An Immune Network Approach for Sensor-Based Diagnosis with Measurement Noise and Gross Errors *

Yoshiteru Ishida † and Yasunori Tokimasa ‡

Key Words - Failure diagnosis, sensor faults, measurement noise, immune nets, data reconciliation, self-organization, simulation

Abstract

We developed a data evaluation technique that allows to evaluate dynamically a kind of reliability of data from sensors by the redundancy among process variables. In this paper, we extend the data evaluation so that it can diagnose not only sensor faults but also process faults, taking reliability of the constraints (such as balance equations) into account. Further, the extended data evaluation is combined with data reconciliation technique to handle the situation when both measurement error and gross error may exist. On one hand, when gross error exist, data evaluation is used to filter out inadequate constraints used in data reconciliation. On the other hand, when measurement error exist, data reconciliation is used to filter out noise, hence provides data evaluation with clean relation among data. It is demonstrated by simulations that the combined method greatly enhance diagnostic performance than both methods are applied independently.

1 Introduction

On-line diagnosis based directly on the sensor data has been a challenging problem, since the data from sensors are noisy and dynamic. Further, some data from a sensor conflicts with other data due to measurement error and process/sensor faults. *Data reconciliation* has been extensively studied for estimating the state of process filtering noisy data by the state estimate methods [1, 2]. However, when there are gross errors such as process/sensor faults, the techniques does not apply, since the constraints used for estimation changes due to these gross errors. Some researches have been made for modifying the data reconciliation so that it will work even when gross errors may exist [1, 3, 4].

We have developed the sensor network independently [5, 6] based on what may be called *data evaluation*. We rather focused on detection and identification of the gross errors(sensor faults process faults [6, 7]), but have not considered the measurement noise as done in data reconciliation. In this paper, we extend the sensor network so that it will work with measurement error by incorporating the data reconciliation.

The motivation of developing sensor network is not only practical reason of using redundant information in sensors of instrumentation system, but a little more ambitious one; we

*This work has been supported in part by Mazda Foundation's Research Grant.

†Graduate School of Information Science, Nara Institute of Science and Technology, 8916-5 Takayama, Ikoma, Nara, 630-01 Japan; email: ishida@is.aist-nara.ac.jp

‡Hiroshima Research and Development Center, Technical Headquarters, Mitubishi Heavy Industry, LTD

try to develop a biological instrumentation system that can self-eliminate faulty sensory information and filter out measurement errors, inspired by immune systems. As a result, it is possible to distribute the knowledge of normal functioning process into all the sensors and the relation among sensors. Thus, the sensor network is robust in the sense that it would work even with faulty sensors. It is also flexible in the sense that it can represent knowledge of relative relation among sensor values as opposed to knowledge of reference values specified by permissible maximum and minimum.

Another biological approach which has extensively studied is using artificial neural networks. After the recent development of neural networks [8, 9], several models of the neural net have been applied to the process diagnosis as compared in the literatures [12, 10]. As often pointed out [11, 13], the important problems in applying the neural net for process diagnosis is that it requires extensive learning and that the generalization for large-scale diagnosis is difficult, hence most applications are based on a central processing diagnosis as opposed to a distributed diagnosis. Since the target systems are large-scale and dynamical systems, the diagnostic systems require not only an intelligence but a modularity [11] that allows scaling up. As the diagnostic systems becomes more intelligent and complicated, reliability consideration on the diagnostic systems become more important. Although the importance of fault-tolerance consideration is often pointed out for sensor network[14], relatively few researches have been done for diagnosis and fault tolerance of process diagnostic systems (most works focus on learning capability, stressing that diagnostic knowledge can be learned and modified by itself). The immunological approach discussed in this paper has been studied in this line.

The main contribution of this paper is the extention of our sensor network approach for process diagnosis in the following two points; (1) The process and sensor diagnosis by the sensor network is modified to work under the existence of measurement noise by incorporating data reconciliation. (2) In this combined method, it is critical to properly determine the parameter of sensitivity (permissible constraint residual, as explained sections 4 and 5). We have set up a standard for that value by simulations.

Section 2 discusses basic idea extracted from immune network. Section 3 briefly reviews our sensor network approach. Section 4 and 5 present the main result; In section 4 sensor network is extended so that it can detect not only sensor fault but process fault. Section 5 further extends the sensor network so that it will work in the noisy environment by incorporating data reconciliation technique.

2 Immune Networks and Diagnosis

In this paper, we do not explore much about how our data evaluation models correspond to the information processing discussed in the theory of immune network. We very briefly review the information processing features of our approach that may be shared with that of theory of immune network.

After the proposal of immune network (idiotypic network) theory by Jerne [15], systems approach have been made extensively to characterize immune response by dynamical models [16, 17, 18, 19, 20]. These dynamical models essentially describe population dynamics of antigen, antibody, and specific immune related cells in a similar manner to the models

describing population dynamics of species in mathematical ecology.

While we use dynamical models, our approach is different from the systems approach mentioned above. We do not intend to elaborate the dynamics of immune response. Rather we try to use the informational features in the theory of immune network as summarized below.

- Recognition of non-self (faults in a plant in our application) is done by distributed units which dynamically interacts with each other in parallel.
- Distributed units reflect redundancy of information carried by the unit (Sensor values reflecting the redundant process values in our application).
- Each unit reacts based only on its own knowledge (i.e. it refers to its own state to judge its consistency with others), but this local information processing connected by the evaluation chain leads to emergent behavior: identification of the identity.
- Memory is realized as stable equilibrium points of the dynamical network. Recognition of the network is done by changing the state of the network from one stable equilibrium to another one by disturbances on the network.

In this paper, we implemented these features in our application of sensor network. However, we do not pursue the parallelism between our model and immune systems any more.

2.1 Data evaluation by dynamic network

This subsection briefly reviews the diagnostic model by dynamical evaluation chain of identity. Before presenting the formal model, we explain how the chain of evaluating identity can be constructed by dynamical systems with an illustrative example.

Fig. 1 shows the evaluation chain of three units. Each unit tries to identify the identity (self/non-self for immune systems, normal/faulty for diagnosis) of other units. Variable (r_i and its normalization R_i) indicating the identity of units are assigned to each unit. Dynamical system is constructed by associating the time derivative of the variable with those of other units connected by the evaluation chain.

$r_i \in (-\infty, \infty)$ may be normalized to $R_i \in [0, 1]$ by the equation $R_i(t) = \frac{1}{1+\exp(-r_i(t))}$. We can use whatever mapping depending on the application in mind. Throughout this paper, we used the mapping of the above equation. The continuous value R_i between 0 and 1, indicating non-self(faulty) and self(normal) of the unit respectively, is assigned to each unit. We call the value R_i calculated by the diagnostic models the reliability measure, to distinguish it from probabilistic concept of reliability.

A possible association corresponding to the evaluation chain would be the following dynamical system:

$$\begin{aligned} dr_1(t)/dt &= R_3 - R_2 \\ dr_2(t)/dt &= -R_1 - R_3 \\ dr_3(t)/dt &= -R_2 + R_1 - 1 \end{aligned}$$

As seen in the first equation, the evaluation of the unit 1, for example, depends upon two variables: R_2 and R_3. Since the unit 1 is evaluated positively by unit 3, the evaluation of unit 1 will be high if the evaluation of the unit 3 is high. On the other hand, since the unit 1 evaluates the unit 2 negatively, the evaluation of unit 1 will be low if the unit 2 is actually evaluated high.

Formally stated, if there are units $U = \{u_i\} i = 1..n$, we first assign the value between 0 and 1 (the initial value of the reliability measure) to each unit. Then we calculate the reliability measures R_i by the following dynamical model.

e_{ij} indicates the test on the unit u_j done by the unit u_i. Thus, the test outcome T_{ij} of the test e_{ij} is defined:

$$T_{ij} = \begin{cases} -1 & \text{if a testing unit } u_i \text{ is normal and a tested unit } u_j \text{ is faulty.} \\ 1 & \text{if both units } u_i \text{ and } u_j \text{ are normal.} \\ -1/1 & \text{if a testing unit } u_i \text{ is faulty.} \\ 0 & \text{if there is no test from unit } u_i \text{ to unit } u_j. \end{cases}$$

$$dr_i(t)/dt = \sum_{e_{ji} \in T^i} T_{ji} R_j + \sum_{e_{ij} \in T_i} T_{ij} R_j - \frac{1}{2} \sum_{e_{ij} \in T_i} (T_{ij} + 1) \tag{1}$$

where T_i is a set of all the tests done by the unit u_i, and T^i is a set of all the tests on the unit u_i.

The right hand side of the equation 1 will be positive (negative) if $R_i = 0(1)$ makes inconsistency with the data T_{ij}, T_{ij} and R_j of the adjacent units. The dynamical model will change to the direction of canceling the inconsistency, i.e. if The right hand side is positive (negative), then r_i hence R_i will increase.

This diagnostic model determines the reliability measure R_i of the unit by summing up the weighted opinions of connected units. By weighting R_i on each opinion, the opinion with higher reliability affects more. It should be noted that the reliability measure of one unit is evaluated not only from the opinions of other units testing the unit, but from the opinions of what the unit said to the other tested units. The former corresponds to the first term of the model 1 and the latter to those of the second and third term.

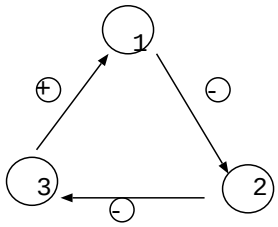

Figure 1: An example of evaluation chain

This model 1 is known to converge by the following Liapunov function [21] in the same manner as that of Hopfield Net Algorithm.

So far, we have assumed that the test done by normal unit is always reliable. That is, if $R_i = 1$ then $T_{ij} = 1$ implies that $R_j = 1$ and that $T_{ij} = -1$ implies that $R_j = -1$. In

existing systems, however, this is not always the case. As will be seen in the sensor diagnosis application, test may not be reliable even if it is done by normal unit due to the sensitivity of test. Formally said, even if $R_i = 1$ and $T_{ij} = 1$, it may be the case that $R_j = 0$ because of the incompleteness of test. The test outcome, then, is defined as:

$$T_{ij} = \begin{cases} 1 & \text{if both a testing unit } u_i \text{ and a tested unit } u_j \text{ is normal.} \\ -1/1 & \text{if either a testing unit } u_i \text{ or a tested unit } u_j \text{ is faulty.} \\ 0 & \text{if there is no test from unit } u_i \text{ to unit } u_j. \end{cases}$$

In that case, the the above diagnostic model should be modified to:

$$\begin{aligned} dr_i(t)/dt &= \sum_{e_{ji} \in T^i} (T_{ji} - 1) R_j(t) + \sum_{e_{ij} \in T_i} (T_{ij} - 1) R_j(t) - r_i(t) \\ &= \sum_j T^+_{ji} R_j(t) - r_i(t) \end{aligned} \tag{2}$$

where

$$T^+_{ij} = \begin{cases} T_{ij} + T_{ji} - 2 & \text{if both tests } e_{ij} \text{ and } e_{ji} \text{ exist.} \\ T_{ij} + T_{ji} - 1 & \text{if one of } e_{ij} \text{ or } e_{ji} \text{ exists.} \\ 0 & \text{if neither } e_{ij} \text{ nor } e_{ji} \text{ exists.} \end{cases}$$

The model 1 is designed to give clear diagnosis, faulty or normal not in ambiguous state between them. That is, the reliability measure hardly stay around the intermediate values near 0.5. In some situations, however, the information of ambiguous state is also necessary rather than making them black or white. The above system keeps the information of ambiguous state of sensor reliability measure. The term $-r_i(t)$ is inhibiting term that is added to keep the ambiguous state, when positive opinions on unit i and negative opinions balanced. In this system, there are equilibrium points satisfying that $r_i(t) = \sum_j T^+_{ji} R_j(t)$. Thus, R_i monotonically reflects the value of $\sum_j T^+_{ji} R_j(t)$. If $\sum_j T^+_{ji} R_j(t)$ is close to 0, then R_i is close to 0.5.

3 Sensor Network

In this section, we describe the application to the sensor network for process In process diagnosis, it is often the case that measurements such as temperature, pressure, flows, which are independently measured have relation with each other. In other words, some of these measurements are redundant. Using the dependencies, many tests among sensors can be constructed. In the sensor network proposed in this section, each sensors test other sensors using these relation, rather than just measuring the process values.

We have carried out simulations as well as investigation using the data from existing plant such as a firing section of a cement plant (about twenty thermometers) [7] and a blast furnace of a steal plant (about five hundred thermometers). In this section, however, we briefly review how the sensor network is constructed based on the dynamical systems mentioned above by an illustrative example.

3.1 Sensor network for sensor fault diagnosis

In this subsection, we briefly review the sensor network for sensor fault diagnosis. Constraints among process variables may be expressed by:

$$f_k(y_i^*, y_j^*) = 0, i \neq j$$

where y_i^* denotes the true value of the process variable measured by the sensor s_i.

We have discussed that the constraint above can be regarded as s_i and s_j testing each other.

$$T_{ij} \equiv \begin{cases} +1 & \text{if } |f_k(y_i(t), y_j(t))| \leq \epsilon_k \\ -1 & \text{if } |f_k(y_i(t), y_j(t))| > \epsilon_k \\ 0 & \text{if there is no constraint between } y_i^*(t) \text{ and } y_j^*(t). \end{cases} \quad (3)$$

where $y_i(t)$ denotes the measured value of sensor s_i. Measured value and true value of the process variable can be related by:

$$y_i(t) = y_i^*(t) + e_i(t), \quad i = 1, \cdots, n$$

where $e_i(t)$ denotes the measurement error. It should be noted that the relation between $y_i(t)$ and $y_j(t)$ is symmetrized in this new definition of T_{ij}. (The definition of T_{ij} in the original data evaluation model is not symmetrical.) This because when there is constraint violation between $y_i(t)$ and $y_j(t)$, the evaluation cannot be determined in one direction in the sensor network application.

$r_k \equiv f_k(\mathbf{y}(t))$ is constraint residuals, and ϵ_k is permissible constraint residual. This permissible constraint residual must be determined by taking the effect of random noise at the sensor into account. If this permissible constraint residual is too small, T_{ij} will be affected sensitively by random noise, while it will be insensible even to the gross error if the permissible constraint residual is too big.

We will describe how the permissible constraint residual can be determined below. Suppose the constraint is linear;

$$f_k(y_i^*, y_j^*) \equiv a_{ki} y_i^* + a_{kj} y_j^* - b_k = 0$$

And also suppose there is no gross error; $e_i \equiv e_{r_i} \sim N(0, \sigma_i^2)$ and $e_j \equiv e_{r_j} \sim N(0, \sigma_j^2)$ where $N(0, \sigma_i^2)$ denotes a Gaussian distribution with zero mean and a known variance.

Then the constraint residual is;

$$\begin{aligned} r_k &\equiv f_k(y_i(t), y_j(t)) \\ &= a_{ki} e_i(t) + a_{kj} e_j(t) \end{aligned}$$

Thus,

$$r_k \sim N(0, a_{ki}^2 \sigma_i^2 + a_{kj}^2 \sigma_j^2)$$

ϵ_k should be determined so that the probability of the above r_k satisfying $|r_k| \leq \epsilon_k$ is sufficiently large. Even if the constraint is non-linear equation or inequality, the optimal permissible residual can be determined if the distribution and variance of the constraint residual r_k are provided.

4 Extentions of the Sensor Network

4.1 Data and relation evaluation by dynamic testing

In this subsection, we extend the data evaluation so that it can evaluate not only data but also the relation among data, to attain process fault diagnosis as well.

As discussed in the section 3, the knowledge of normal process is embedded in the constraints among sensors. Thus, when process faults occurs, it amount to violation of the constraints. In fact, when process fault occurs, reliability measures of many sensors related to the constraints become low simultaneously. Fig. 2 illustrates the situation when process fault occurs.

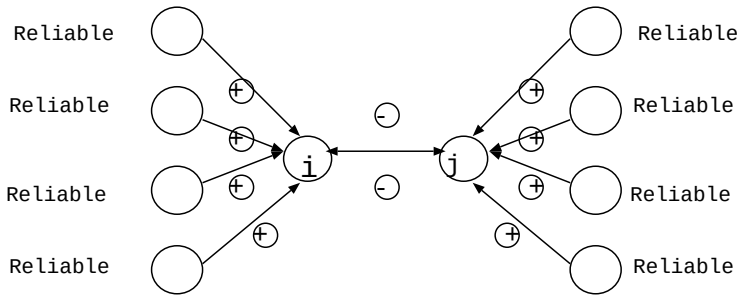

Figure 2: Situation when process fault occurs

Therefore, one way of detecting process fault by sensor net is to introduce reliability measure for testing relation. Let $R_{T_{ji}}$ denote the reliability measure of the test T_{ji}. Then the dynamical model 2 becomes as follows:

$$dr_i(t)/dt = \sum_j T_{ji}^{+} R_j(t) R_{T_{ji}} - r_i(t) \tag{4}$$

The reliability measure $R_{T_{ji}}$ simply reflects how contradictory the testing relation is.

$$dr_{T_{ji}}(t)/dt = \begin{cases} T_{ji}^{+} R_j(t) R_i(t) - r_{T_{ji}}(t) & \text{if } T_{ji} = -1 \\ -r_{T_{ji}}(t) & \text{if } T_{ji} = 1 \end{cases} \tag{5}$$

$$R_{T_{ji}} = \begin{cases} 1 & \text{if there is no contradiction in the relation } T_{ji} \\ 0 & \text{if there is contradiction in the relation } T_{ji} \end{cases} \tag{6}$$

That is, $R_{T_{ji}} = 1$ when corresponding process is normal, and $R_{T_{ji}} = 0$ when the corresponding process is abnormal. The first term of the equation above 5 reflects the degree of contradiction in the testing relation T_{ji}. At the equilibrium point of this equation, $r_{T_{ji}}(t) = T_{ji}^{+} R_i(t) R_j(t)$ when $T_{ji} = -1$. Thus, $R_{T_{ji}}(t)$ monotonically reflects a degree of contradiction: $T_{ji}^{+} R_i(t) R_j(t)$.

The following equation continuously map from the domain of the degree of contradiction of the testing relation T_{ji} $(-\infty, 0]$ to the domain of the reliability measure $[0, 1]$.

$$R_{T_{ji}}(t) = \frac{2}{1 + \exp(-r_{T_{ji}}(t))}$$

The equation 4 is a modification naturally resulted by considering the effect of the reliability measure of the test T_{ji}. The change rate of the unit i; $dr_i(t)/dt$ should reflect all the opinions of other units weighted not only with the reliability of these other units but with those of their testing.

The equation 5 comes from the fact that the testing relation is considered to be unreliable only when $T_{ji}, R_j(t)$ and $R_i(t)$ are contradictory; $T_{ji} = -1, R_j(t) = 1$ and $R_i(t) = 1$.

4.2 Sensor net for multiple testings

Constraints among process variables may be expressed by the vector equation: $\mathbf{f}(\mathbf{y}^*, t) = 0$ where $\mathbf{y}^*$ is the true values of process variables. Let $S = \{s_1, \cdots, s_n\}$ be set of all the sensors and let S_k be the subset involved in the constraint f_k. The testing results for multiple testing can be defined as:

$$T_k \equiv \begin{cases} 1 & \text{if } |f_k(\mathbf{y}(t), t)| \leq \epsilon_k \\ -1 & \text{if } |f_k(\mathbf{y}(t), t)| > \epsilon_k \end{cases} \tag{7}$$

for the permissible constraint residual ϵ_k. This permissible constraint residual ϵ_k must be determined by taking measurement error and noise into consideration.

When the constraint of the process variables is linear;

$$f_k(\mathbf{y}^*) \equiv \sum_i a_{ki} y_i^* - b_k = 0,$$

also when there is no gross error;

$$e_i \equiv e_{r_i} \sim N(0, \sigma_i^2),$$

the constraint residual r_k will be given:

$$\begin{aligned} r_k &\equiv f_k(\mathbf{y}(t)) \\ &= \sum_i a_{ki} e_i(t). \end{aligned}$$

Thus,

$$r_k \sim N(0, \sum_i a_{k_i}^2 \sigma_i^2).$$

The value of ϵ_k should be determined so that the probability of $|r_k| \leq \epsilon_k$ becomes sufficiently large. When the constraint of the process variables is given by inequalities, the value of permissible constraint residual ϵ_k can be determined in a similar manner if the distribution and variance of the residual r_k is obtained.

S_k denotes a set of all the sensors s_i that appear their corresponding process variables y_i appear in the constraint $f_k(\mathbf{y}(t))$. When there are more than three variables in f_k, the model 4 and 5 can be modified as follows:

$$dr_i(t)/dt = \sum_{k, s_i \in S_k} \{T_k^+ R_{T_k}(t) \prod_{\substack{j, s_j \in S_k \\ j \neq i}} R_j(t)\} - r_i(t) \tag{8}$$

$$dr_{T_k}(t)/dt = \begin{cases} T_k^+ \prod_{i, s_i \in S_k} R_i(t) - r_{T_{ji}}(t) & \text{if } T_k = -1 \\ -r_{T_{ji}}(t) & \text{if } T_k = 1 \end{cases} \tag{9}$$

$$R_{T_k}(t) = \frac{2}{1 + \exp(-r_{T_k}(t))}$$

where $T_k^+ \equiv n_k(T_k - 1) + \lambda_k, 0 \leq \lambda_k \leq n_k$ and $n_k \equiv |S_k|$: the number of the sensor involved in the constraint f_k. λ_k represents a parameter for tuning relative weight between normal recognition and faulty recognition in the sensor set S_k.

Example of the sensor network with multiple testing is presented in [22].

5 Data evaluation and data reconciliation combined in the sensor network: fault detection under noise

In the instrumentation of process plants, sensor values do not necessarily correspond to the true process values. Data from sensors may conflict each other through the process constraints due to measurement errors and process faults.

Data reconciliation is a method to estimate the true values of process variables using constraints (such as balance equations) among process variables. Data reconciliation is designed to work under the situation where the process values may be noisy or under the effects of disturbances. However, data reconciliation method do not work when there are gross errors whose effects may violate the process constraints on which data reconciliation technique depends. In this section, we first discuss features of data reconciliation. Then, we will present how the data reconciliation technique can be combined with data evaluation by the sensor net.

5.1 Data reconciliation

Generally, data reconciliation problem can be formulated into the following non-linear programming problem:
(NLP1) $min_{\hat{\mathbf{y}}} \Phi(\mathbf{y}, \hat{\mathbf{y}}; \sigma)$ subject to

$$\begin{aligned} \mathbf{f}(\hat{\mathbf{y}}) &= 0 \\ \mathbf{g}(\hat{\mathbf{y}}) &\geq 0 \end{aligned}$$

where the following sum of weighted least-square will be used for the objective function.

$$\Phi(\mathbf{y}, \hat{\mathbf{y}}; \sigma) \equiv \sum_{j=0}^{k} (\mathbf{y_j} - \hat{\mathbf{y}}(t_j))^T \mathbf{Q}^{-1} (\mathbf{y_j} - \hat{\mathbf{y}}(t_j)) \tag{10}$$

Here, $\mathbf{y_j}$ and $\hat{\mathbf{y}}(t_j)$ denote the measured and estimated process values at time t_j. t_0 and t_k are the initial and current time. $\mathbf{f}$ and $\mathbf{g}$ are process constraints of equation and inequality. $\mathbf{Q}$ is a variance-covariance matrix defined by;

$$\begin{cases} Q_{ii} \equiv \sigma_i^2 \\ Q_{ij} (i \neq j) \equiv 0 \end{cases}$$

σ_i^2 is the variance of the random noise included in the measured value y_i.

If there is no gross error, i.e., when the measurement error e_i satisfies $e_i \equiv e_{r_i} \sim N(0, \sigma_i^2)$ then the solution $\hat{\mathbf{y}}$ of the NLP1 is the optimal estimates of the process variables. When there is a gross error (due to sensor fault or process fault), then the solution $\hat{\mathbf{y}}$ of the NLP1 is not the optimal estimate, since $e_i \equiv e_{r_i} + e_{g_i} \not\sim N(0, \sigma_i^2)$ and the gross error e_{g_i} of a measured value may affect other process values through the process constraints. Therefore, gross error must be filtered out before carrying out data reconciliation. There already have been several methods for dealing with gross error. We present how the sensor net will be used to handle the situation when both gross error and measurement noise exist; the sensor net is used to detect the gross errors in the sensor network.

5.2 Combination of sensor net and data reconciliation

By substituting $\mathbf{y} - \mathbf{y}^* = e$, the objective function of the NLP1 above can be rewritten as:
$\Phi(\mathbf{y}, \mathbf{y}^*; \sigma) = \mathbf{e}^T Q^{-1} \mathbf{e}$

where $\mathbf{y}^*$ is the true value of the process variables. $\Phi(\mathbf{y}, \hat{\mathbf{y}}; \sigma) \leq \Phi(\mathbf{y}, \mathbf{y}^*; \sigma)$ holds, since $\Phi(\mathbf{y}, \hat{\mathbf{y}}; \sigma) > \Phi(\mathbf{y}, \mathbf{y}^*; \sigma)$ contradicts to the fact that $\hat{\mathbf{y}}$ is the solution of NLP1.

When there is no gross error, the measurement error is given by; $e_i \equiv e_{r_i} \sim N(0, \sigma_i^2)$. Then $\Phi(\mathbf{y}, \mathbf{y}^*; \sigma) = \mathbf{e}^T Q^{-1} \mathbf{e}$ obeys Chi-square distribution with n degrees of freedom. However, when gross error exists in the measurement error, $\Phi(\mathbf{y}, \hat{\mathbf{y}}; \sigma)$ becomes greater than the case when gross error does not exist. Thus, the value of $\Phi(\mathbf{y}, \hat{\mathbf{y}}; \sigma)$ can be used to detect gross error. Some threshold φ for the value $\Phi(\mathbf{y}, \hat{\mathbf{y}}; \sigma)$ should be determined beforehand.

Before carrying out diagnosis based on the sensor net, if $\Phi(\mathbf{y}, \hat{\mathbf{y}}; \sigma) \leq \varphi$ we presume there is no gross error and use the estimate $\hat{\mathbf{y}}$ in determining testing results among sensors. When $\Phi(\mathbf{y}, \hat{\mathbf{y}}; \sigma) > \varphi$ we use the measured value $\mathbf{y}$ in determining the testing results rather than the estimate $\hat{\mathbf{y}}$ that is unreliable due to the gross error. Formally, the testing results in the sensor net will be as follows:

- $\Phi(\mathbf{y}, \hat{\mathbf{y}}; \sigma) \leq \varphi$ (No gross error)

$$T_k = \begin{cases} 1 & |f_k(\hat{\mathbf{y}}(t), t)| \leq \epsilon_k \\ -1 & |f_k(\hat{\mathbf{y}}(t), t)| > \epsilon_k \end{cases}$$

- $\Phi(\mathbf{y}, \hat{\mathbf{y}}; \sigma) > \varphi$ (With gross error)

$$T_k = \begin{cases} 1 & |f_k(\mathbf{y}(t), t)| \leq \epsilon_k \\ -1 & |f_k(\mathbf{y}(t), t)| > \epsilon_k \end{cases}$$

When gross error exists, process value will be estimated after identifying the gross error by the above sensor net.

- For this data reconciliation after the sensor fault s_i is identified by the sensor net, diagonal elements of variance-covariance matrix Q must be updated as follows:

  $Q_{ii} \equiv \sigma_i^{2\prime} (\gg \sigma_i^2)$

  Thus, making smaller weight on the process variable than before. In other words, the update above will make the effect from erroneously measured value by faulty sensor smaller.

- When the process fault corresponding to the constraint f_k is identified by the sensor net, the effect from this constraint should be removed in NLP1. Thereby, solving the modified NLP1, the estimates of process variable that are not affected by gross error can be obtained.

For the piping diagram shown in the left of Fig. 3 where only the flow rate is measured at each sensor. As shown in the right of Fig. 3, the sensor net includes multiple testing as follows:

$$
\begin{aligned}
f_0 &: y_0^* = y_1^* \\
f_1 &: y_1^* = y_2^* + y_5^* \\
f_2 &: y_2^* + y_3^* = y_4^* \\
f_3 &: y_4^* + y_5^* = y_6^* \\
f_4 &: y_6^* + y_7^* = y_8^*
\end{aligned}
$$

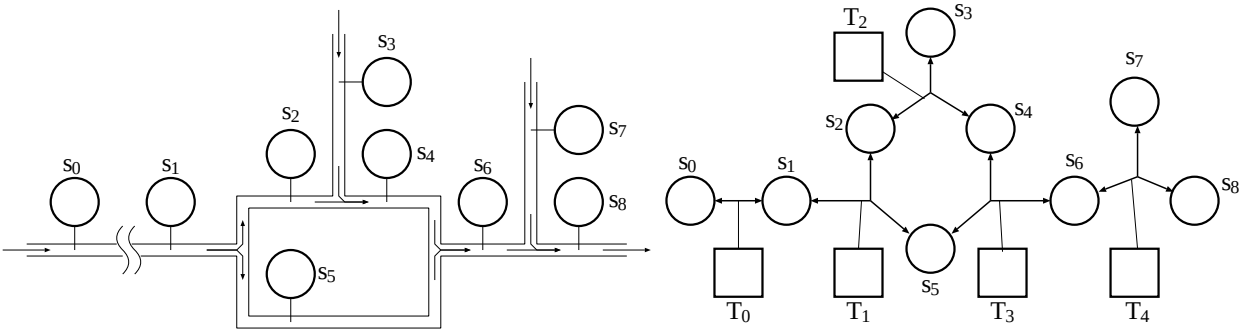

Figure 3: Piping diagram where only flow rate is measured at each sensor and its sensor net with multiple testing

Simulation for this example is carried out using the model 8 and 9 where $\lambda_k \equiv n_k$ and $\sigma_i^2 \equiv 1.0$. In this simulation, we compare two cases: (A) data evaluation and data reconciliation is performed independently, and (B) data evaluation and data reconciliation are combined in the sensor network as explained above. Further, in order to see the sensitivity of ϵ_k on the diagnosis results. Both comparisons are done for a sensor fault as well as a process fault.

- Sensor fault: The sensor s_4 failed at $t = 50$ and recovered at $t = 100$. When sensor s_4 failed, the constraints f_2, f_3 are violated. Fig. 4 shows the true values and measured values.

  - When $\epsilon_k \equiv (\sum_i a_{ki}^2 \sigma_i^2)^{1/2}$ is used: Figs. 5 show the result when the sensor net and data reconciliation are carried out independently. Figs. 6 show the result when the sensor net and data reconciliation are combined as presented in this section. Figs. 7 show estimates of s_4, s_6. Fig. 8 shows the values of objective function by data reconciliation.
  - When $\epsilon_k \equiv 2(\sum_i a_{ki}^2 \sigma_i^2)^{1/2}$ is used: Figs. 9 show the result when the sensor net and data reconciliation are carried out independently. Figs. 10 show the result when the sensor net and data reconciliation are combined as presented in this section. Figs. 11 show estimates of s_4, s_6 by data reconciliation. Fig. 12 shows the values of objective function for data reconciliation.

- Process fault: Flow lost at the point corresponding to the constraint f_3 occurred at $t = 50$ and recovered at $t = 100$. The constraint f_3 is violated in this process failure.

  Fig. 13 shows the true values and measured values.

  - When $\epsilon_k \equiv (\sum_i a_{ki}^2 \sigma_i^2)^{1/2}$ is used: Figs. 14 show the result when the sensor net and data reconciliation are carried out independently. Figs. 15 show the result when the sensor net and data reconciliation are combined as presented in this section. Figs. 16 show estimates of s_4, s_6 by data reconciliation. Fig. 17 shows the values of objective function for data reconciliation.
  - When $\epsilon_k \equiv 2(\sum_i a_{ki}^2 \sigma_i^2)^{1/2}$ is used: Figs. 18 show the result when the sensor net and data reconciliation are carried out independently. Figs. 19 show the result when the sensor net and data reconciliation are combined as presented in this section. Figs. 20 show estimates of s_4, s_6 by data reconciliation. Fig. 21 shows the values of objective function for data reconciliation.

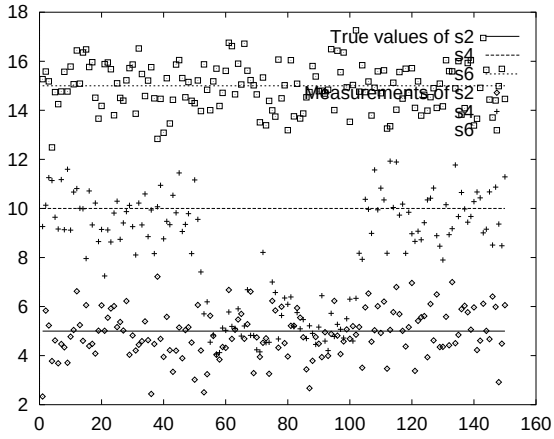

Figure 4: True values and measured values when sensor s_4 failed

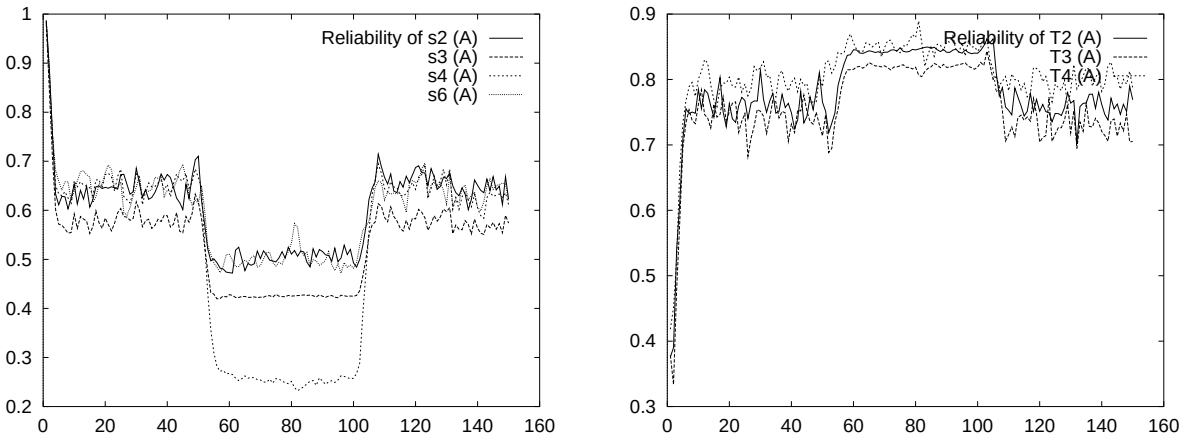

Figure 5: Time evolution of reliability measures using data reconciliation and data evaluation independently when the sensor s_4 failed at $t = 50$ and recovered at $t = 100$.

First, we compare the case when $\epsilon_k \equiv (\sum_i a_{ki}^2 \sigma_i^2)^{1/2}$ and the case when $\epsilon_k \equiv 2(\sum_i a_{ki}^2 \sigma_i^2)^{1/2}$ to investigate the effect of permissible constraint residual. For the case when $\epsilon_k \equiv (\sum_i a_{ki}^2 \sigma_i^2)^{1/2}$ of sensor fault, both independent and combined methods indicate the lowest reliability measure of the faulty sensor s_4. However, The other reliability measures of the other normal sensors are also dragged to be low, and hence it would be difficult to identify the faulty sensor. Process fault was not diagnosed at all in this case. This is because random noise affected the testing results. For the case when $\epsilon_k \equiv 2(\sum_i a_{ki}^2 \sigma_i^2)^{1/2}$, both sensor fault and process fault are diagnosed for both independent and combined method.

Secondly, if we compare the independent method and the combined method, higher reliability measures for sensors and testings are obtained for the combined method. This tendency is especially true when there is no gross error. Thus, gross error can be more clearly identified by the combined method for both sensor fault and process fault. The combined method generally gives better estimates for process values. When $\epsilon_k \equiv (\sum_i a_{ki}^2 \sigma_i^2)^{1/2}$, however, both

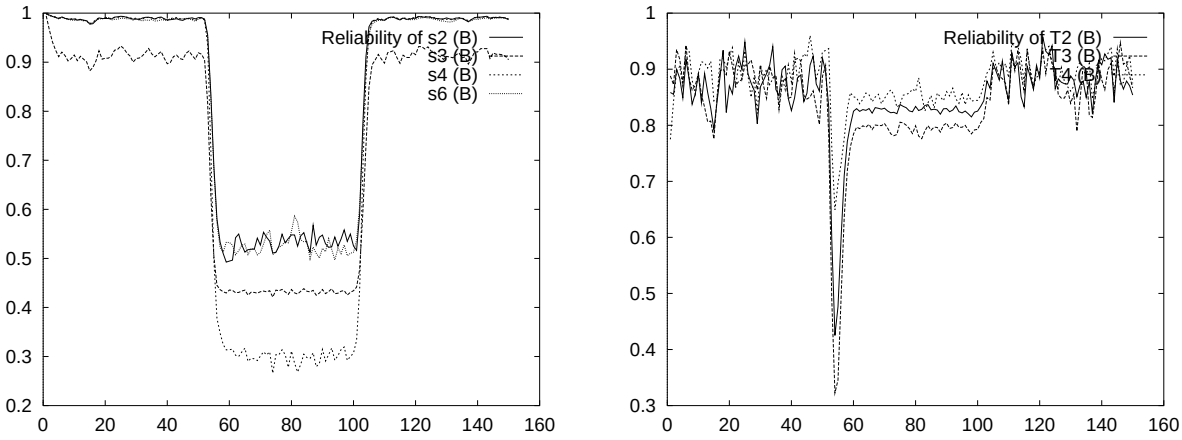

Figure 6: Time evolution of reliability measures using data reconciliation and data evaluation combinedly when the sensor s_4 failed at $t = 50$ and recovered at $t = 100$.

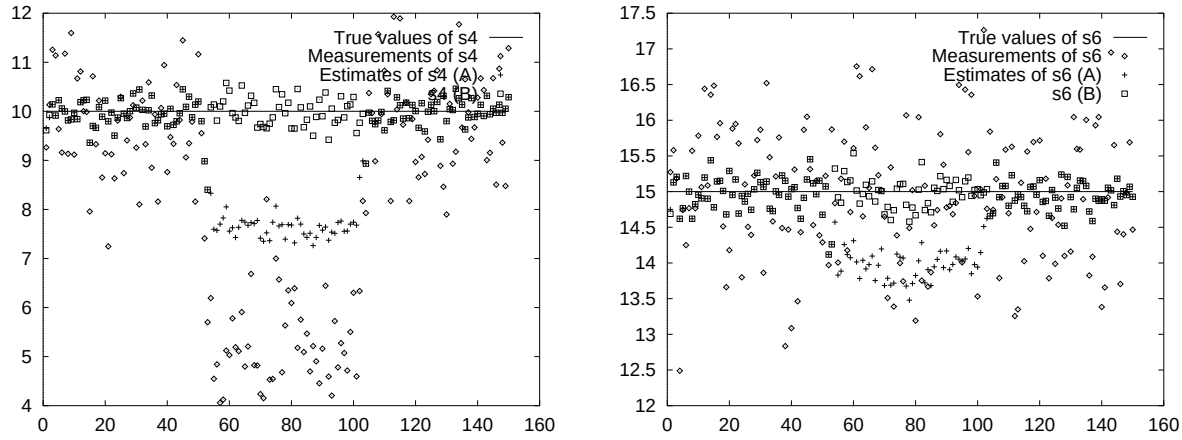

Figure 7: Estimates of s_4 and s_6 by data reconciliation when the sensor s_4 failed at $t = 50$ and recovered at $t = 100$.

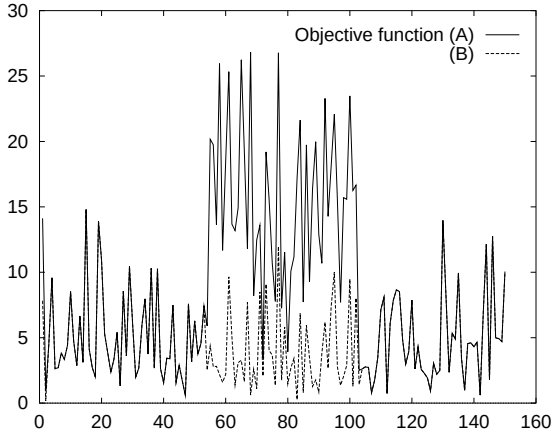

Figure 8: Values of objective function by data reconciliation when the sensor s_4 failed at $t = 50$ and recovered at $t = 100$.

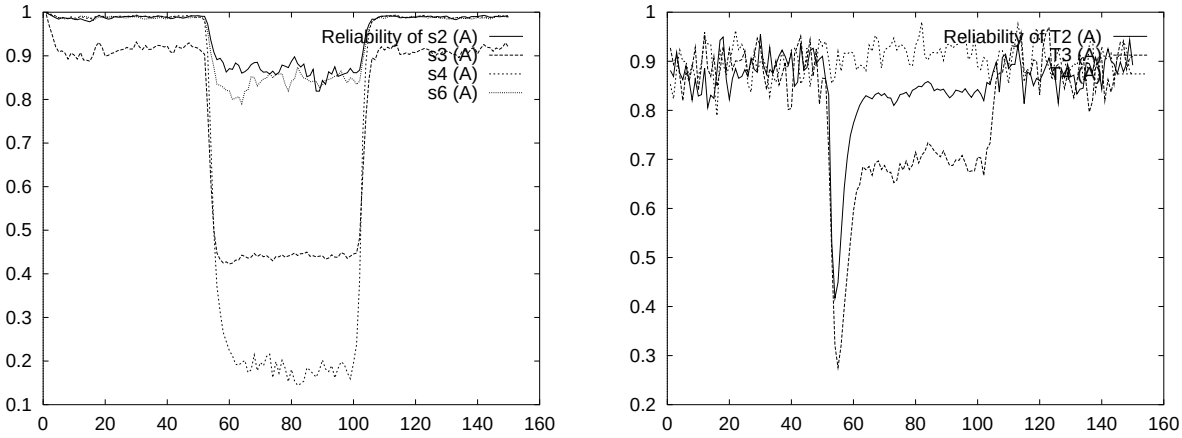

Figure 9: Time evolution of reliability measures using data reconciliation and data evaluation independently when the sensor s_4 failed at $t = 50$ and recovered at $t = 100$.

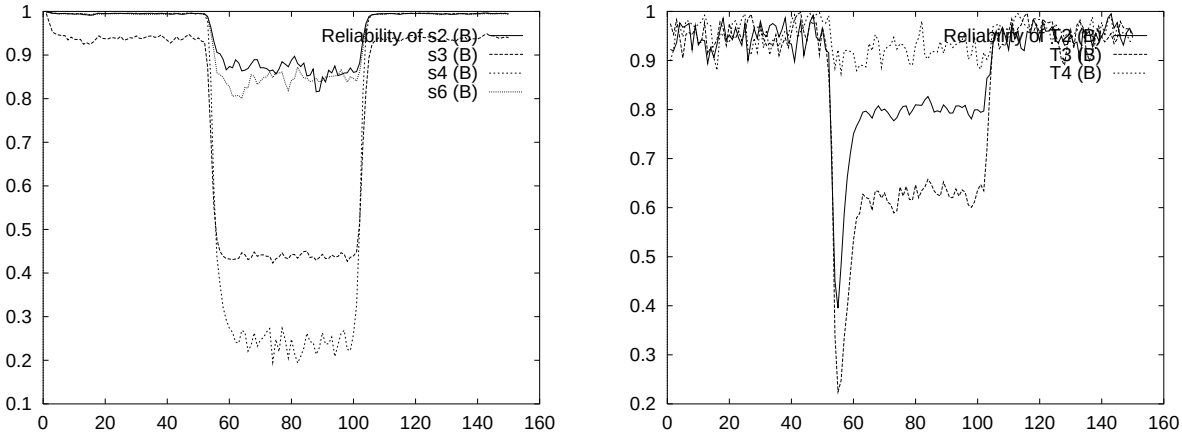

Figure 10: Time evolution of reliability measures using data reconciliation and data evaluation combinedly when the sensor s_4 failed at $t = 50$ and recovered at $t = 100$.

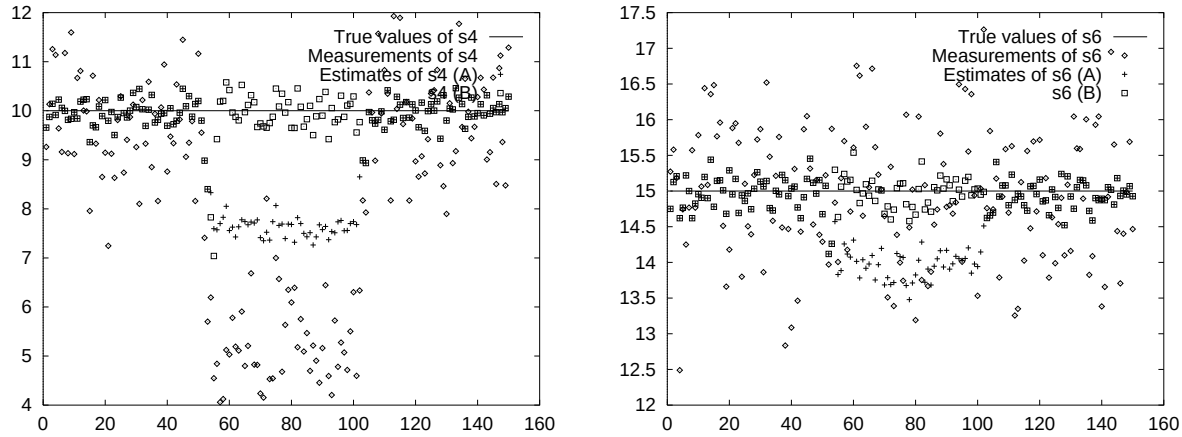

Figure 11: Estimates of s_4 and s_6 by data reconciliation when the sensor s_4 failed at $t = 50$ and recovered at $t = 100$.

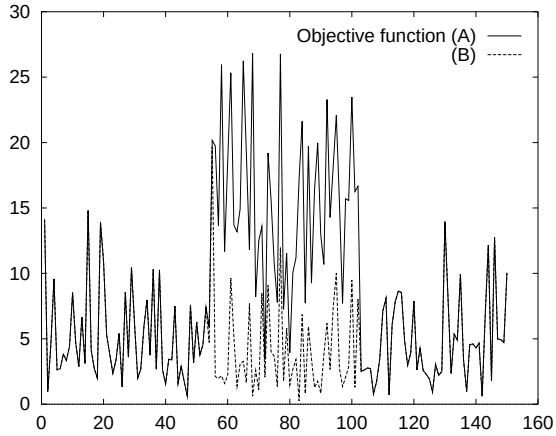

Figure 12: Values of objective function by data reconciliation when the sensor s_4 failed at $t = 50$ and recovered at $t = 100$.

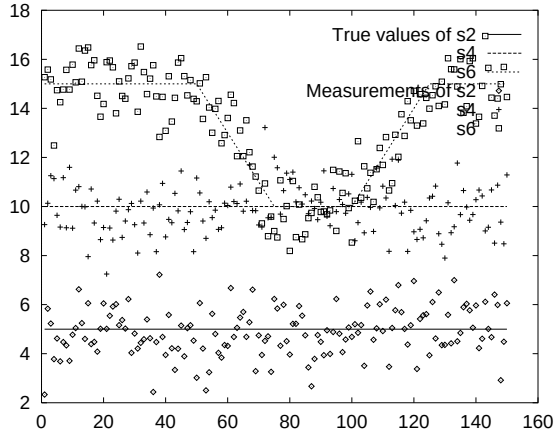

Figure 13: True values and measured values when flow lost violating f_3 occurred

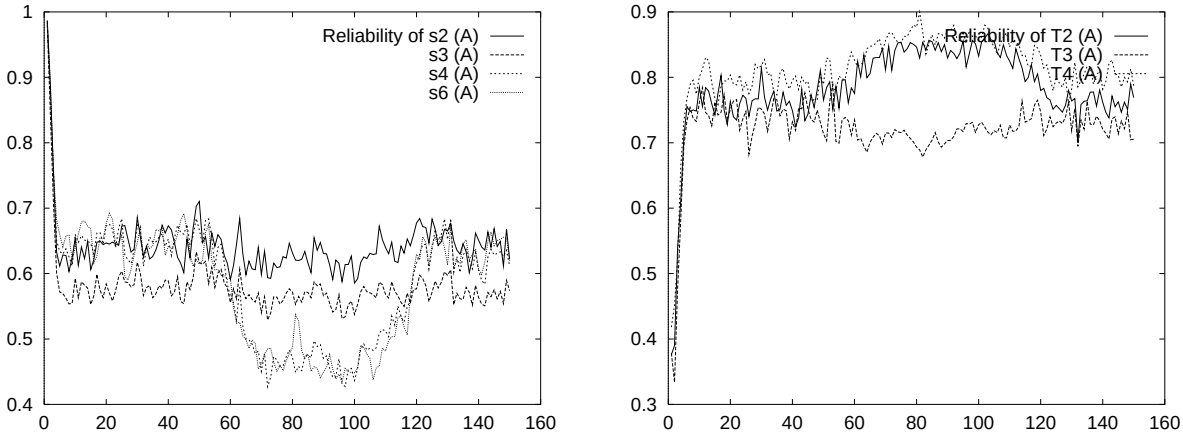

Figure 14: Time evolution of reliability measures using data reconciliation and data evaluation independently when flow lost violating f_3 occurred at $t = 50$ and recovered at $t = 100$.

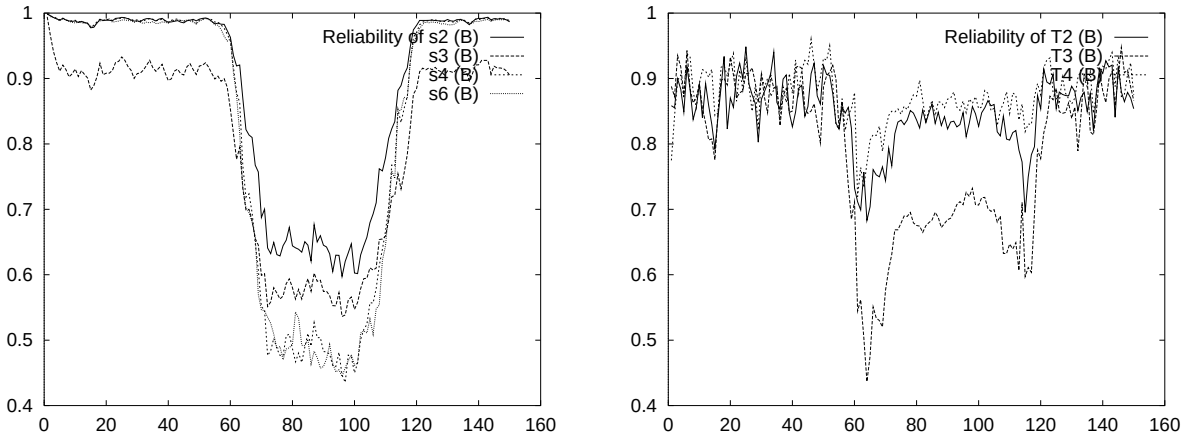

Figure 15: Time evolution of reliability measures of sensors using data reconciliation and data evaluation combinedly when flow lost violating f_3 occurred at $t = 50$ and recovered at $t = 100$.

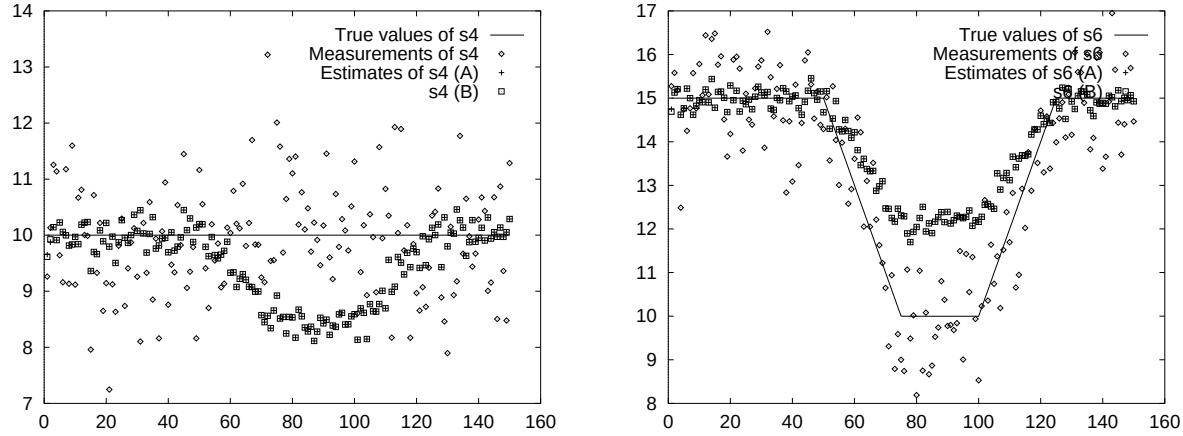

Figure 16: Estimates of s_4 and s_6 by data reconciliation when flow lost violating $f_3$occurred at $t = 50$ and recovered at $t = 100$.

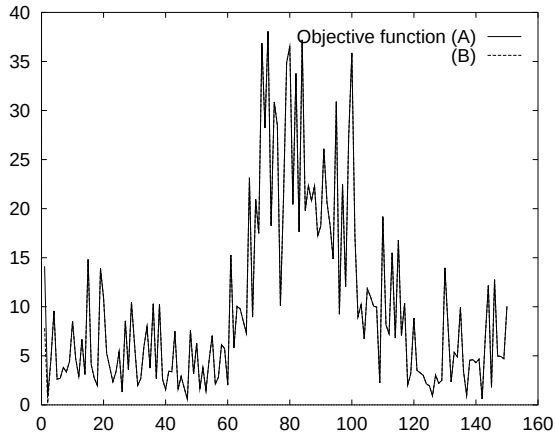

Figure 17: Values of objective function by data reconciliation when flow lost violating f_3 occurred at $t = 50$ and recovered at $t = 100$.

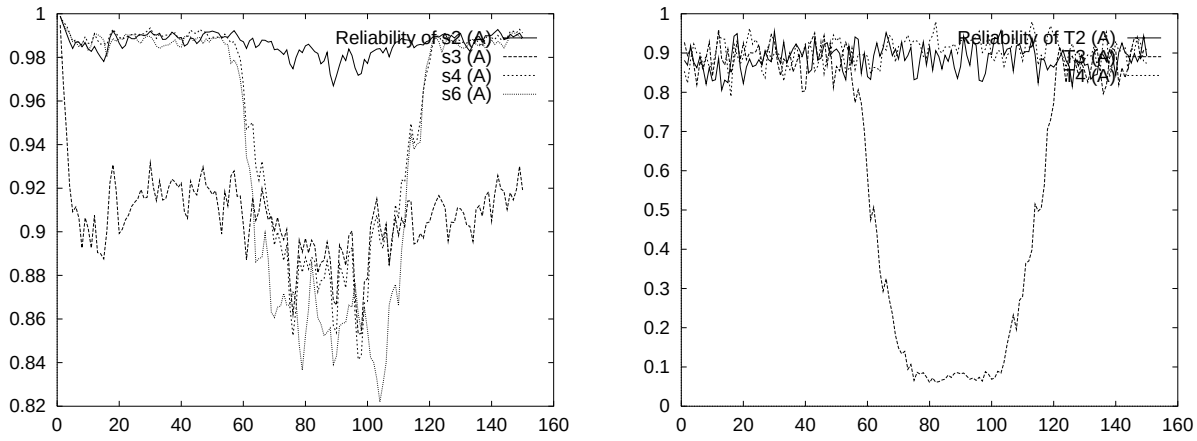

Figure 18: Time evolution of reliability measures using data reconciliation and data evaluation independently when flow lost violating f_3 occurred at $t = 50$ and recovered at $t = 100$.

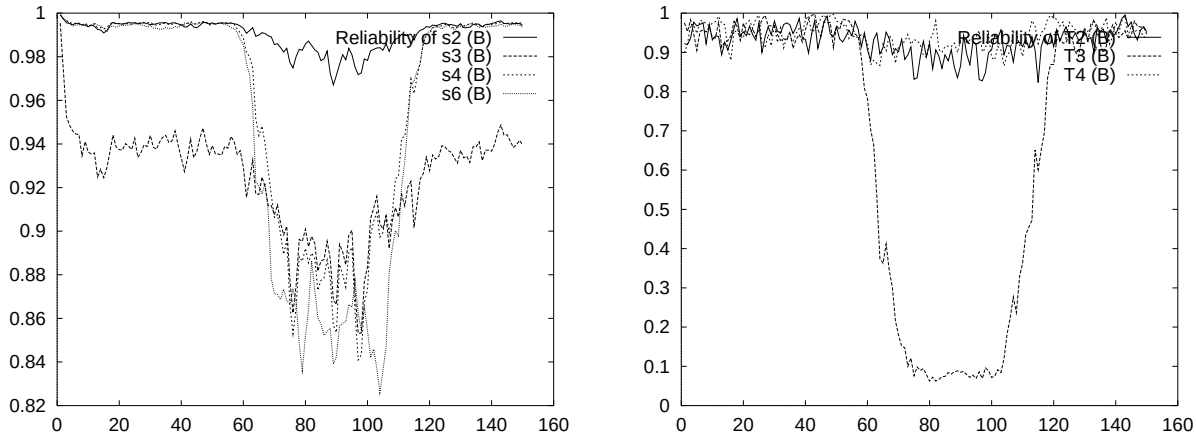

Figure 19: Time evolution of reliability measures using data reconciliation and data evaluation combinedly when flow lost violating f_3 occurred at $t = 50$ and recovered at $t = 100$.

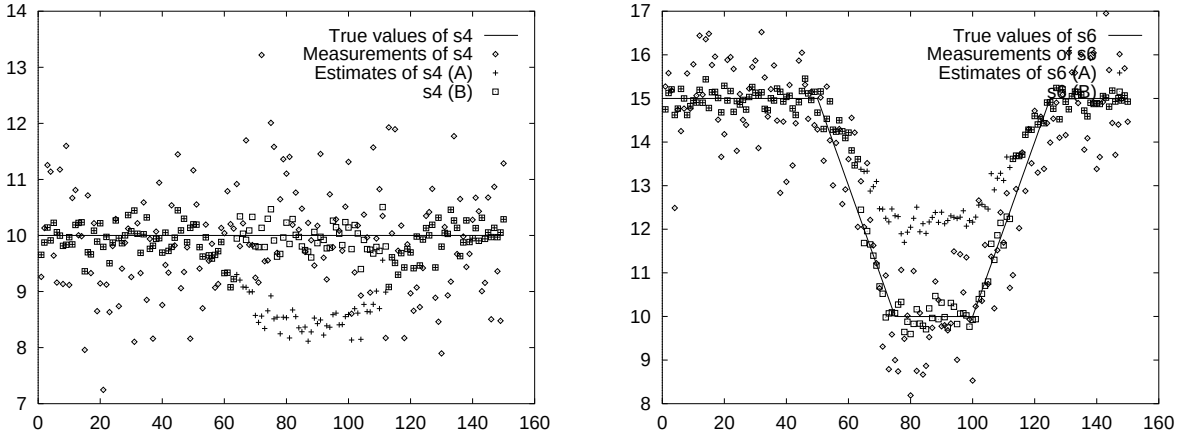

Figure 20: Estimates of s_4 and s_6 by data reconciliation when flow lost violating f_3 occurred at $t = 50$ and recovered at $t = 100$.

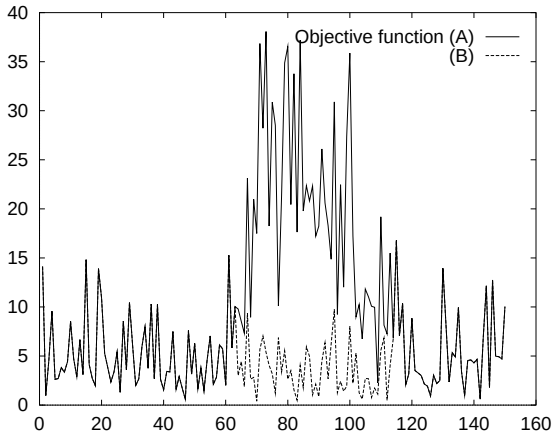

Figure 21: Values of objective function by data reconciliation when flow lost violating f_3 occurred at $t = 50$ and recovered at $t = 100$.

independent and combined method failed to give good estimate of process values. This is because the gross error could not be identified due to too small value of ϵ_k.

As a summary of the results, the combined method works well. However, proper determination of the permissible constraint residual ϵ_k would be critical.

6 Conclusion

Redundancy among process variables have been used to filter measurement noise in data reconciliation. We have shown that the redundancy can filter not only the measurement noise (as in data reconciliation) but also false constraints due to sensor/process fault (by the data evaluation). Further, we have shown that both data evaluation and data reconciliation can be effectively combined, filling up weak points with each other; Data reconciliation is used to filter out the measurement noise to obtain the clear testings used for data evaluation. Data evaluation is used to filter out gross errors to obtain clean estimates in data reconciliation. Simulation results for illustrative example demonstrated that the combined method has shown better diagnostic performance than that when both method are applied independently. Simulations have also shown that determination the permissible constraint residual ϵ_k would be critical. We have suggested some criterion for the permissible constraint residual relative to the variance of Gaussian distribution of the measurement noise.

References

[1] B. Toja and L.T. Biegler , "Simultaneous strategies for data reconciliation and gross error detection of nonlinear systems", *Computers chem. Engng*, Vol. 15, No. 10, pp. 679-690, 1991.

[2] M.J. Liebman, T.F. Edgar, and L.S. Lasdon, "Efficient Data Reconciliation and Estimation for Dynamic Process Using Nonlinear Programming Techniques", *Computers chem. Engng*, Vol. 16, No. 10/11, pp. 963-986, 1992.

[3] S. Narasimhan and P. Harikumar , "A Method to Incorporate Bounds in Data Reconciliation and Gross Error Detection I: The Bounded Data Reconciliation Problem", *Computers chem. Engng*, Vol. 17, No. 11, pp. 1115-1120, 1993.

[4] S. Narasimhan and P. Harikumar , "A Method to Inforporate Bounds in Data Reconciliation and Gross Error Detection II: Gross Error Detection Strategies", *Computers chem. Engng*, Vol. 17, No. 11, pp. 1121-1128, 1993.

[5] Y. Ishida, " Fully Distributed Diagnosis by PDP Learning Algorithm: Towards Immune Network PDP Model", *Proc. International Joint Conference on Neural Networks*, San Diego, pp.777-782, 1990.

[6] Y. Ishida and F. Mizessyn, "Learning Algorithms on Immune Network Model: Application to Sensor Diagnosis" *Proc. International Joint Conference on Neural Networks*, pp. 33-38, 1992.

[7] M. Francois and Y. Ishida, Immune Networks for Cement Plants, *Proc. of Intenational Symposium on Autonomous Decentralized Systems*, pp.282-288 , 1993.

[8] D. E. Rummelhart, and J. L. McClelland, *Parallel Distributed Processing*, MIT Press, 1986.

[9] J.J. Hopfield, "Neural network and physical systems with emergent collective computational abilities," *Proc. Natl. Acad. Sci.* USA, vol. 81, April 1982, pp. 3088-3092.

[10] T. Sorsa and H.N. Koivo, " Application of artificial neural networks in process fault diagnosis," *Automatica*, vol. 29, pp.843-849, 1993.

[11] J.A. Leonard and M.A. Kramer, "Diagnosing dynamic faults using modular neural nets," *IEEE Trans. Expert*, vol. 8, pp.44-53,1993.

[12] P.A. Jokinen, "Comparison of neural network models for process fault detection and diagnosis problems," *Proc. International Joint Conference on Neural Networks*, Seattle, pp.239-244, 1991.

[13] T. Sorsa, H.N. Koivo and H. Koivisto, " Neural networks in process fault diagnosis," *IEEE Trans. Syst. Man Cybern.*, vol. 21, pp.815-824, 1991.

[14] S.S. Iyengar, D.N. Jayasimha and D. Nadig, "A versatile architecture for the distributed sensor integration problem," *IEEE Trans. Comput.*, C-43, pp.175-185, Feb. 1994.14

[15] N. K. Jerne, "The Immune System" *Sci. Am.* Vol. 229, No.1, pp. 52-60, 1973.

[16] P. H. Richter, "Complexity and Regulation of the Imm¡une System: The Network Approach," in *Proc. Working Conf. on System Theory in Immunology,* pp. 219-227, Rome, Italy, May 1978 (Springer-Verlag, New York).

[17] S. Levin, Eds, *System Theories in Immunology*, in Lecture Notes in Biomathematics, Springer-Verlag, 1978.

[18] R. R. Mohler, C. Bruni, and A. Gandolfi, "A Systems Approach to Immunology", *Proc. of the IEEE*, Vol. 68, No. 8, pp. 964-990, August, 1980.

[19] A. S. Perelson *Eds.*, *Theoretical Immunology*, Part I,II, Addison Wesley, 1988.

[20] H. Atlan and I. R. Cohen, Eds, *Theories of Immune Networks*, Springer-Verlag, 1989.

[21] M. A. Cohen, and S. Grossberg, "Absolute Stability of Global Pattern Formation and Parallel Memory Storage by Competetive Neural Networks," *IEEE Trans Sys. Man Cybern.*, SMC-13, pp. 815-826, 1983.

[22] Y. Ishida, "An Immune Network Approach to Sensor-Network with Self-Organization for Sensor and Process Faults", Technical Report of Nara Institute of Science and Technology, 1995.