 대처하기 위해 gross error에 대한 정보를 신경망에 주기 위해, gross error가 포함되어 있는 측정치의 학습이 요구된다.

Gross error가 포함된 데이터의 처리를 위해 AAN을 이용한 경우의 결과는 Table 3과 같다. 6-12-1-12-6의

**Table 4. Performance results of gross error detection by Robust AAN**

Number of gross error	Learning stage	Recall stage		
		0	1	2
Sum of squared error	34.01	11.98	12.21	13.47
Average error reduction	92.58	92.84	95.49	96.40
Standard deviation	6.38	5.84	4.21	3.99
Max. value	99.79	99.98	99.81	99.84
Min. value	58.60	72.97	75.39	75.39

\*Architecture of robust autoassociative neural networks: 6-20-1-12-6

신경망 구조에서 학습 단계의 자료 세트의 수는 300, 재현단계에서는 100개의 자료 세트를 이용하였다. 학습 자료에서 200개의 세트는 gross error가 1개씩 포함된 것으로 구성하였다. 결과를 살펴보면, gross error가 포함되어 있지 않은 측정치의 data reconciliation에서 보다 모든 경우에서 우수한 결과를 보이고 있다. 특히 gross error의 수가 많아질수록 오차 감소의 수치가 증가하고 있으며, 이에 따라 오차 감소의 표준편차도 줄어들어, 전체적으로 gross error가 상당히 제거되었음을 확인할 수 있다.

이 6-12-1-12-6의 신경망 구조에서 bottleneck layer 이후의 구조(1-12-6)를 바탕으로 하는 Robust AAN (RAAN)을 이용한 gross error detection 및 data reconciliation의 결과는 Table 4와 같다. 최종적으로 가장 좋은 성능을 보인 구조는 6-20-1-12-6이었다. AAN의 경우와 같이 각각 300개와 100개의 자료 세트를 학습과 재현에 이용하였다. 2단계에 걸친 학습에 의해 신경망의 학습과 재현단계의 SSE는 AAN의 경우보다 다소 크게 나타났으나, 이는 큰 문제가 될 수 없다. 왜냐하면, SSE가 작을 경우에는 오히려 오차가 제거되기 보다는 오차를 학습하는 결과가 나타나기 때문이다. 오차 감소를 살펴보면, RAAN이 AAN에서보다 우수한 결과를 보이고 있으며, 특히 오차 감소의 표준편차에서는 AAN보다 매우 우수한 결과를 보여 각 세트에서 균일한 자료 처리가 이루어졌음을 알 수 있다.

이를 통해 볼 때, RAAN을 이용할 경우 학습에 소요되는 시간은 오래 걸리지만 성능면에서 볼 때 AAN보다 우수한 결과를 구할 수 있음을 알 수 있다. 또한 통계학적 기법 및 수학적 모델을 이용하지 않고서도 측정된 데이터만을 이용하여 data reconciliation이 가능함을 알 수 있었고, 역시 통계학적 검증을 이용하지 않고서도 기존의 알고리즘을 대체 또는 혼합하여 사용하여 우수한 gross error detection 및 data reconciliation 등의 자료 처리 도구로 이용될 수 있다고 생각된다.

## 5. 결 론

CSTR의 공정데이터에 포함된 random 오차와 gross 오차를 인공신경망을 이용한 data reconciliation과 gross error detection 방법으로 처리하여 본 결과 다음과 같은 결론을 얻을 수 있었다. 첫째로, 인공신경망을 이용하면 통계학적 기법에서 사용하는 수학적 모델을 이용하지 않기 때문에 간단한 방법으로 참값에 가까운 좋은 추정치를 구할 수 있었다. 두번째로, 인공신경망의 경우 위 알고리즘에서 사용하는 통계학적 검증을 사용하지 않고서도 오차 감소에서 월등한 결과를 나타내었다.

즉, 인공신경망을 이용하면 기존의 reconciliation 및 gross error detection 알고리즘에서와는 달리 시스템의 모델을 이용하지 않고, 측정치만을 이용하는 간단한 방법으로 신뢰성이 높은 추정치를 얻을 수 있었다. 따라서 이러한 신뢰성이 높은 데이터를 이용하여 공정의 모델링 및 최적화에 이용함으로써 공정성능을 향상시키는데 도움을 줄 것으로 기대된다. 실제로 인공신경망을 이용하여 원유 공정으로부터의 데이터만을 이용하여 공정을 모델링하고 최적화하는데 이용되었다[1]. 그러나 이 경우에는 데이터의 신뢰성을 높이는 데 필요한 data reconciliation 및 gross error detection 등의 선행 처리를 거치지 않고 모델링에 이용하였기 때문에 측정 데이터에 오차가 포함되었다고 가정하였을 경우, 제시한 모델이 실제 공정과 차이를 보일 수 있다고 가정할 수 있다. 그러므로 이러한 데이터만을 이용하여 공정을 모델링하고 최적화할 수 있는 인공신경망의 전처리 단계로서 autoassociative 인공신경망을 사용하여 데이터의 신뢰성을 높이게 된다면 실제 공정을 좀더 정확하게 묘사하고 설명할 수 있게 됨으로써 공정의 최적화에 많은 도움이 될 수 있을 것으로 기대된다.

## 사용기호

- E : sum of squared error, Eq. (4)
- $E_1, E_2$  : total error in(scaled) data before and after reconciliation
- f : number of bottleneck nodes
- G : mapping function, Eq. (1)
- H : demapping function, Eq. (2)
- m : number of input-output variables
- $M_1$  : number of nodes in mapping layer
- $M_2$  : number of nodes in demapping layer
- n : number of measured variables
- x : vector of true measurement
- Y : input measurement vector

$Y'$  : filtered measurement vector  
 $z$  : vector of variables of which measurements are available

#### 그리스 문자

$d$  : vector of systematic errors  
 $e$  : vector of random errors

#### 하첨자

$s$  : scaled value

#### 참고문헌

1. Eom, T. I., Kim, I. W. and Park, S. W.: "Distillation Process Optimization with Neural Networks", in Submitted to I&EC(1995).
2. Ferguson, R. B.: International Technical Symposium at SICHEM'92.
3. Kuehn, D. R. and Davison, H.: *Chem. Eng. Prog.*, **57**, 44(1961).
4. Hlavacek, V.: *Computers Chem. Engng.*, **1**, 75(1977).
5. Mah, R.: Proceedings of the Engineering Foundation Conference on Chemical Process Control(Vol. 2), New York: Engineering Foundation, 525-540 (1981).
6. Crowe, C.: *AIChE J.*, **32**, 616(1986).
7. Crowe, C., Garcia-campos, Y. and Hrymak, A.: *AIChE J.*, **29**, 881(1983).
8. Knepper, J. and Gorman, J.: *AIChE J.*, **26**, 260 (1980).
9. Kim, I. W., Liebman, M. J. and Edgar, T. F.: *AIChE J.*, **36**, 985(1990).
10. Mah, R. and Tamhane, A.: *AIChE J.*, **28**, 828(1982).
11. Stephenson, G. and Shewchuk, C.: *AIChE J.*, **32**, 247(1986).
12. Iordache, C., Mah, R. and Tamhane, A.: *AIChE J.*, **31**, 1187(1985).
13. Serth, R. and Heenan, W.: *AIChE J.*, **32**, 733(1986).
14. Kramer, M. A.: *AIChE J.*, **37**, 233(1991).
15. Leonard, J. and Kramer, M. A.: *Computers Chem. Engng.*, **14**, 337(1990).
16. Kramer, M. A.: *Computers Chem. Engng.*, **16**, 313 (1992).
17. Serth, R., Valero, C. and Heenan, W.: *Chem. Eng. Comm.*, **51**, 89(1987)