Korean J. Chem. Eng., **17**(2), 184-192 (2000)

Robust Nonlinear PLS Based on Neural Networks and Application to Composition Estimator for High-Purity Distillation Columns

June Liu*, Kwanggi Min, Chonghun Han† and Kun Soo Chang

*Mechanical Engineering Team, Samsung Engineering Co., Ltd.
Department of Chemical Engineering, Automation Research Center, Pohang University of Science and Technology
(*Received 22 September 1999 • accepted 10 January 2000*)

Abstract−The accurate and reliable on-line estimation of product quality is an essential task for successful process operation and control. This paper proposes a new estimation method that extends the conventional linear PLS (Partial Least Squares) regression method to a nonlinear framework in a more robust manner. To handle the nonlinearities, nonlinear PLS based on linear PLS and neural network has been employed. To improve the robustness of the nonlinear PLS, the autoassociative neural network has been integrated with nonlinear PLS. The integration allows us to handle the nonlinear correlation as well as nonlinear functional relationship with fewer components in a more robust manner. The application results have shown that the proposed Robust Nonlinear PLS (RNPLS) performs better than previous linear and nonlinear regression methods such as PLS, NNPLS, even for the nonlinearities due to operating condition changes, limited observations, and measurement noise.

Key words: PLS, Neural Networks, Nonlinear Correlation, Composition Estimator, Distillation Column

INTRODUCTION

The lack of on-line sensors or analyzers for measuring product quality variables has been a frequently encountered problem in industries. While process variables such as temperature, pressure, and flow rate can be easily measured on-line, key quality variables such as concentration in reactors and molecular weight of polymer are usually measured off-line. This infrequent product quality measurement has difficulties in controlling the processes to break through today's increasing pressures for production cost reductions and more stringent quality requirements. Recently, although great advances have been made in analytical instrumentation and sensors to provide on-line measurements, the use of new sensors has not yet been preferred because they still suffer from large measurement delays, narrow operating conditions, high investment/maintenance costs, low reliability and so on [Kiparissides and Papadopoulos, 1995].

For these reasons, there have been attempts to overcome the problems caused by the lack of on-line quality measurements. In these efforts, state estimation methods [Baratti et al., 1995a, b] and soft-sensor techniques [Baratti et al., 1995; Medjell and Skogestad, 1991a, b; Piovoso and Owens, 1991] have been proposed for inferring difficult-to-measure and unmeasurable quality variables. Nevertheless, there have been very few industrial applications and only a small number of pilot-scale experimental studies have been made for state estimation based on Kalman filters or extended Kalman filters. In applying state estimation methods to highly nonlinear and time varying processes, there are several technical difficulties to be overcome. They are much dependent upon the availability of a first-principle mathematical model together with the measurement of secondary variables with low noise corruption. Additional problems may be caused by the multi-rate nature of the data, obtained from sensors and analyzers. The development of rigorous mathematical models for estimating the product quality is solely dependent on state-of-the-art of mechanistic model building.

As an alternative to state estimation methods, data-based empirical models, based on input-output data pair, have been proposed. These non-mechanistic model approaches can be used for the basis of soft-sensors. In these non-mechanistic models, a regression technique is often used to infer the relationship between input and output data. Multivariate statistical data analysis methods such as partial least squares (PLS), principal component analysis (PCA), principal component regression (PCR), etc., are considered as state-of-the-art because they can provide a general model for building empirical inferential models even when the input data has a large number of process variables and these variables are highly correlated with each other. However, they also have limitations when the data has nonlinearities in nature. This is mainly due to the fact that they only provide linear models. Using linear methods in nonlinear problems can sometimes be inadequate [Palus and Dvrak, 1992; Xu et al., 1992].

Another well-known modeling method is artificial neural networks. After the universal approximation property of neural networks was proven [Hornik et al., 1989], numerous applications were made in the modeling of chemical processes [Su and McAvoy, 1993] and also in building an inferential model [Barrati et al., 1995; Piovoso and Owens, 1991]. Although the direct neural networks approach shows much better performance than that of linear techniques in building an inferential model, it also has overparameterization problems when the number of observations is smaller than that of samples [Qin and McAvoy, 1992].

For these reasons, there have been efforts (NNPLS) to take advantage of the two methods-PLS and neural networks [Orfanidis, 1990; Holcomb and Morari, 1992; Qin and McAvoy, 1992]. In

†To whom correspondence should be addressed.
E-mail: chan@postech.ac.kr

practice, these efforts failed to show great improvement over standard linear PLS for most cases and showed nearly the same performance to each other since they focused only on capturing nonlinear functional relationships between input and output data, not on nonlinear correlation of the dataset.

In this paper, we present a new method of integrating PLS and neural networks and apply it to estimating the product compositions of high-purity distillation columns using multiple temperature measurements. This problem can be considered as a good example in that it shows how the proposed method can treat nonlinear behavior and collinearity. This method takes the universal approximation property of neural networks to extend the standard linear PLS modeling method to a nonlinear framework. In addition, we incorporate the structure of autoassociative neural networks [Krammer, 1992] into this nonlinear PLS so that the method has the properties of the nonlinear PCA (principal component analysis) capturing nonlinear correlation in the data. The Robust Nonlinear PLS (RNPLS) shows better performance compared to NNPLSs as well as standard linear PLS in view of nonlinear-mapping ability, noise suppression, and capturing nonlinear correlation.

CURRENT APPROACHES FOR BUILDING AN INFERENTIAL MODEL

Building inferential model is based on a reference (calibration) dataset which can be separated into two matrices; one matrix **X** is associated with the process measurements and the other matrix **Y** is associated with the quality measurements which are not generally available in on-line. The objective is to develop an inferential model that can predict current (or future) values of quality variables using current measurements of the process variables.

In this section, we briefly discuss current methodologies used for obtaining an inferential model based on input-output data. Here we consider building static and dynamic inferential models.

1. PLS (Partial Least Squares or Projection to Latent Structure)

In inferring a relationship between two matrices **X** and **Y**, Multiple Linear Regression (MLR) method has been frequently used so far. However, if variables in the matrices are highly correlated, its prediction accuracy becomes worse. On the other hand, it is reported that PLS, one of the multivariate statistical methods, can handle this problem effectively and provide good prediction power and robustness to process noise and sensor failure [Kresta et al., 1994]; therefore, it has become the main regression technique these days. Here we briefly explain the key idea of PLS. For more detailed description, the reader should refer to Hoskuldsson [1988].

In PLS, principal components (PCs) are obtained through principal component analysis (PCA) and then a model is found by least squares regression between the two blocks, which consist of PCs. PLS modeling procedures are as follows.

① Construct two matrices **X**(N×M) and **Y**(N×K) as mentioned above (Here, N is the observation number and M and K are the number of variables in **X** and **Y**, respectively).

② Apply PCA to **X** and **Y** and find outer relations [Eqs. (1) and (2)] of the two matrices.

③ Construct two blocks that contain each PCs and find inner relation [Eq. (3)] by least squares.

$$\mathbf{X}=\mathbf{T}\mathbf{P}^T+\mathbf{E} \tag{1}$$

$$\mathbf{Y}=\mathbf{U}\mathbf{C}^T+\mathbf{F} \tag{2}$$

$$\mathbf{U}=\mathbf{T}\mathbf{B}+\mathbf{G} \tag{3}$$

where $\mathbf{B}=(\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T\mathbf{U}$

Here **E**, **F**, **G** are residual matrices, **T**, **U** are N×A matrices, **P**, **C**, **B** are M×A, K×A, A×A matrices, respectively, and A is the number of PLS components. The application area of PLS is extensive in that the monitoring of chemical processes and the composition estimation of distillation columns are representative examples of successful applications in chemical engineering. However, applying PLS to highly nonlinear problems such as high-purity, nonideal distillation columns may sometimes be inadequate since it is a linear method.

2. Artificial Neural Networks

In 1986 when more than 40 years had passed since artificial neural networks were proposed, Rumelhart and McClelland proposed a backpropagation learning algorithm, and this research provided a catalyst for much of the subsequent research in this field. In addition, Hornik et al. [1989], based on the Stone-Weierstrass theorem, showed that a two-layer feedforward network with an arbitrary large number of nodes in the hidden layer can approximate any continuous function to a desired accuracy. Through this remarkable research, neural networks have been introduced and applied to many fields including process system engineering, and have shown better results than previous methods.

In inferential model building, a direct neural network approach [Barrati et al., 1995; Piovoso and Owens, 1991] is proposed for relating matrices **Y** to **X** by neural network as:

$$\mathbf{Y}=\mathbf{N}(\mathbf{X})+\mathbf{E} \tag{4}$$

where **E** is the residual matrix after regression, and **N**(·) stands for the nonlinear function of network. Although the direct network approach performs better than linear techniques in some cases, it has similar problems to the ordinary least squares method in the case of correlated data and limited observations compared to the number of variables. In this case, when neural networks are applied for building the inferential model, they converge very slowly or fail to converge so that the prediction accuracy of the model becomes worse [Orfanidis, 1990]. Particularly when applied to problems with limited observation, the number of weights in a multiplayer network could be larger than the number of observations. Therefore, some of weights cannot be uniquely determined from the data and the direct approach leads to overfitting [Qin and McAvoy, 1992].

3. Neural Networks/PLS (NNPLS)

To supplement the drawbacks of linear PLS and neural networks, many researchers have proposed new methods that can handle nonlinearity as well as the correlation between input and output data. The early ones combined PCA with neural networks in order to decorrelate input data through PCA for fast learning speed and convergence [Orfandis, 1990]. The recent ones integrate PLS regression and neural networks to construct a unified model which can handle nonlinearity, correlated data and limited obser-

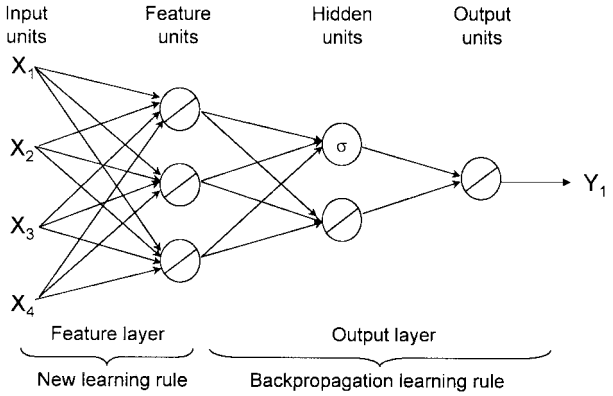

Fig. 1. Neural networks/PLS structure proposed by Holcomb and Morari [1992]. σ's and diagonals denote sigmoidal nodes and linear nodes, respectively.

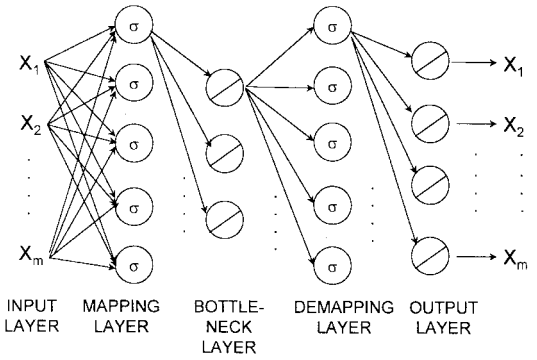

Fig. 2. Network architecture for NLPCA using autoassociative networks. σ's and diagonals have same meaning as in Fig. 1.

vations [Qin and McAvoy, 1992; Holcomb and Morari, 1992]. For example, Holcomb and Morari [1992] proposed a structure of NNPLS (Fig. 1) and its training algorithm.

The full algorithm is:

① Perform PCA to decide number of directions.
② Initialize feature layer with PCA directions.
③ Perform training on output layer; make no changes to feature layer.
④ Perform training with full networks including output layer.
⑤ If performance is unsatisfactory, add new feature nodes and go to step ③.

In Fig. 1, the feature layer acts as an outer relation in PLS, and the output layer does inner relation. Also, neurons in the feature layer correspond to principal components.

However, these nonlinear PLS methods failed to show performance improvement over linear PLS methods for most cases and showed nearly the same performance as the other. The reason is that they focused only on capturing nonlinear functional relationships between input and output data, not on nonlinear correlations in the dataset. Therefore, when NNPLS, based on linear PCA, is applied to the highly nonlinear problems, it cannot guarantee good results.

ROBUST NONLINEAR PLS (RNPLS)

1. Autoassociative Networks

Nonlinear principal component analysis (NLPCA) is a novel technique for multivariate data analysis, similar to PCA. NLPCA, like PCA, is used to identify and remove correlations among variables through dimensionality reduction, visualization, and exploratory data analysis. While PCA identifies only linear correlations, NLPCA uncovers both linear and nonlinear correlations without any restriction on the character of the nonlinearities in the data. This is because NLPCA estimates a curve or hyper-plane, not a straight line as in PCA, passing through the middle of the observations using least squares:

$$J=\sum_{i=1}^{N}\|\mathbf{x}_i-\mathbf{f}(\mathbf{s}_f(\mathbf{x}_i))\|^2 \quad (5)$$

where $\mathbf{s}_f$ does nonlinear projection to lower dimensional spaces and $\mathbf{f}$ is a remapping function from lower dimensional space to its original space. For this reason, the term, 'principal curve' is used instead of 'principal component'.

Kramer [1991, 1992] showed that AAN (AutoAssociative Networks) having five layers are adequate for NLPCA and proposed guidelines for determining the number of hidden nodes and selecting types of activation functions. Dong and McAvoy [1996] showed that NLPCA using this AAN could capture variations of the data better than PCA with fewer components. In addition, when NLPCA is properly trained, it can be used for data preprocessing so that sensor-based calculations can be performed correctly even in the presence of large sensor noises, biases, and failures [Kramer, 1991, 1992; Dong and McAvoy, 1996]. Due to this property, Kramer named it Robust AutoAssociative networks (RAAN). Its detailed structure is given in Fig. 2. In the figure, mapping layer estimates $\mathbf{s}_f$, demapping layer does $\mathbf{f}$ and bottleneck layer corresponds to principal components in PLS or PCA.

2. Robust Nonlinear PLS

As mentioned earlier, using linear methods in nonlinear problems can sometimes be inadequate [Palus and Dvrak, 1992; Xu et al., 1992]. For example, it has been shown that if PCA is applied to nonlinear problems, minor components do not always consist of white noise or unimportant variance, but they contain important information [Xu et al., 1992]. If the minor components are kept, the PCA might contain too many components. For this reason, this places severe limitations on their performance. However, NLPCA can handle this problem with fewer components because it estimates a curve or hyper-plane passing through the middle of the observations as in Eq. (5), not a straight line. Therefore, by using the structure of AAN instead of feature layers as in previous NNPLS like Fig. 1, it can capture nonlinear correlations and provide better performance with fewer components than the previous ones. In addition, based on the ability of NLPCA to reduce dimension, this new NNPLS can show better performance against sensor noise and sensor failure than previous ones and its robustness can be improved through training on purpose as mentioned earlier. For this reason, we name this new NNPLS method as Robust Nonlinear PLS (RNPLS). RNPLS's training algorithm is sequential as that of Holcomb and Morari [1992] and the detailed algorithm is as follows:

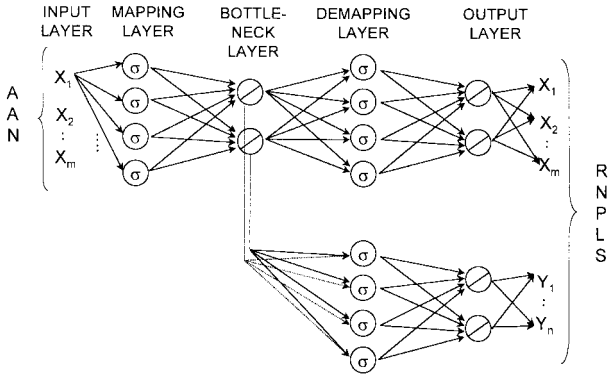

Fig. 3. Proposed RNPLS' architecture.

① Set the number of nodes in the bottleneck layer of AAN equal to 1 and initialize weights and biases of AAN.

② Perform training of AAN.

③ Perform training of RNPLS; make no changes to AAN.

④ If performance is unsatisfactory, increase the number of nodes in the bottleneck layer of AAN by one and return to step ②.

The resulting structure of RNPLS is given in Fig. 3. Here, mapping a layer of AAN corresponds to a feature layer of NNPLS [Holcomb and Morari, 1992] in Fig. 1.

APPLICATION–Composition Estimator for a Distillation Column

As mentioned in the introduction, the same problem occurs in composition control of distillation columns. Gas chromatography suffers from larger time delays of about 10 to 20 min, high investment and maintenance cost, and low reliability. This imposes severe limitations on achieving desirable control performance. The most popular alternative to analyzers is to use secondary measurements which are capable of inferring product composition. The secondary measurements used here must be reliable, inexpensive and have negligible measurement delays. For this reason, tray temperature has been frequently used, and it was reported that the use of other secondary measurements such as feed flow does not improve the estimator performance [Mejdell and Skogestad, 1993]. Therefore, until the late 80's when multivariable regression techniques (PCR and PLS) began to get attention within the process control community, the number of tray temperature measurements that should be used had been a big issue. Although the temperature at the column end is an exact indicator of composition for binary distillation columns at constant pressure, the use of a single temperature to indicate product composition is generally not reliable for many reasons [Mejdell and Skogestad, 1991a]. While use of more tray temperature measurements can give more accurate estimate of composition, choosing the right trays requires considerable insight into the column responses and this method cannot be extended to other processes. In addition, it is difficult for standard regression methods to handle a larger number of strongly correlated temperature measurements. It was found that by using all available temperature measurements, the PLS-based estimator shows good performance compared to other estimators such as the dynamic Kalman filter and the static Browsilow inferential estimator [Mejdell and Skogestad, 1991a, 1993]. It was also found that the PLS estimator can overcome nonlinearity by introducing logarithmic transformations on temperatures and compositions, but becomes sensitive to noise [Mejdell and Skogestad, 1991a, b]. In addition, various scaling methods were proposed to improve performance and robustness of the estimator [Martens and Næs, 1989; Mejdell and Skogestad, 1991a].

In the following sections, we will briefly present the definitions of the composition estimator, evaluation criteria developed to evaluate estimator's performance, scaling of variables, and variable transformation techniques used to reduce nonlinearity.

1. Problem Definition

Consider the binary distillation column with constant pressure, and feed and reflux stream as saturated liquid. Specifying each value of feed composition z_F, distillate composition y_D, and bottom product composition x_B yields a unique steady-state profile of the tray temperatures. The objective is to obtain the best estimate of the product compositions, $\hat{\mathbf{y}}$, using these steady-state tray temperatures, θ. The general form of the estimator may be written as

$$\hat{\mathbf{y}}=\mathbf{K}(\theta) \tag{6}$$

where $\hat{\mathbf{y}}=(\hat{y}_D\hat{x}_B)^T$ and the $\mathbf{K}(\cdot)$ becomes the constant matrix for PLS and nonlinear function matrix for RNPLS. For binary distillation columns with n-trays, the dimension of the PLS matrix $\mathbf{K}$ is $2\times(n+1)$ and the problem is to find optimal values of $2(n+1)$ parameters.

2. Evaluation Criteria

Explained Prediction Variance (EPV) [Mejdell and Skogestad, 1991a] is often used to evaluate the performance of the estimator:

$$\mathrm{EPV(k)}=100\times\left(1-\frac{\mathrm{MSEP(k)}}{\mathrm{MSEP(0)}}\right) \tag{7}$$

where k is the number of principal components (or factors), MSEP is the mean square error of prediction obtained from

$$\mathrm{KSEP(k)}=\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i(k)-y_i)^2 \tag{8}$$

Here, N is the number of datasets used. Also, Prediction Error Sum of Squares (PRESS) [Kresta, 1994] is used to evaluate the absolute performance:

$$\mathrm{PRESS}=\sum_{i=1}^{N}(\hat{y}_i-y_i)^2 \tag{9}$$

3. Variable Transformation Techniques

Since the composition and temperature profiles are nonlinear functions of the operating variables, many attempts have been made to overcome these nonlinearities. A simple and efficient way is to use nonlinear transformations on each variable. Logarithmic transformation of the product compositions has been proposed by several authors [Joseph and Brosilow, 1978; Skogestad and Morari, 1988; Mejdell and Skogestad, 1991a] as an effective way to linearize the static as well as dynamic responses. For binary mixtures the following transformation is used:

$$Y_D=\ln\left(\frac{y_D}{1-y_D}\right) \tag{10}$$

where y_D is the distillate composition. Various transformation tech-

niques were investigated by Mejdell and Skogestad [1991a] and they took logarithmic transformations of both the composition and the temperatures. The proposed transformation is

$$L=\ln\left(\frac{\theta-\theta_L}{\theta_H-\theta}\right) \tag{11}$$

where θ is the tray temperature, and θ_L and θ_H are reference temperatures for light and heavy components.

4. Scaling of Variables

Scaling is usually done to improve the estimate by giving each temperature a weight, corresponding to the inherent prediction ability. The most common approach is to scale variables to unit variance. The weight for the ith temperature is

$$W_i=1/s_{ci} \tag{13}$$

where s_{ci} is the standard deviation of the i-th tray temperature. In addition, other scaling methods were presented by several authors to make it more robust to noise [Martens and Næs, 1989; Mejdell and Skogestad, 1991a].

5. Example Column

The column has 15 theoretical stages with condenser and reboiler. We use a partial condenser rather than a total condenser in order to test the estimator's robustness according to pressure variations since temperature is strongly affected by pressure variations. The feed stream enters at stage 8 as saturated liquid. A binary mixture case, one light and heavy key component, is considered. In order to make the problem nonlinear, we consider a high-purity distillation column. Steady state and dynamic simulations have been performed using a rigorous process simulator, HYSYS™. The steady-state and dynamic simulation conditions are given in Tables 1 and 2. In distillation columns, nonlinearity is primarily due to the nonlinear behavior of Vapor-Liquid-Equilibrium. Fig. 4 also shows the nonlinear relationship, that is, the nonlinear principal inner relation between the score values u_1 and t_1. As shown in Fig. 4, although the linear PLS method provides a good model when the data are correlated and limited, it has difficulty in treating the nonlinearity underlying the data. To overcome the weakness of the linear PLS method, various nonlinear PLS methods have been proposed and considered as alternative methods.

Table 1. Steady-state simulation conditions

	Base case condition	Variations in steady-state reference set
Inputs		
Feed flow rate	36 m^3/hr	Constant
Feed temperature	73 °C	71-75.5 °C
Feed composition		
Methanol	50%	40-60%
Water	50%	40-60%
Outputs		
Distillate		
Methanol	99%	97-99.667%
Water	1%	0.333-3%
Bottom		
Methanol	1%	0.333-3%
Water	99%	97-99.667%

Table 2. Dynamic simulation condition

Tray size	
Diameter	10.1 cm
Weir height	1 cm
Condenser vessel volume	10.1 L
Reboiler vessel volume	8 L
Tower volume	20 L
Cooling volume	4.6 L
Liquid holdup time	5 min
Setpoint change	
Top	97% → 99% → 99.6%
Bottom	3% → 1% → 0.3%

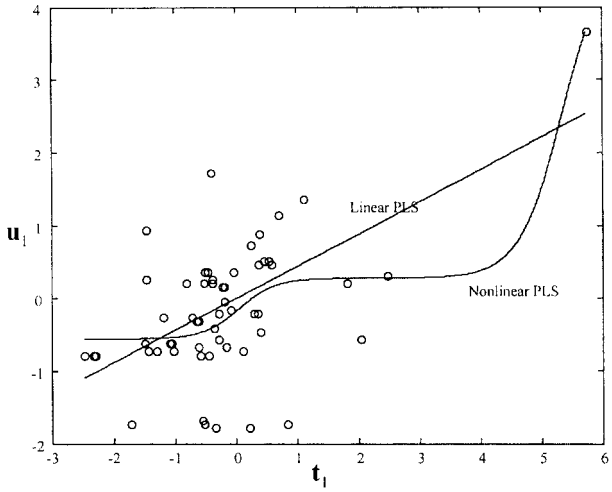

Fig. 4. Principal inner relations by PLS and NNPLS.

SIMULATION RESULTS

1. Model Building from Reference Data

Using HYSYS™ and Table 1 as simulation conditions, we obtained 64 reference datasets. The reference datasets consist of 64 different simulation runs. The y_D outputs x_B and the feed composition z_F were specified, and the corresponding steady-state temperature profiles were obtained by using HYSYS™. The data were uniformly spread around y_D, x_B, and z_F. From the data, we built four different models-PLS without transformation (PLS), PLS with transformation (PLS W/TRNS), NNPLS, and RNPLS. For all cases, mean centering and unit variance scaling [Eq. (13)] of data were done.

In transformation for PLS W/ TRNS, we take logarithmic transformation of both the composition and the temperatures [Eqs. (10) and (11)] using condenser temperature and boiling point of water as reference temperatures in Eq. (11). NNPLS model is based on Holcomb and Morari's method. In order to test NNPLS and RNPLS under the same conditions, we made the structure of the output layer of NNPLS and RNPLS the same as 2 layer networks with the same number of nodes (5-2). In building NNPLS and RNPLS, training was continued until there was no further progress in reducing network output error. In addition, training was repeated with different random initial values of weights and biases

to increase the probability of finding the global optimum. For more efficiently training the network, the Levenberg-Marquardt algorithm [Scales, 1985], a variation of Newton's method, is used.

Before determining the model dimension (neural network structure) of RNPLS, we must first select the number of mapping and demapping nodes of autoassociative networks. Since our training algorithm of RNPLS focused only on bottleneck nodes, the number of bottleneck nodes which correspond to overall model dimension of RNPLS cannot be determined before mapping and demapping layers' sizes are determined. In fact, because the number of nodes in each layer cannot be selected independently, it is a very difficult task to determine networks' structure simultaneously. Kramer [1991, 1992] proposed the guideline for determining the structure of autoassociative networks as follows:

$$M_1+M_2 \ll m(n-f)/(m+f+1) \tag{14}$$

$$f \ll (m,n); \quad m < M_1, M_2 \tag{15}$$

where M_1 and M_2 are the number of nodes in mapping and demapping layers, respectively, and f is the number of nodes in the bottleneck layer. In our example, the number of variables in **X**(m) is 16 and the number of observations (n) is 64, and we set f equal to one. So, $16<M_1$, M_2 and $M_1+M_2 \ll 56$. To prevent over-parameterization, we use final prediction error (FPE) and Akaike's information theoretic criterion (AIC) that express trade-offs between fitting accuracy and the number of adjustable parameters in the model [Ljung, 1987; Soderstrom and Stoica, 1989]. Minimization of these criteria identifies models that are neither over-parameterized nor under-parameterized. We tested different numbers of mapping layers in these experiments, but for simplicity we set M_1 and M_2 equal. Table 3 shows the number of mapping nodes, error, and the model selection criteria of FPE and AIC for each model. These results show that 19 mapping nodes are required to achieve the desired performance. We fixed the number of mapping and demapping nodes as 19 in determining the model dimension of RNPLS later.

The number of PLS dimensions for good prediction is usually determined by cross-validation [Wold, 1978]. We also used this method in determining the dimension of NNPLS and RNPLS. In cross-validation, the reference data set is partitioned into several subsets (usually 4-7 subsets). Using all subsets excluding one of these subsets, a new model with another PLS dimension is determined. This model is used to predict the **Y** values in the remaining subset, and the PRESS for this subset is calculated. This procedure is repeated until each data subset has been excluded once and only once; then, the PRESS values for each subset are summed to obtain the overall PRESS. If this overall PRESS value shows a benefit in adding an additional PLS dimension to the model, then the entire data set is used to recalculate the final values of the latent variables. The optimal model order corresponds to a minimum in the overall PRESS. Equivalently, *Percentage of explained variance* can be used instead and it is calculated as follows.

$$\textit{Percentage explained variance} = \frac{\text{SSTO}-\text{PRESS}}{\text{SSTO}} \tag{16}$$

where SSTO is the total sum of squared error calculated from

$$\text{SSTO} = \sum_{i=1}^{N} (y_i - \bar{y}) \tag{17}$$

and $\bar{y}$ is mean value of y. Comparing Eqs. (7) with (16), one can see that the two equations are the same since $\hat{y}_i(0)$ is equal to $\bar{y}$. In this case, when the maximum value of *EPV* (or *Percentage of explained variance*) is reached, the model order is considered as optimal. In practice, this value increases continuously as the model dimension increases. However, the optimal model dimension can easily be found since the increment in *EPV* value decreases drastically after the optimal dimension. For each modeling method, the results of the model determination procedure using *EPV* or *Percentage explained variance* are summarized in Tables 4, 5, 6, and 7, and the optimal model dimension is represented by boldface in the tables. For detailed procedure and calculations of cross-

Table 3. Determination of number of mapping nodes of autoassociative networks

No. of mapping nodes of AAN	Error E	FPE	AIC
17	867.14	1.770	0.371
18	756.29	1.736	0.304
19	**565.29**	**1.475**	**0.0834**
20	528.81	1.585	0.0860
21	499.62	1.747	0.0992
22	468.31	1.951	0.105

Table 4. Cross-validation information used to determine model dimension (PLS)

Model dimension (no. of PCs)	Percentage variance explained or EPV	
	X	**Y**
1	66.3	42.3
2	90.3	89.4
3	99.2	97.5
4	**99.8**	**99.7**
5	99.9	99.9

Table 5. Cross-validation information used to determine model dimension (PLS W/TRNS)

Model dimension (no. of PCs)	Percentage variance explained or EPV	
	X	**Y**
1	64.4	42.1
2	89.1	91.0
3	**99.9**	**99.8**
4	100.0	100.0

Table 6. Cross-validation information used to determine model dimension (NNPLS)

Model dimension (no. of feature units)	Percentage variance explained or EPV	
	X	**Y**
1	63.8	90.2
2	94.1	96.1
3	**98.4**	**100.0**

Table 7. Cross-validation information used to determine model dimension (RNPLS)

Model dimension (no. of bottleneck nodes)	Percentage variance explained or EPV	
	X	**Y**
1	64.9	66.3
2	**99.8**	**100.0**

Table 8. Comparison of model dimension

Model	Model dimension	Percentage variance explained or EPV	
		X	**Y**
PLS	4	99.8	99.7
PLS W/TRNS	3	99.9	99.8
NNPLS	3	98.4	100.0
RNPLS	2	99.8	100.0

validation, refer to Wold [1978].

As model building results, PLS and PLS W/TRNS have 4 and 3 principal components, respectively, and network structures for NNPLS and RNPLS are (16-3-5-2) and (16-19-2-5-2). To compare both models' dimension and their prediction power, the best model for each method is summarized in Table 8. From the table, RNPLS is the best since it can explain the process variance nearly perfectly with remarkably fewer principal components (model dimension) than others, and from which it can be known that dimensional reduction ability of RNPLS is the best among the models as mentioned earlier.

2. Prediction with New Data

To test static and dynamic performance of the models, we obtained test datasets-another 32 sets of steady-state data and setpoint change data-from Tables 1 and 2. For both cases (static and dynamic), we prepared noisy data which contain three different noise levels (±0.1 °C, ±0.2 °C, ±0.3 °C) to show robustness of models. To produce these noisy data, we added randomly distributed noise with magnitude of 0.1, 0.2, and 0.3 to tray temperatures. Also, in order to test estimators' own robustness to noise, we did not train NNPLS and RNPLS to these noisy data.

2-1. Static Performance

The static performance of models is summarized in Tables 9 and 10. For top composition estimation (Table 9), PLS shows the worst performance, while NNPLS and RNPLS show the best performance for both cases when there is no noise in the data. However, as the noise level increases, the performance of PLS W/TRNS becomes worse than that of PLS. These same results have already been reported [Mejdell and Skogestad, 1991a]. However, it should be noted that the robustness of PLS is worse than that of PLS W/TRNS in bottom composition estimation. This means that bottom composition estimation is less affected by noise added to tray temperatures. Therefore, estimation performance of bottom composition by logarithmic transformation is not much deteriorated because bottom composition is less correlated to tray temperatures than top composition is. To investigate this, degree of correlation of compositions to tray temperatures is shown in Figs. 5 and 6. Comparing the two figures clearly shows that top composition is much more correlated than bottom composition, and this holds for estimation in multi-component distillation columns. Also, as noise level increases, NNPLS shows same trends as PLS in its performance for both cases where RNPLS maintains the best performance. The reason is that because NNPLS uses line-

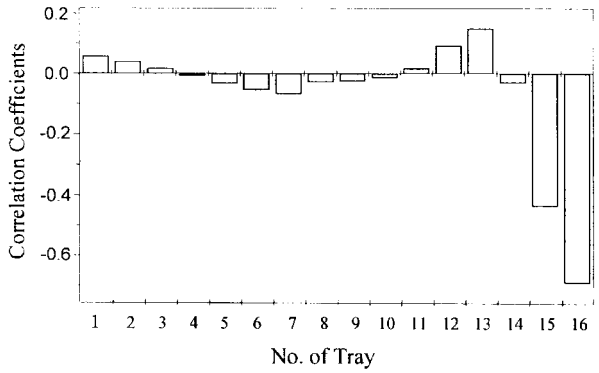

Fig. 5. Cumulative contribution of tray temperatures to bottom composition (From PLS model coefficients).

Table 9. Static performance of estimators to steady-state test data (Top composition)

	PRESS				EPV			
	PLS	PLS W/TRNS	NNPLS	RNPLS	PLS	PLS W/TRNS	NNPLS	RNPLS
No noise	7.70e-6	2.07e-6	9.97e-8	1.05e-7	99.7	99.9	100.0	100.0
0.1 noise	3.56e-5	5.41e-5	2.61e-5	9.49e-7	98.7	98.1	99.1	100.0
0.2 noise	1.18e-4	2.05e-4	1.10e-4	5.87e-6	95.8	92.7	96.1	99.8
0.3 noise	1.87e-4	5.04e-4	1.70e-4	9.68e-6	93.3	82.0	93.9	99.7

Table 10. Static performance of estimators to steady-state test data (Bottom composition)

	PRESS				EPV			
	PLS	PLS W/TRNS	NNPLS	RNPLS	PLS	PLS W/TRNS	NNPLS	RNPLS
No noise	1.00e-5	2.08e-5	1.74e-8	1.83e-8	99.6	99.3	100.0	100.0
0.1 noise	1.12e-5	2.72e-5	2.37e-6	1.20e-7	99.6	99.0	99.9	100.0
0.2 noise	4.91e-5	4.56e-5	1.10e-5	5.41e-7	98.2	98.4	99.6	100.0
0.3 noise	7.04e-5	3.32e-5	2.03e-5	1.18e-6	97.5	98.8	99.3	99.9

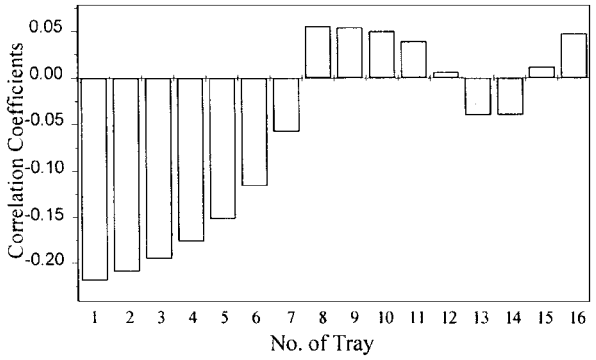

Fig. 6. Cumulative contribution of tray temperatures to top composition (From PLS model coefficients).

ar PCA, its noise cancellation ability is the same as that of PLS and is inferior to RNPLS using NLPCA. In the view of prediction power and robustness, RNPLS shows the best results for estimating the top and bottom compositions.

2-2. Dynamic Case

For most cases, the dynamic performance of the estimator is more important than static performance because an inferential model is frequently used for control purposes. Therefore, in addition to good static performance, good dynamic performance is essential for inferential models. Dynamic performances according to set-point changes are shown in Figs. 7 and 8. On the contrary to static case, PLS W/TRNS shows worse performance than that of PLS for noise-free data. This can be explained as that controller

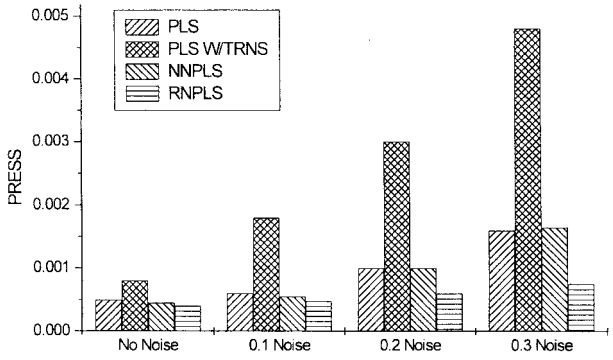

Fig. 7. Dynamic performance of estimators to setpoint change data (Top composition).

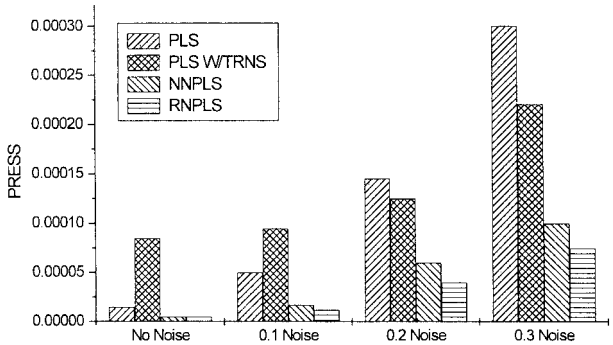

Fig. 8. Dynamic performance of estimators to setpoint change data (Bottom composition).

actions and pressure variations during setpoint change induce small temperature variations in dynamic data, and these variations act as a noise so that the performance of PLS W/TRNS is worse than that of PLS. As in the static case, the robustness of PLS W/TRNS is better than that of PLS in bottom composition estimation, and NNPLS shows the same robustness as that of PLS. Also, the prediction power and robustness of RNPLS exceed all other methods for the top and bottom composition estimations. This means that RNPLS captures nonlinear correlation well, which cannot be captured by other methods due to its small variances. Although this nonlinear correlation is minor, it has important information about system dynamics.

CONCLUSIONS

In this work, we proposed a new method that extends linear PLS to nonlinear frameworks based on neural networks. In addition to its nonlinear mapping ability, the proposed RNPLS provides a more parsimonious model than other nonlinear PLS methods because RNPLS based on NLPCA can capture nonlinear correlations with fewer components. For composition estimation in high-purity binary distillation columns, RNPLS shows good prediction power and robustness in both static and dynamic cases under the existence of measurement noise. Especially when used for control, RNPLS can guarantee good control performance since it shows excellent performance for dynamic cases compared to others. Also, it should be noted that the characteristics of composition estimation at the top and bottom are different; therefore, these characteristics must be considered when one designs an estimator using conventional PLS methods.

ACKNOWLEDGEMENT

This work was supported by the Brain Korea 21 project.

NOMENCLATURE

A : number of PLS components
B : least square solution, $\mathbf{B}=(\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T\mathbf{U}$
C : matrix consisting of loading vector of **Y**
E, F, G : residual matrices
f : vector of smooth functions mapping from $\Re^A$ to $\Re^M$
J : objective function to minimize in NLPCA
K : number of variables (columns) in **Y**
K : inferential model mapping from **X** to **Y**
L : transformed temperature
M : number of variables (columns) in **X**
N : number of observations (number of rows in **X** or **Y**)
N : nonlinear mapping found by neural networks
n : number of trays
P : matrix consisting of loading vector of **X**
$\mathbf{s}_f$: vector of smooth functions mapping from $\Re^M$ to $\Re^A$
T : matrix consisting of score vector of **X**
T : temperature
U : matrix consisting of score vector of **Y**
X : process measurements matrix
$\mathbf{x}_i$: i-th row of **X**

x_B : bottom composition
$\mathbf{Y}$: quality measurements matrix
Y_D : transformed composition
$\hat{y}$: predicted y by model
y_i : i-th row of $\mathbf{Y}$
y_D : distillate (top) composition
z_F : feed composition

Greek Letters

θ : secondary variables (here, tray temperature)
σ : sigmoid activation function

Superscript

b : boiling point

Subscripts

B : bottom product
D : distillate
F : feed
f : vector of functions, $\mathbf{f}$
H : heavy key component
i : observation index (1,..., N)
L : light key component

REFERENCES

Baratti, R., Bertucco, A., Da Rold, A. and Morbidelli, M., "Composition Estimators for Distillation Column : Theory and Experiments," DYCORD '95, Copenhagen, Denmark, 475 (1995).

Baratti, R., Bertucco, A., Da Rold, A. and Morbidelli, M., "Development of a Composition Estimator for Binary Distillation Columns. Application to a Pilot Plant," *Chem. Eng. Sci.*, **50**(10), 1541 (1995).

Baratti, R., Vacca, G. and Servida, A., "Neural Network Modeling of Distillation Columns," *Hydrocarbon Processing*, June, 35 (1995).

Dong, D. and McAvoy, T. J., "Nonlinear Principal Component Analysis-Based on Principal Curves and Neural Networks," *Computers chem. Engng.*, **20**(1), 65 (1996).

Holcomb, T. R. and Morari, M., "PLS/Neural Networks," *Computers Chem. Engng.*, **16**(4), 393 (1992).

Hornik, K., Stinchcombe, M. and White, H., "Universal Approximation of an Unknown Mapping and its Derivatives using Multilayer Feedforward Networks," *Neural Networks*, **3**, 551 (1990).

Hoskuldsson, A., "PLS Regression Methods," *J. Chemometrics*, **2**, 211 (1988).

Joseph, B. and Browsilow, C. B., "Inferential Control of Processes," *AIChE J.*, **24**, 485 (1978).

Kiparissides, C. and Papadopoulos, E., "Real-time Optimization and Model-based Control of Polymer Reactors," Methods of Model Based Process Control, Berber, R., eds., Kluwer Academic Publisher, Netherlands, 495 (1995).

Kramer, M. A., "Autoassociative Neural Networks," *Computers chem. Engng.*, **16**(4), 313 (1992).

Kramer, M. A., "Nonlinear Principal Component Analysis Using Autoassociative Neural Networks," *AIChE J.*, **37**(2), 233 (1991).

Kresta, J. V., Marlin, T. E. and MacGregor, J. F., "Development of Inferential Process Models using PLS," *Computers chem. Engng.*, **18**(7), 597 (1994).

Ljung, L., "System Identification-Theory for User," Prentice Hall, Englewood Cliffs, NJ (1987).

Martens, H. and Næs, T., "Mutivariate Calibration," John Wiley & Sons, New York (1989).

Mejdell, T. and Skogestad, S., "Composition Estimator in a Pilot-Plant Distillation Column Using Multiple Temperatures," *Ind. Eng. Chem. Res.*, **33**, 2555 (1991).

Mejdell, T. and Skogestad, S., "Estimation of Distillation Compositions from Multiple Temperature Measurement Using Partial-Least-Squares Regression," *Ind. Eng. Chem. Res.*, **33**, 2543 (1991).

Mejdell, T. and Skogestad, S., "Output Estimation Using Multiple Secondary Measurements: High-Purity Distillation," *AIChE J.*, **39** (10), 1641 (1993).

Orfandis, S. J., "Gram-Schimidt Neural Networks," *Neural Computation*, **2**, 116 (1990).

Palus, M. and Dvrak, I., "Singular-value Decomposition in Attractor Reconstruction: Pitfalls and Precautions," *Physica D*, **55**, 221 (1992).

Piovoso, M. J. and Owens, A. J., "Sensor Data Analysis using Artificial Neural Networks," Int. Conf. Chem. Process Control, CPC IV, Texas (1991).

Qin, S. J. and McAvoy, T. J., "Nonlinear PLS Modeling Using Neural Networks," *Computers chem. Engng.*, **16**(4), 379 (1992).

Rumelhart, D. E. and McClelland, J. L., "Parallel Distributed Processing, Explorations in the Microstructure of Cognition, Vol. 1: Foundations," MIT Press, Cambridge (1986).

Scales, L. E., "Introduction to Non-Linear Optimization," Springer-Verlag, New York (1985).

Skogestad, S. and Morari, M., "Understanding the Dynamic Behavior of Distillation Columns," *Ind. & Eng. Chem. Res.*, **27**(10), 1848 (1988).

Soderstrom T., and Stoica, P., "System Identification," Prentice Hall, Englewood Cliffs, New Jersey (1989).

Su, H. and McAvoy, T. J., "Integration of Multilayer Perceptron Networks and Linear Dynamic Models : A Hammerstein Modeling Approach," *Ind. Eng. Chem. Res.*, **32**, 1927 (1993).

Wold, S., "Cross-validity Estimation of the Number of Components in Factor and Principal Components Models," *Technometrics*, **20**(4), 397 (1978).

Xu, L., Oja, E. and Suen, C. Y., "Modified Hebbian Learning for Curve and Surface Fitting," *Neural Networks*, **5**, 441 (1992).