USE OF AN AUTOASSOCIATIVE NEURAL NETWORK FOR DYNAMIC DATA RECONCILIATION

Shuanghua Bai, Jules Thibault and David D. McLean

Department of Chemical Engineering
University of Ottawa, Ottawa, Ontario, Canada K1N 6N5

Abstract: The technique of dynamic data reconciliation has been previously studied in the literature and shown to be an effective tool to better estimate the true values of process variables by using information from both measured values and process models. Real-time implementation of dynamic data reconciliation involves solving complex optimization problem, leading to large computation time. This paper presents a study on the use of a dynamic **A**uto**a**ssociative **N**eural **N**etwork (AANN) for dynamic data reconciliation. Once trained, the AANN can be directly used for online signal validation. Closed-loop performance of the AANN for both linear and nonlinear processes was evaluated using simulations of two storage tank processes. The AANN provided accurate estimates of measured values for the two processes studied in this investigation. *Copyright © 2005 IFAC*

Keywords: Data reconciliation, dynamic neural network, controller performance.

1. INTRODUCTION

Signal validation plays an important role in real-time plant operation since measured signals of process variables are often contaminated by measurement errors that include random and nonrandom errors. The random error is often referred to as measurement noise that can be approximated by a Gaussian distribution. On the other hand, the nonrandom errors include measurement outliers and biases. When the corrupted measurements are used for process monitoring and control, the knowledge of the true state of the process is inaccurate and the performance of controllers may be deteriorated. Therefore, it is imperative to validate the measured signals prior to their use as inputs to the controllers or in management system. The estimation of system variables is traditionally performed using filters such as Exponentially Weighted Moving Average (EWMA) or Moving Average (MA) filters. These filters use measurement temporal redundancy, meaning that past measurements are used to estimate the current state of the process. The simple EWMA or MA filter performs well for steady-state or slow dynamic processes. However, for processes having significant dynamics and for the purpose of detecting nonrandom measurement errors, model-based filters are preferable. The model-based filters employ process dynamic models, such that both measurement temporal and spatial redundancies are used to estimate the current state of the dynamic system. A well known model-based filter is the Kalman filter that employs stochastic linear state-space process and measurement models. The most attractive advantage of the Kalman filter lies in its optimal estimation in the sense of minimum mean squared prediction errors (Kamen and Su, 1999). However, the optimality of the Kalman filter requires two restrictive prerequisites, linear state-space models and independent Gaussian white noise for both process and measurements. In its implementation, the Kalman filter is commonly tuned by adjusting the process and measurement noise covariances and treating them as design parameters. For nonlinear processes, Extended Kalman Filters (EKF) are used whereby nonlinear process models are linearized at each sampling time. However, in chemical

engineering, the applications of EKF have met some problems, such as divergent and unreliable results and difficulty in tuning the filter (Wilson et al., 1998).

An alternative approach to the Kalman filter for real-time plant signal validation is to use Dynamic Data Reconciliation (DDR) technique. Data reconciliation integrates information originating from measurements and process models to provide more accurate estimates of process variables. More importantly, the reconciled data are consistent with relationships that exist between process variables, such as mass and heat balances. Data reconciliation techniques were initially developed for steady-state processes in order to calculate process mass and heat balances (Kuehn and Davidson, 1961; Mah and Stanley, 1976; Crowe et al., 1983; and Crowe, 1986). Recent work (e.g., Darouach and Zasadzinski, 1991; Liebman et al., 1992; Albuquerque and Biegler, 1996; Bagajewicz and Jiang, 1997) extended the concept of data reconciliation for dynamic processes. In the application of data reconciliation techniques, measured variables are optimally adjusted to satisfy exactly process models and constraints. In practice, however, both measurements and process models are prone to errors such that the following objective function is minimized to perform DDR.

$$\begin{aligned} J(\hat{\mathbf{y}}_t, \hat{\mathbf{z}}_t) &= (\mathbf{y}_t - \hat{\mathbf{y}}_t)^T \mathbf{V}^{-1} (\mathbf{y}_t - \hat{\mathbf{y}}_t) + \\ &\quad \mathbf{f}^T(\hat{\mathbf{y}}_t, \hat{\mathbf{z}}_t) \boldsymbol{\Omega}^{-1} \mathbf{f}(\hat{\mathbf{y}}_t, \hat{\mathbf{z}}_t) \\ \text{subject to} \quad & \mathbf{y}_{t,l} \le \hat{\mathbf{y}}_t \le \mathbf{y}_{t,u} \\ & \mathbf{z}_{t,l} \le \hat{\mathbf{z}}_t \le \mathbf{z}_{t,u} \end{aligned} \tag{1}$$

where $\mathbf{y}_t$ is a vector of measured values of process variables at time t , $\hat{\mathbf{y}}_t$ is a vector of reconciled values of the M measured process variables, $\mathbf{V}$ is a covariance matrix of the measurement error. $\mathbf{f}(\hat{\mathbf{y}}_t, \hat{\mathbf{z}}_t)$ is a functional vector of process algebraic or discretized differential equations, $\hat{\mathbf{z}}_t$ is a vector of unmeasured variables or model parameters simultaneously estimated by the DDR algorithm. $\mathbf{y}_{t,l}$, $\mathbf{z}_{t,l}$, $\mathbf{y}_{t,u}$ and $\mathbf{z}_{t,u}$ are vectors of lower and upper bounds. $\boldsymbol{\Omega}$ is a covariance matrix of model errors whose elements are often treated as tuning parameters due to the difficulty to determine them.

The benefits of using DDR for real-time process monitoring and control have been reported. Ramamurthi et al. (1993) proposed that DDR led to better closed-loop performance of nonlinear predictive controller. Soderstrom et al. (2000) implemented a real-time dynamic data reconciliation strategy to improve inventory calculations for a diluent plant. Abu-el-zeet et al. (2002) claimed that DDR, in conjunction with systematic bias detection, enhanced a model predictive control scheme. Bai et al. (2004) applied a data reconciliation filter embedded in PI feedback control loops for the control of a binary distillation column, and demonstrated that the controller performance was significantly improved by the DDR. The benefits of using DDR for real-time signal validation include improving controller performance, reducing the variability of controlled variable at process nominal steady-state, estimating unmeasured process variables, as well as detecting measurement biases.

The DDR algorithm requires nonlinear programming at each sampling time such that long computer time may be needed for complex processes. To reduce the computation, this paper proposes to train an Autoassociative Neural Network (AANN) to directly perform data reconciliation. Once trained, the neural network can perform data reconciliation without any iteration, and the neural DDR becomes more suitable for real-time applications.

2. AUTOASSOCIATIVE NEURAL NETWORK

2.1 Architecture of AANN

An AANN is a network having similar architecture as a conventional feedforward neural network composed of an input layer, hidden layers and an output layer as illustrated in Figure 1. The input and output layers have the same number of neurons determined by the nature of problem being solved. The first and third hidden layers, called mapping and demapping layers respectively, contain a relatively larger number of neurons. The second hidden layer, called bottleneck layer, contains less neurons. The AANN is trained to reproduce its inputs as its outputs. The key feature of AANN is to perform data compression by the bottleneck layer. The input layer, mapping and bottleneck layers compress the input information to a lower dimension, then the demapping and output layers regenerate the main underlining features of the original information of the inputs. The mapping/demapping process enables the network to represent the input information in a compressed form that can often reveal the essence of the data. For training AANNs, the input and the target vectors presented to the network are identical, and the mean of squared errors between the network outputs and its inputs are minimized. After successful training, the AANN can be regarded as a filter that can be used for filtering random noise and detecting fault sensors for steady-state or stationary processes (Kramer, 1992).

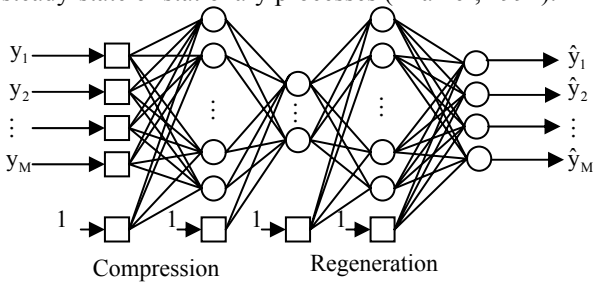

Fig. 1. Architecture of AANN.

Du et al. (1997) used an autoassociative neural network for nonlinear steady-state data reconciliation. They developed a mass-balance-

related AANN scheme to rectify flow rates and mass compositions. The mass balance of a system was incorporated directly into the objective function such that the redundant information for the measurements was taken into account in training the network. An identical objective function to Equation (1) was used to train the AANN. With the redundant information provided by the process models, the performance of the AANN is expected to perform significantly better than that of an AANN trained using only the first term of Equation (1) for measurement noise reduction, systematic bias detection, and estimation of unmeasured process variables.

2.2 Dynamic AANN for DDR

For dynamic data reconciliation, the static structure of the feedforward AANN of Figure 1 must be modified to encapsulate the dynamics of the process. The AANN architecture of Figure 2 has been selected. To capture the dynamics of the process, the reconciled output $\hat{y}_t$, delayed a number of times, is fed back to the input layer. This AANN incorporates both temporal and spatial patterns. In Figure 2, D represents the required number of time delays for the process output variables, and u_{t-d}, …, u_{t-d-I} represent the inputs with time delay of d, d+1, …, d+I.

It is important to note that the number of neurons in the input and output layers of the dynamic AANN are not necessarily identical as they were in the static AANN, but their number is problem-dependent. In addition, the number of neurons in each hidden layer is determined experimentally.
Prior information relating the input and output variables of the process is used to train the AANN for DDR. The objective function to train the network can be written as

$$J = \frac{1}{N}\sum_{t=0}^{N}\Big[\left(\mathbf{y}_t-\hat{\mathbf{y}}_t\right)^T\mathbf{V}^{-1}\left(\mathbf{y}_t-\hat{\mathbf{y}}_t\right)+ \mathbf{f}^T\left(\hat{\mathbf{y}}_t,\hat{\mathbf{y}}_{t-1},...,u_{t-d},u_{t-d-1},...\right)\boldsymbol{\Omega}^{-1} \mathbf{f}\left(\hat{\mathbf{y}}_t,\hat{\mathbf{y}}_{t-1},...,u_{t-d},u_{t-d-1},...\right)\Big] \quad (2)$$

where $\mathbf{f}\left(\hat{\mathbf{y}}_t,\hat{\mathbf{y}}_{t-1},...,u_{t-d},u_{t-d-1},...\right)$ is a functional vector of process dynamic models. In the first iteration in training of the dynamic AANN, the output vector fed back to the input layer is not known, but can be assigned the raw measurements. Then, the network is trained until satisfactory convergence criterion is met. After the first iteration, the vectors of the network outputs are fed back as the inputs, and then the network is trained again. After several recurrent iterations, the feedback vectors and the objective function will not change, indicating the training of the dynamic network has been completed. After the successful training, the dynamic AANN can be incorporated to the plant for real-time data reconciliation.

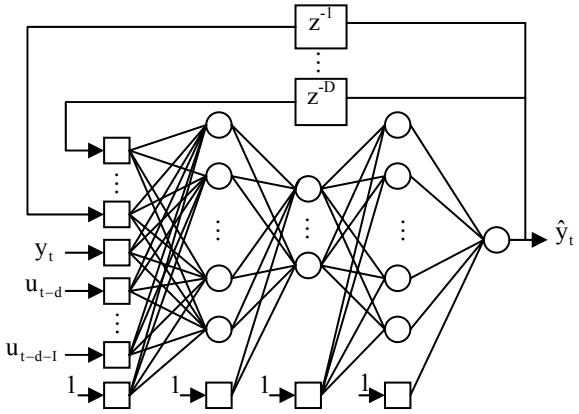

Fig. 2. Architecture of an AANN for dynamic system.

3. EXAMPLES

To illustrate the use of autoassociative neural networks for dynamic data reconciliation, two examples were investigated. One example is a cylindrical storage tank process. The other is a spherical storage tank process. Both processes are controlled by conventional PI controllers. The developed dynamic AANNs were used as DDR filters to provide better estimates of the controlled variable for the controller. The performance of the dynamic AANNs was examined with closed loop control of the processes as illustrated by Figure 3.

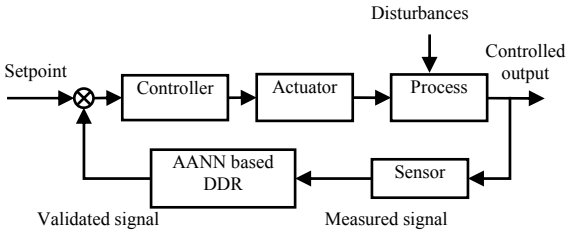

Fig. 3. Scheme for implementing an AANN for signal validation embedded in a control loop.

3.1 Cylindrical Storage Tank Process

The schematic diagram of the cylindrical storage tank process is shown in Figure 4. A PI controller was used to regulate the liquid level of the tank by manipulating the outlet flow. The feed flow to the tank was measured but not controlled. The sampling interval was 1 min. The nominal feed flow to the tank was 1.8 m^3/h. Random Gaussian white noise with standard deviations σ_{Fi}=0.09 m^3/h for the feed flow and σ_H=0.1 m for the tank level corrupted the measurements. The dynamics of the measuring device and control valve were neglected. The model of the process is given by the mass balance

$$A\frac{dH}{dt} = F_i - F_o \quad (3)$$

where H is the liquid height, F_i and F_o are the feed and outlet flow rates, and A is the cross-sectional area of the tank.

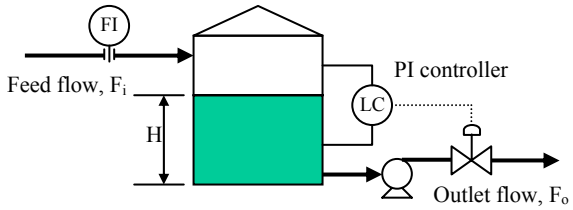

Fig. 4. Cylindrical storage tank process.

Under open-loop conditions and with measurement noise, the tank was perturbed by random step changes in the feed and outlet flow rates. 1000 samples were simulated. 90% of the samples were used to train an AANN and the remaining 10% was used for network validation. An AANN having the structure [5, 6, 3, 6, 2] determined experimentally, shown in Figure 5, was trained by minimizing the objective function

$$J = \sum_{p=1}^{N} \left\{ \frac{1}{\sigma_{F_i}^2}\left(F_{i,t-1} - \hat{F}_{i,t-1}\right)^2 + \frac{1}{\sigma_H^2}\left(H_t - \hat{H}_t\right)^2 + \frac{1}{\sigma_{Model}^2}\left[\hat{H}_t - \hat{H}_{t-1} - \frac{\Delta t}{A}\left(\hat{F}_{i,t-1} - F_{o,t-1}\right)\right]^2 \right\} \quad (4)$$

where the discretized model of Equation (3) was incorporated into the objective function and Δt represents the sampling time interval.

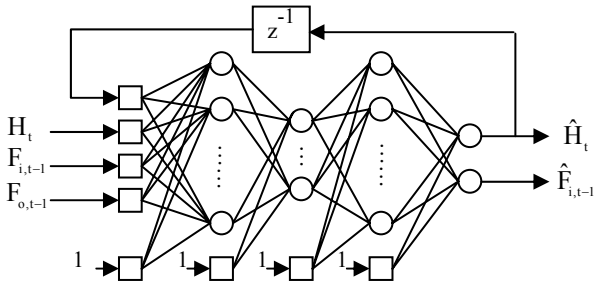

Fig. 5. Structure of dynamic AANN for the cylindrical storage tank process.

Because the variance of the model error was not known, the value of σ_{Model}^2 in the objective function of Equation (4) was treated as a tuning parameter. For a value of σ_{Model}^2, the network was trained completely, and then tested using the validation data sets. The variances of the reconciled tank level $\hat{H}_t$ and the reconciled feed flow $\hat{F}_{i,t}$, evaluated using the validation data set, are presented in Figure 6. The variance of $\hat{H}_t$ decreased and then increased with an increase of σ_{Model}^2. When σ_{Model}^2 was set to small values, the model mismatch distorted the reconciled tank level, so that the variance of $\hat{H}_t$ increased dramatically. On the other hand, when σ_{Model}^2 was set to larger values, the measurements were not constrained severely such that the variance of $\hat{H}_t$ approached the variance of the raw values. However, the variance of the reconciled feed flow was not affected by the change of σ_{Model}^2, and nearly matched the variance of raw measurements, because the noise in the disturbances was damped by the inertia of the process and the model had nearly no constraint on it.

The optimal value for the tuning parameter was $\sigma_{Model}^2 = 9.0\times10^{-4}$, and the corresponding variance of the reconciled tank level evaluated was $\hat{\sigma}_{H_t}^2 = 9.72\times10^{-4}$. Given $\sigma_{H_t}^2 = 1.0\times10^{-2}$, the AANN performed very well in estimating the true values of the tank level. Using the optimal value of σ_{Model}^2 to train the network, results of the raw, reconciled and true values of the tank level for network training and validation for the last 200 data sets are presented in Figure 7.

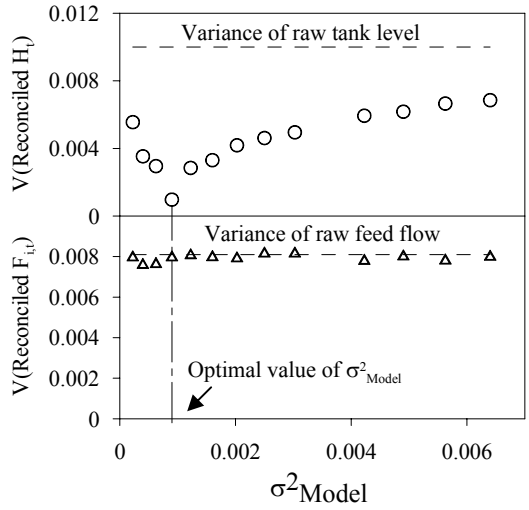

Fig. 6. Variance of reconciled tank level and feed flow as a function of σ_{Model}^2.

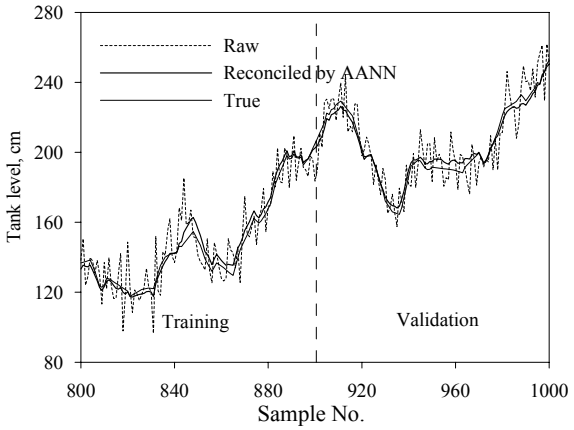

Fig. 7. Raw, reconciled and true values for tank level in network training and validation.

The AANN was then embedded inside the feedback loop before the controller calculates the control moves. Closed-loop performance of the AANN was examined for controller setpoint changes. Results of raw, reconciled and true values for the controlled variables as well as the control moves with and without the AANN, are presented in Figure 8. The reconciled values for the tank level were less noisy than the raw measurements and close to their true values. In addition, the saturated high-frequency oscillations of the control moves were significantly

reduced with the embedded AANN filter. It is worth noting that the reconciled tank level displayed some degree of deviations from the true values after the change in setpoint. This is due to the saturation in manipulated variables, which resulted in less accurate predictions of the AANN.

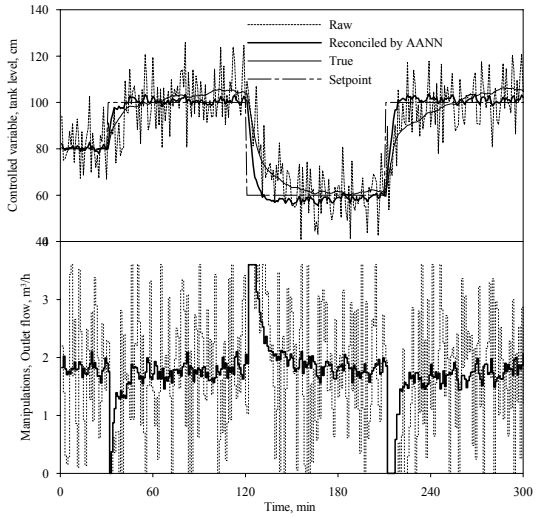

Fig. 8. Closed-loop performance of AANN for setpoint changes. Dashed line in manipulated variable represents control moves without AANN.

3.2 Spherical Storage Tank Process

The spherical storage tank process, studied in this work, is shown in Figure 9. The radius of the tank was R = 5.0 m. The nominal feed flow to the tank was 60.0 m^3/h. A PI controller was used to control the tank level by manipulating the outlet flow rate. The sampling time was 1 min. The nominal measurement noise level for the feed flow rate was σ_{F_i} =0.5 m^3/h, and σ_H =0.1 m for the liquid level. The process model for the spherical tank is

$$(2\pi RH - \pi H^2)\frac{dH}{dt} = F_i - F_o \qquad (5)$$

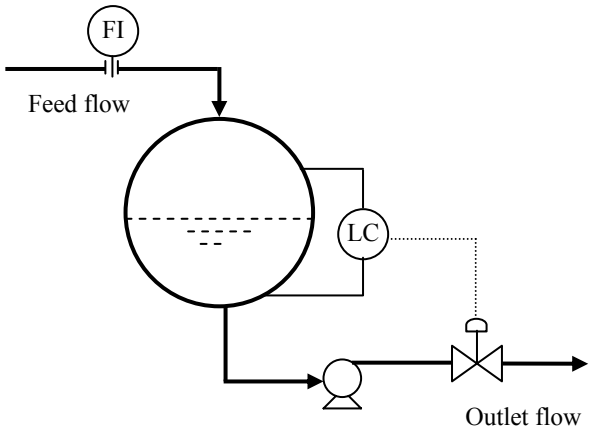

Fig. 9. Spherical storage tank process.

The tank, in open-loop, was disturbed by a series of random step changes in the feed flow rate as well as in the outlet flow rate. An AANN having the structure [5, 14, 7, 14, 2] was trained by minimizing the objective function

$$J = \sum_{p=1}^{N}\left\{\frac{1}{\sigma_{F_i}^2}\left(F_{i,t-1} - \hat{F}_{i,t-1}\right)^2 + \frac{1}{\sigma_H^2}\left(H_t - \hat{H}_t\right)^2 + \frac{1}{\sigma_{Model}^2}\left[\left(2\pi R\hat{H}_{t-1} - \pi\hat{H}_{t-1}^2\right)\left(\hat{H}_t - \hat{H}_{t-1}\right) - \Delta t\left(\hat{F}_{i,t-1} - F_{o,t-1}\right)\right]^2\right\} \qquad (6)$$

where the disretized model of Equation (5) was incorporated into the objective function to train the network. The "best" value of σ_{Model}^2 used to train the network was found to be 0.2, and after successfully training it resulted in the variance of the reconciled liquid level for the validation data sets was $\hat{\sigma}_{H_t}^2 = 2.83\times10^{-4}$ and the variance of reconciled feed flow was $\hat{\sigma}_{\hat{F}_{i,t-1}}^2 = 0.26$. Compared to $\sigma_{H_t}^2 = 1.0\times10^{-2}$ and $\sigma_{F_{i,t-1}}^2 = 0.25$, the network filtered almost completely the noise for the controlled variable, but for the disturbance variable, the variance of the reconciled values was not affected. The raw measurements, reconciled and true values for the tank level for network training and validation for the last 400 data sets are presented in Figure 10. It shows, the noise in the controlled variable was significantly reduced and the reconciled values by the AANN performed well in tracking their true values.

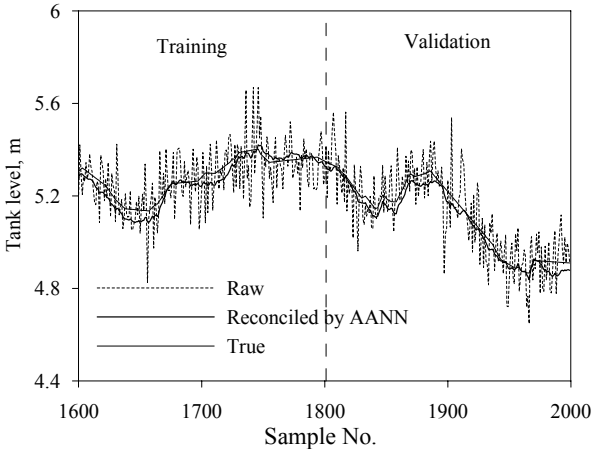

Fig. 10. Raw, reconciled and true values of spherical tank level for network training and validation.

The closed-loop performance of the AANN for the spherical tank was examined for step changes in the feed flow rate, having magnitudes of -33% and 66% of steady state value, and for control setpoint changes. Results of raw, reconciled and true values for the controlled and manipulated variables for the load changes are presented in Figures 11 and results for the controller setpoint changes are presented in Figure 12. For both cases, the AANN performed very well in tracking the true values of the controlled variable even if the process had significant dynamic changes and the process is nonlinear. In addition, the high-frequency oscillations of control moves were eliminated. Due to the significant reduction of noise propagation inside the control loop, the controller was allowed to be tuned more aggressively, such that

faster dynamic response of the process was expected for load and setpoint changes. As a consequence, the performance of the controller was improved by the embedded AANN.

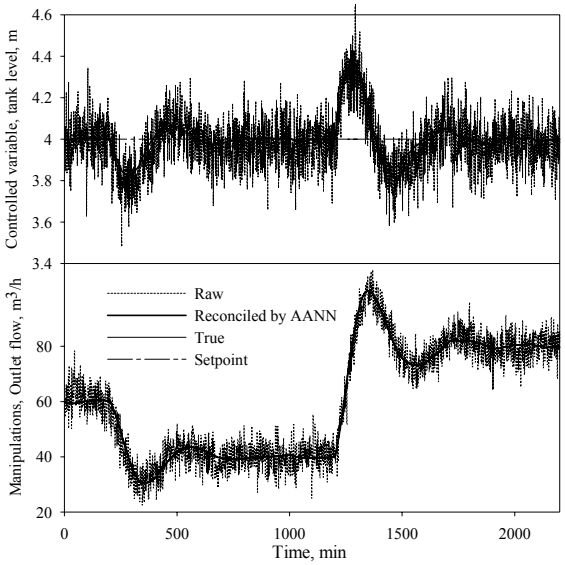

Fig. 11. Raw, reconciled and true values for the controlled and manipulated variables for the spherical tank for step changes in the feed flow rate. Dashed line in manipulated variable represents control moves without AANN.

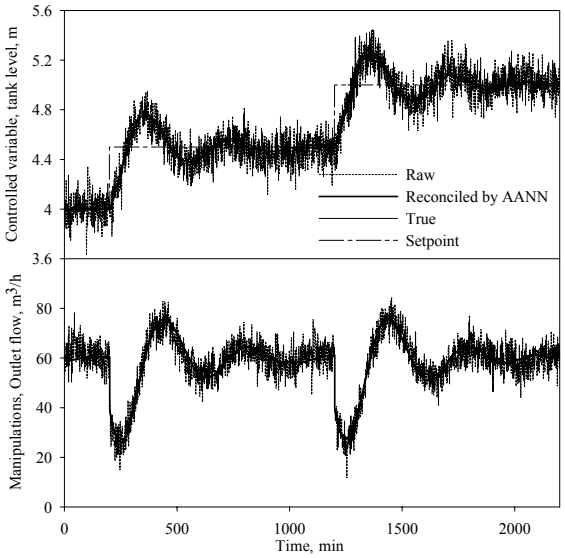

Fig. 12. Raw, filtered and true values for the controlled and manipulated variable for setpoint changes. Dashed line in manipulated variable represents control moves without AANN.

4. CONCLUSION

The dynamic AANN presented in this work has shown to be an alternative approach for real-time signal validation for dynamic processes. Because it is trained by incorporating intrinsic process models, the performance of the dynamic AANN is more effective for estimation of the current state of the process. Although implementation of AANN filters for complex processes (e.g., multivariable systems) and comparisons for the AANN to DDR using linear models are required for further demonstration, we believe that the use of this technique can greatly enhance plant monitoring as well as controller performance.

REFERENCES

Abu-el-zeet, Z., P.D. Roberts and V.M. Becerra (2002). Enhancing model predictive control using dynamic data reconciliation. *AIChE*, 48, 324-333.

Albuquerque, J. and L. Biegler (1996). Data reconciliation and gross error detection for dynamic systems. *AIChE*, 42, 2841-2856.

Bagajewicz, M. and Q. Jiang (1997). Integral approach to plant linear dynamic reconciliation. *AIChE*, 43, 2546-2558.

Bai, S., J. Thibault and D.D. McLean (2004). Closed-loop data reconciliation for the control of a binary distillation column, *Chem. Eng. Comm*., in print.

Crowe, C.M., Y.A.G Campos and A. Hrymak (1983). Reconciliation of process flow rates by matrix projection. Part I: Linear case. *AIChE*, 29, 881-888.

Crowe, C.M. (1986). Reconciliation of process flow rates by matrix projection. Part II: Nonlinear case. *AIChE*, 32, 616-623.

Darouach, M. and M. Zasadzinski (1991). Data reconciliation in generalized linear dynamic systems. *AIChE*, 37, 193-201.

Du, Y., D. Hodouin and J. Thibault (1997). Use of novel autoassociative neural network for nonlinear steady-state data reconciliation, *AIChE*, 43, 1785-1796.

Kamen, E.W. and Su, J.K. (1999). *Introduction to Optimal Estimation*, Springer, London.

Kramer, M.A. (1992). Autoassociative neural networks. *Computers Chem. Engng*, 16, 313-328.

Kuehn, D.R. and H. Davidson (1961). Computer control, II Mathematics of control. *Chemical Engineering Progress*, 57, 44.

Liebman, M.J., T.F. Edgar and L.S. Lasdon (1992). Efficient data reconciliation and estimation for dynamic processes using nonlinear programming techniques. *Computers Chem. Engng*, 16, 963-986.

Mah, R.S.H., G. Stanley and D. Downing (1976). Reconciliation and rectification of process flow and inventory data. *Ind. Eng. Chem. Proc. Des. Dev.*, 15, 175-183.

Ramamurthi, Y., P.B. Sistu and B.W. Bequette (1993). Control relevant dynamic data reconciliation and parameter estimation. *Computers Chem. Engng*, 17, 41-59.

Soderstrom, T.A., T.F. Edgar, L.P. Russo and R.E. Young (2000). Industrial application of a large-scale dynamic data reconciliation strategy. *Ind. Eng. Chem. Res*. 39, 1683-1693.

Wilson, D.I., M. Agarwal and D.W.T. Rippin (1998). Experience implementing the extended Kalman filter on an industrial batch reactor. *Computers Chem. Engng*, 22, 1653-1672.