Resilient Alarm Logic Design for Process Networks

Chii-Shang Tsai

Department of Chemical Engineering, Far East College, Hsin Shih, Tainan, Taiwan, Republic of China

Chuei-Tin Chang,* Kwan-Hwa Chen, and Shang-Kai Chu

Department of Chemical Engineering, National Cheng Kung University, Tainan, Taiwan 70101, Republic of China

A novel alarm-system design strategy, which takes full advantage of the inherent hardware and spatial redundancy in a process network, is proposed in this paper. Specifically, systematic procedures have been developed to identify independent methods for evaluating any alarm variable in the process and to synthesize corresponding alarm generation logic. To implement this logic, the error models in data reconciliation and the formulas for evaluating conditional probabilities of type I and II mistakes have also been derived. The results of applying the proposed approach to the application example show that it is indeed superior to any of the existing design techniques. This is because the resulting alarm system is appropriately tailored to minimize the expected loss. More importantly, it is resilient in the sense that the system performs satisfactorily even under the influence of various sensor malfunctions.

Introduction

Alarm generation is a basic function of the protective system in any chemical process plant. The current practice in the industry is simply to compare online measurements of the variable of interest with a threshold value. In most applications, this threshold value is identified on the basis of *process* considerations, e.g., the onset temperature of a runaway reaction or the rupture pressure of a storage vessel. Thus, the implementation issues of an alarm generation strategy are really *not* the same as those concerning the SPC-related techniques for incipient fault detection.[1] Notice that, in the latter case, the control limits are determined mainly according to *statistical* data obtained during normal operation.

To protect the process against potential hazards, the decision concerning whether to set off an alarm usually must be made as soon as a new batch of online data become available. However, all sensor measurements are inevitably subject to random and/or gross errors.[2] Thus, two types of mistakes may be committed in the above decision-making process. First, spurious alarms may be produced because of measurement errors when the variations of the process variables are actually within acceptable limits (type I mistakes). Second, the system may fail to detect the existence of hazardous operating conditions, and thus no alarms are generated (type II mistakes).

A common industrial practice to reduce the chance of misjudgment is to introduce *hardware redundancy* in the alarm system.[3] Specifically, several independent sensors are installed to monitor the same process variable. Any inconsistency identified in the measurement data obtained from different sensors is usually resolved on the basis of operation experience or an arbitrarily chosen alarm logic. It should be noted that the implied objective of such a practice is to achieve a compromise between the conflicting emphases on decreasing type I and II mistakes. Although this approach is effective on a qualitative basis, there are still several deficiencies. In particular, the conventional alarm strategy utilizes only the information obtained from the redundant sensors for measuring the process variable of interest. Consequently, a large amount of additional useful information embedded in the process system is neglected entirely. Also, the alarm-generating logics adopted in industrial applications have always been developed on an ad hoc basis and thus may not be cost optimal. This drawback can be significant in cases when the financial loss of misjudgment is large.

One possible way to improve the current operation is to make use of the *spatial redundancy* (or analytical redundancy) embedded in any process network. The hardware redundancy built in a traditional alarm system can actually be viewed as a special case of spatial redundancy in the sensor network of the overall process. Consequently, the reconciled data, rather than the raw measurement data, are adopted in this study for alarm generation purposes. There are several advantages in taking this approach. The most obvious one is that the variance of each estimate is known to be smaller than that of the raw data.[4] However, it should be noted that, as a result of gross and random errors in the measurement data, all reconciled estimates are also subject to errors and thus type I and II mistakes may still occur in generating alarms. In a previous study,[5] a systematic method for synthesizing the optimal *flow* alarm logics has been developed according to the design criterion of minimizing expected loss. Although this method was demonstrated to be cost-effective if only random errors exist, the problems caused by gross errors were not discussed in sufficient detail. Because the decision to set off an alarm in the proposed system is made on the basis of multiple sets of *independent* measurements, we feel that its overall performance should still be better;

* To whom all correspondence should be addressed. Tel: 886-6-208-6969 ext. 62663. Fax: 886-6-234-4496. E-mail: ctchang@mail.ncku.edu.tw.

10.1021/ie0002780 CCC: $19.00
Published on Web 12/04/2000

i.e., the chance of misjudgment is lower than that of the other approaches even under the influence of undetected gross errors. Therefore, to verify our proposition concerning alarm resiliency and to extend the approach to all measurable variables in the process network, i.e., the flow rates, the temperatures, and the concentrations, it is necessary to perform a more comprehensive study.

Several specific tasks have been accomplished in this work. First, notice that the mass balances are linear equations and the energy and component balances are essentially nonlinear in nature. The reconciliation error models obtained on the basis of only the former are apparently not applicable when the latter two types of constraint equations are also included in the estimation algorithm. A modified version is thus derived in this work to facilitate the development of a comprehensive alarm strategy. Second, a systematic procedure has been established to identify independent methods for evaluating any alarm variable with available measurement data. Third, the design techniques for trip systems[6,7] have been modified for the purpose of synthesizing optimal alarm logics. Because the probabilities of false alarms and undetected failures must be obtained in order to implement this logic synthesis method, an online estimation procedure for these parameters has also been developed in this work. Finally, a series of simulation studies have been carried out to demonstrate the advantages of our approach. In particular, the proposed alarm logic is not only cost optimal but also resilient. The alarm resilience is, in fact, ensured by incorporating the redundant information embedded in the mass, component, and energy balance relations. As a result, the logic should be effective even under the influence of gross measurement errors.

Error Models of Process Variables

To explore the structural characteristics of chemical processes, it is convenient to represent the process flow diagrams with *process networks*.[2] On each arc in a process network, the variables of interest are the total flow rate, the temperature, and the component concentrations. In this paper, they are referred to as the *process variables*. The true values of these variables can be viewed as

$$v_j^{\mathrm{t}} = v_j^{\mathrm{d}} + \Delta_j \qquad j = 1, 2, ..., N_{\mathrm{V}} \tag{1}$$

where v_j^{t} denotes the true value of the jth process variable, v_j^{d} represents its design value, Δ_j is the corresponding difference resulting from unknown disturbances, and N_{V} is the total number of process variables. Notice that N_{V} can be determined with the number of species C and the number of arcs M in a process network; i.e., $N_{\mathrm{V}} = (C + 2)M$.

In this study, Δ_j is assumed to be a normally distributed random variable with a *time-variant* mean. Specifically, its expected value is zero when the system is operated at normal steady state and otherwise when faults occur. Notice that, although eq 1 is applicable to the temperature, concentrations, and total flow rate, these three types of variables are treated equally here for the sake of conciseness. More explicit notation will be introduced later.

Next, let us assume that the total number of unmeasured process variables is U. The measurement errors are related to their true values according to the following constraint:

$$v_j^{(i)} = v_j^{\mathrm{t}} + e_j^{(i)} \qquad i = 1, 2, ..., m_j \qquad j = 1, 2, ..., N_{\mathrm{V}} - U \tag{2}$$

where $v_j^{(i)}$ represents the measurement value of the jth variable using sensor i, $e_j^{(i)}$ denotes the corresponding random error, and m_j is the total number of independent sensors used to measure the jth variable. In this study, $e_j^{(i)}$ is treated as a normally distributed random variable with zero mean. It should be reasonable to believe that the variance of each measurement error can be acquired from the vendor or an analysis of its historical data.

As indicated above, there may be multiple redundant sensors installed on a stream to measure the same variable. To facilitate concise formulation of the error model for data reconciliation, this stream is treated in our study as several fictitious streams connected in series by fictitious nodes. If the temperature, concentrations, and total flow rate of a particular process stream are measured respectively with r_j^{T}, $r_{j,k}^{\mathrm{x}}$ ($k = 1, 2, ..., C$), and r_j^{m} redundant sensors, then the number of fictitious streams n_j^{fic} should be

$$n_j^{\mathrm{fic}} = \max(r_j^{\mathrm{T}}, r_j^{\mathrm{m}}, r_{j,1}^{\mathrm{x}}, r_{j,2}^{\mathrm{x}}, ..., r_{j,C}^{\mathrm{x}}) \tag{3}$$

Consequently, there can be *at most* one sensor for each process variable on a fictitious stream and thus the superscripts of $v_j^{(i)}$'s and $e_j^{(i)}$'s in eq 2 can be dropped without causing confusion.

Estimation Errors in Reconciled Data

In this work, the reconciled values of the process variables are utilized in the alarm-generation process. Consequently, the corresponding estimation errors must also be analyzed. It should be noted that the choice of constraint equations used in the reconciliation calculation is dependent upon the variables involved in alarm logic. For example, it is only necessary to consider mass balance in the flow−alarm algorithm, but both mass and energy balances must be included in a temperature−alarm system. The issue of selecting the appropriate set of constraint equations for data reconciliation has already been addressed fully in the past, e.g., see work by Mah.[2] Let us express these constraint equations in a general form, i.e.

$$\boldsymbol{\Phi}(\mathbf{v}^{\mathrm{t}}, \mathbf{u}^{\mathrm{t}}, \boldsymbol{\eta}^{\mathrm{t}}) = 0 \tag{4}$$

where $\boldsymbol{\Phi} = [\Phi_1, \Phi_2, ..., \Phi_{N_{\mathrm{E}}}]^{\mathrm{T}}$ denotes the vector of constraint functions, $\mathbf{v}^{\mathrm{t}} = [v_1^{\mathrm{t}}, v_2^{\mathrm{t}}, ..., v_{N_V-U}^{\mathrm{t}}]^{\mathrm{T}}$ and $\mathbf{u}^{\mathrm{t}} = [u_1^{\mathrm{t}}, u_2^{\mathrm{t}}, ..., u_U^{\mathrm{t}}]^{\mathrm{T}} = [v_{N_V-U+1}^{\mathrm{t}}, ..., v_{N_V}^{\mathrm{t}}]^{\mathrm{T}}$ represent respectively the true values of the measured and unmeasured process variables, and $\eta^{\mathrm{t}} = [\eta_1^{\mathrm{t}}, \eta_2^{\mathrm{t}}, ..., \eta_{N_{\mathrm{P}}}^{\mathrm{t}}]^{\mathrm{T}}$ is the vector of unknown parameters, i.e., the reaction extent and the split fraction. In principle, the mass and/or energy balance described in eq 4 should be exact. If empirical equations are used as the constraint equations in data

reconciliation, they are assumed to be a fairly accurate approximation of the reality in this study.

If the constraint equations in eq 4 are linear, then the reconciled values of process variables can be determined analytically.[8] Otherwise, an iterative computation procedure is needed.[9] Notice that the intention for installing an alarm is usually to protect the process against certain hazards. Prompt remedial actions must be taken if such emergency situations occur. Thus, the standard reconciliation algorithm for nonlinear systems is really inappropriate in alarm-generation applications because of its iterative nature. In this study, a linearized version of the constraint equations is adopted to produce estimates of the process variables. In particular, let us linearize eq 4 with respect to the measurement values, i.e.

$$\mathbf{A}^{(1)}\Delta\mathbf{v} + \mathbf{A}^{(2)}\Delta\mathbf{u} + \mathbf{A}^{(3)}\Delta\boldsymbol{\eta} = -\delta \tag{5}$$

where

$$\Delta\mathbf{v} = \mathbf{v}^{\mathrm{t}} - \mathbf{v} = [v_1^{\mathrm{t}} - v_1,\ v_2^{\mathrm{t}} - v_2,\ ...,\ v_{N_{\mathrm{V}}-U}^{\mathrm{t}} - v_{N_{\mathrm{V}}-U}]^{\mathrm{T}} \tag{6}$$

$$\Delta\mathbf{u} = \mathbf{u}^{\mathrm{t}} - \mathbf{u} = [u_1^{\mathrm{t}} - u_1,\ u_2^{\mathrm{t}} - u_2,\ ...,\ u_U^{\mathrm{t}} - u_U]^{\mathrm{T}} \tag{7}$$

$$\Delta\boldsymbol{\eta} = \boldsymbol{\eta}^{\mathrm{t}} - \boldsymbol{\eta} = [\eta_1^{\mathrm{t}} - \eta_1,\ \eta_2^{\mathrm{t}} - \eta_2,\ ...,\ \eta_{N_{\mathrm{P}}}^{\mathrm{t}} - \eta_{N_{\mathrm{P}}}]^{\mathrm{T}} \tag{8}$$

It is assumed that the values of u_k's ($k = 1, 2, ..., U$) and η_l ($l = 1, 2, ..., N_{\mathrm{P}}$) can be determined by solving various subsets of the N_{E} constraint equations in eq 4 according to the measurement values v_j's ($j = 1, 2, ..., N_{\mathrm{V}} - U$). Several systematic procedures will be described later in this paper to identify the suitable evaluation functions for this purpose. The coefficient matrices in eq 5 can be expressed as

$$\mathbf{A}^{(1)} = \left(\frac{\partial\Phi_i}{\partial v_j^{\mathrm{t}}}(\mathbf{v},\mathbf{u},\boldsymbol{\eta})\right)_{N_{\mathrm{E}}\times(N_{\mathrm{V}}-U)} \tag{9}$$

$$\mathbf{A}^{(2)} = \left(\frac{\partial\Phi_i}{\partial u_j^{\mathrm{t}}}(\mathbf{v},\mathbf{u},\boldsymbol{\eta})\right)_{N_{\mathrm{E}}\times U} \tag{10}$$

$$\mathbf{A}^{(3)} = \left(\frac{\partial\Phi_i}{\partial \eta_j^{\mathrm{t}}}(\mathbf{v},\mathbf{u},\boldsymbol{\eta})\right)_{N_{\mathrm{E}}\times N_{\mathrm{P}}} \tag{11}$$

$$\boldsymbol{\delta} = [\Phi_1(\mathbf{v},\mathbf{u},\boldsymbol{\eta}),\ \Phi_2(\mathbf{v},\mathbf{u},\boldsymbol{\eta}),\ ...,\ \Phi_{N_{\mathrm{E}}}(\mathbf{v},\mathbf{u},\boldsymbol{\eta})]^{\mathrm{T}} \tag{12}$$

Again notice that the elements of the above matrices and vector are functions of the measured and *calculated* values of the process variables and parameters.

To derive an explicit formula to estimate reconciliation errors, it is often necessary to rearrange the order of variables in vector $\mathbf{v}$ and partition $\mathbf{A}^{(1)}$ accordingly into two matrices: one $N_{\mathrm{E}} \times (N_{\mathrm{V}} - N_{\mathrm{E}} + N_{\mathrm{P}})$ matrix $\mathbf{A}^{(11)}$ and another $N_{\mathrm{E}} \times (N_{\mathrm{E}} - U - N_{\mathrm{P}})$ matrix $\mathbf{A}^{(12)}$. Specifically,

$$\mathbf{A}^{(1)} = [\mathbf{A}^{(11)} \quad \mathbf{A}^{(12)}] \tag{13}$$

Furthermore, this partition must be done in such a way that a nonsingular matrix $\mathbf{T}$ can be constructed with $\mathbf{A}^{(12)}$, $\mathbf{A}^{(2)}$, and $\mathbf{A}^{(3)}$, i.e.

$$\mathbf{T} = [\mathbf{A}^{(12)} \quad \mathbf{A}^{(2)} \quad \mathbf{A}^{(3)}]_{N_{\mathrm{E}}\times N_{\mathrm{E}}} \tag{14}$$

When $\mathbf{T}^{-1}$ is premultiplied, eq 5 can be transformed into

$$\begin{bmatrix} \mathbf{C}_1 & \mathbf{I}_1 & \mathbf{0} & 0 \\ \mathbf{C}_2 & \mathbf{0} & \mathbf{I}_2 & \mathbf{0} \\ \mathbf{C}_3 & \mathbf{0} & 0 & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{v}_1 \\ \Delta\mathbf{v}_2 \\ \Delta\mathbf{u} \\ \Delta\boldsymbol{\eta} \end{bmatrix} = -\begin{bmatrix} \boldsymbol{\delta}_1 \\ \boldsymbol{\delta}_2 \\ \boldsymbol{\delta}_3 \end{bmatrix} \tag{15}$$

where $\Delta\mathbf{v}_1$ is a $N_{\mathrm{V}} - N_{\mathrm{E}} + N_{\mathrm{P}}$ vector corresponding to the columns of $\mathbf{A}^{(11)}$; $\Delta\mathbf{v}_2$ is a $N_{\mathrm{E}} - U - N_{\mathrm{P}}$ vector corresponding to the columns of $\mathbf{A}^{(12)}$; the $\mathbf{0}$'s denote matrices in which all entries are zero; $\mathbf{I}_1$, $\mathbf{I}_2$, and $\mathbf{I}_3$ are identity matrices; and the matrices $\mathbf{C}_1$, $\mathbf{C}_2$, and $\mathbf{C}_3$ can be obtained by partitioning the product of $\mathbf{T}^{-1}$ and $\mathbf{A}^{(11)}$. Notice that the number of rows in $\mathbf{C}_1$, $\mathbf{C}_2$, and $\mathbf{C}_3$ should be $N_{\mathrm{E}} - U - N_{\mathrm{P}}$, U, and N_{P}, respectively. Finally, the vectors $\boldsymbol{\delta}_1$, $\boldsymbol{\delta}_2$, and $\boldsymbol{\delta}_3$ can be determined in a similar way by partitioning $\mathbf{T}^{-1}\boldsymbol{\delta}$.

On the basis of eq 15, it can be shown that the reconciled values of the measured variables $\hat{\mathbf{v}}$ can be expressed as

$$\hat{\mathbf{v}} = \mathbf{v} - \mathbf{QB}^{\mathrm{T}}(\mathbf{BQBT})^{-1}\boldsymbol{\delta}_1 \tag{16}$$

where $\mathbf{Q}$ is the covariance matrix associated with $\mathbf{v}$ and matrix $\mathbf{B}$ is defined as

$$\mathbf{B} = [\mathbf{C}_1 \quad \mathbf{I}_1] \tag{17}$$

Notice that eq 16 is the result of minimizing the weighted sum of squared errors $(\hat{\mathbf{v}} - \mathbf{v})^{\mathrm{T}}\mathbf{Q}^{-1}(\hat{\mathbf{v}} - \mathbf{v})$ under the constraint of eq 5. This derivation is omitted in the present paper for the sake of brevity.

Let us now define the vector of estimation errors $\mathbf{d}$ as

$$\mathbf{d} = \hat{\mathbf{v}} - \mathbf{v} \tag{18}$$

where

$$\mathbf{d} = [d_1,\ d_2,\ ...,\ d_{N_{\mathrm{V}}-U}]^{\mathrm{T}}$$

Substituting eq 18 and the first part of eq 15 into eq 16, one can produce the following result:

$$\mathbf{d} = \mathbf{e} - \mathbf{QB}^{\mathrm{T}}(\mathbf{BQB}^{\mathrm{T}})^{-1}\mathbf{Be} = [\mathbf{I} - \mathbf{QB}^{\mathrm{T}}(\mathbf{BQB}^{\mathrm{T}})^{-1}\mathbf{B}]\mathbf{e} \tag{19}$$

Thus, the error of every reconciled value can be estimated by a linear combination of the measurement errors of all sensors, i.e.

$$d_k = \sum_{j=1}^{N_{\mathrm{V}}-U} w_{kj}e_j \qquad k = 1, 2, ..., N_{\mathrm{V}} - U \tag{20}$$

where the coefficients w_{kj} can be determined by eq 19.

Threshold Limits of Alarm Variables

In a chemical plant, a subset of the $N_{\mathrm{V}} - U$ measured process variables may be selected as the *alarm variables* on the basis of process considerations. More specifically, each of these variables must satisfy one or more

operational constraints. A typical constraint can be written as

$$v_A^t - v_A^L \geq 0 \tag{21a}$$

$$v_A^U - v_A^t \geq 0 \tag{21b}$$

where v_A^t represents the true value of the Ath process variable ($1 \leq A \leq N_V - U$) and v_A^L and v_A^U denote respectively the lower and upper threshold limits. For convenience, the operational constraint can be expressed in an alternative general form as

$$G(v_A^t) \geq 0 \tag{22}$$

where G is referred to as a *performance function* in this study.

Obviously, an alarm is set off as an indication of constraint violation. Because the true process state can never be determined, one has to rely on the measurement data to evaluate the performance function. In other words, the values of *indicator function* $G^{(s)}$ must be computed according to

$$G^{(s)} = G(v_A^{(s)}) \qquad s = 1, 2, ..., N_A \tag{23}$$

where $v_A^{(s)}$ denotes the value of the alarm variable obtained with the sth independent method and N_A is the total number of evaluation methods.

Apparently, the alarm variable can be monitored *directly* with a sensor. Although there is at most one sensor for each variable on any arc in the process network, it is still possible to identify more than one independent method to determine the alarm variable *indirectly* according to the measurement data of other process variables. These indirect methods can be identified mainly by exploiting the inherent spatial redundancy implied in the mass, component, and energy balance relations. Because of measurement errors, the values of the indicator function evaluated with data obtained from different methods are, in general, not consistent with one another. Nonetheless, one is still required to make a decision concerning whether to set off an alarm with these data. Thus, let us now turn our attention to the development of an optimal alarm-generation strategy.

Optimal Alarm-Generation Logic

As indicated above, there may be several different evaluation methods available for the purpose of monitoring the same variable of interest. Let us express these methods with a set of *evaluation functions* $\Psi_A^{(s)}$, i.e.

$$v_A^{(s)} = \Psi_A^{(s)}(\mathbf{v}) \tag{24}$$

$$s = 1, 2, ..., N_A$$

where each $\Psi_A^{(s)}$ is derived from a subset of the constraint equations in eq 4 and $\mathbf{v}$ is the vector of measurement values of all measured process variables. The explicit forms of these evaluation functions can be identified with the procedures described in the next section. Assuming that such functions are available, one should be able to compute $v_A^{(s)}$'s online and then substitute them into the performance function G to assess the current operation status. On the basis of these results, a set of binary *indicator variables* y_s can be determined accordingly, i.e.

$$y_s = \begin{bmatrix} 1 & \text{if } G^{(s)} < 0 \\ 0 & \text{otherwise} \end{bmatrix} \tag{25}$$

where $G^{(s)}$ is the indicator function defined in eq 23.

The system alarm should then be generated on the basis of these indicators. The logic for setting off the alarm can be explicitly expressed with an *alarm function* $f(\mathbf{y})$, i.e.

$$f(\mathbf{y}) = \begin{bmatrix} 1 & \text{if the system is generating an alarm} \\ 0 & \text{otherwise} \end{bmatrix} \tag{26}$$

where $\mathbf{y} = [y_1, y_2, ..., y_{N_A}]^T$.

Obviously, the values of the indicator variables y_s may not be consistent with the values of true state v_A^t. Let us consider the true value of the performance function, i.e.

$$G^t = G(v_A^t) \tag{27}$$

There are two kinds of mistakes that can be identified accordingly; i.e., y_s is set to be 1 when $G^t \geq 0$ (type I mistake) or y_s is set to be 0 when $G^t < 0$ (type II mistake). Similarly, the mistakes committed in generating the *system alarm* can also be classified into types I and II. The conditional probabilities associated with these two mistakes, i.e., P_a and P_b, can be expressed as

$$P_a = Pr\{f(\mathbf{y}) = 1 | G^t \geq 0\} \tag{28}$$

$$P_b = Pr\{f(\mathbf{y}) = 0 | G^t < 0\} \tag{29}$$

Because both types of mistakes result in financial losses, there are incentives for developing an optimal alarm-generation logic which minimizes the expected loss L, i.e.

$$\min_{f(\mathbf{y})} L \tag{30}$$

where

$$L = C_a(1 - P_F)P_a + C_b P_F P_B \tag{31}$$

where C_a and C_b respectively denote the losses caused by type I and II mistakes in alarm generation. P_F is the demand probability, which is defined as the probability of violating the constraint; i.e., $P_F = Pr\{G^t < 0\}$.

It can be shown[5] that the expected loss is minimized if the alarm function is chosen such that

$$f(\mathbf{y}) = \begin{cases} 1 & \text{if } h(\mathbf{y}) > 0 \\ 0 & \text{if } h(\mathbf{y}) \leq 0 \end{cases} \tag{32}$$

where

$$h(\mathbf{y}) = C_b P_F Pr\{\mathbf{y} | G^t < 0\} - C_a(1 - P_F)Pr\{\mathbf{y} | G^t \geq 0\} \tag{33}$$

After obtaining the values of $f(\mathbf{y})$ for all possible $\mathbf{y}$, its functional form can be constructed accordingly. With the

functional form given, the logic associated with $f(\mathbf{y})$ can be implemented as a hard-wired circuit or as a computer program.

It should be noted that, to compute $h(\mathbf{y})$ and then construct $f(\mathbf{y})$, the estimates of demand probability and also conditional probabilities $Pr\{\mathbf{y}|G^t < 0\}$ and $Pr\{\mathbf{y}|G^t \geq 0\}$ must be obtained first. For illustration purpose, let us consider the constraint given in eq 21b. Using the relation presented in eq 2, the demand probability can be expressed as

$$P_F = Pr\{v_A^t > v_A^U\} = Pr\{\Delta_A > v_A^U - v_A^d\} \quad (34)$$

Because the mean of the deviation Δ_A is time-variant and thus unknown, it is not possible to evaluate P_F with eq 34. On the other hand, from the assumption that the means of measurement errors are negligible, the demand probability can be estimated according to the reconciled value of the alarm variable ($\hat{v}_A$), i.e.

$$P_F = Pr\{d_A < \hat{v}_A - v_A^U\} \quad (35)$$

where d_A denotes the estimation error. Notice that $\mathbf{B}$ and $\boldsymbol{\delta}_1$ in eqs 16 and 19 must be computed each time when a new batch of measurement data $\mathbf{v}$ is available and, then, $\hat{v}_A$ and the parameters of the probability density function of d_A can be determined accordingly. Thus, it is necessary to evaluate P_F online based on the updated value of $\hat{v}_A - v_A^U$ with a standard look-up table for normal probability distribution.

The computation of the conditional probabilities can be greatly simplified if the measurement methods are *s*-independent. In particular, they can be written as

$$Pr\{\mathbf{y}|G^t \geq 0\} = \prod_{s=1}^{N_A} a_s^{y_s}(1 - a_s)^{1-y_s} \quad (36)$$

$$Pr\{\mathbf{y}|G^t < 0\} = \prod_{s=1}^{N_A} (1 - b_s)^{y_s} b_s^{1-y_s} \quad (37)$$

where

$$a_s = Pr\{y_s = 1|G^t \geq 0\} = Pr\{v_A^{(s)} > v_A^U | v_A^t \leq v_a^U\} \quad (38)$$

$$b_s = Pr\{y_s = 0|G^t < 0\} = Pr\{v_A^{(s)} \leq v_A^U | v_A^t > v_A^U\} \quad (36)$$

From the definition of conditional probability, these two parameters can be written as

$$a_s = \frac{Pr\{v_A^{(s)} > v_A^U,\ v_A^t \leq v_A^U\}}{Pr\{v_A^t \leq v_A^U\}} \quad (40)$$

$$b_s = \frac{Pr\{v_A^{(s)} \leq v_A^U,\ v_A^t > v_A^U\}}{Pr\{v_A^t > v_A^U\}} \quad (41)$$

Because the denominators in eqs 40 and 41 are simply $1 - P_F$ and P_F, respectively, it is thus only necessary to develop formulas for evaluating the numerators of these two parameters. Again, they are evaluated in this study according to the reconciled value of the alarm variable, i.e.

$$Pr\{v_A^{(s)} > v_A^U,\ v_A^t \leq v_A^U\} = Pr\{d_A - \epsilon_A^{(s)} < \hat{v}_A - v_A^U,\ d_A \geq \hat{v}_A - v_A^U\} \quad (42)$$

$$Pr\{v_A^{(s)} \leq v_A^U,\ v_A^t > v_A^U\} = Pr\{d_A - \epsilon_A^{(s)} \geq \hat{v}_A - v_A^U,\ d_A < \hat{v}_A - v_A^U\} \quad (43)$$

where $\epsilon_A^{(s)}$ represents the error of the *s*th evaluation method in determining the alarm variable. It is estimated in this study according to

$$\epsilon_A^{(s)} = \sum_{i=1}^{N_V - U} \frac{\partial \Psi_A^{(s)}}{\partial v_i} e_i \quad (44)$$

$$s = 1, 2, ..., N_A$$

where all partial derivatives are evaluated with measurement values. Consequently, the expected value of every $\epsilon_A^{(s)}$ is assumed to be very close to zero in this work. The parameters of the joint probability density function associated with the two random variables, $d_A - \epsilon_A^{(s)}$ and d_A, can be obtained from the means and variances of the measurement errors e_i using eqs 20, 24, and 44. Thus, it is always possible to evaluate these joint probabilities online by performing the numerical integration implied in eqs 42 and 43.

Identification of Independent Evaluation Methods

For illustration clarity, it is now necessary to divide the measured process variables into three different categories, i.e., the flow rates, temperatures, and concentrations. Specifically, let

$$V = \{M, T, X_1, ..., X_C\} \quad (45)$$

where

$$V = \{v_j | j = 1, 2, ..., N_V - U\} \quad (46)$$

$$M = \{m_l | l \text{ is the label of an arc on which a flow sensor is installed}\} \quad (47)$$

$$T = \{T_l | l \text{ is the label of an arc which a temperature sensor is installed}\} \quad (48)$$

$$X_i = \{x_{li} | l \text{ is the label of an arc on which a sensor is installed to measure the concentration of component } i\} \quad (49)$$

In the above definitions, m_l and T_l denote respectively the measurement values of the mass flow rate and temperature of the process stream associated with arc *l*, and x_{li} is that of the *i*th component's mass fraction in the same stream.

Let us first consider the flow alarm installed on one particular arc, say *a*, in a process network. It has been well established that all of the different ways of *indirectly* evaluating the same mass flow are given by the

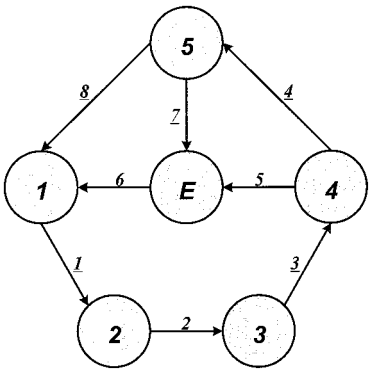

Figure 1. Example to illustrate the procedure for identifying independent evaluation methods: the process network.

cut sets that contain arc *a* in which the mass flow of every other arc is measured.[10,11] However, because some of the elements may appear in more than two of these cut sets, the corresponding evaluation methods are statistically dependent. Consequently, it is necessary to select out cut sets that do not contain common arcs other than *a*. Tsai and Chang[5] proposed a simple digraph-based procedure to perform this task. For the sake of completeness, this procedure (algorithm I) is outlined below:

1. Consider the original process network. Merge the input and output nodes of every arc on which the flow sensor is not installed. Let the resulting process graph be the *current digraph* and also $i = 1$.

2. Find a cut set $\mathbf{K}^{(i)}$ of the current digraph which contains arc *a*.

3. Merge the input and output nodes of every arc in $\mathbf{K}(i)$ except those of arc *a*. Let the resulting graph be the *current digraph* and $i = i + 1$.

4. Repeat steps 2 and 3 until arc *a* itself forms a loop.

It should be noted that the sets $\mathbf{K}^{(i)}$ ($i = 1, 2, ...$) obtained with algorithm I may not be unique. This is due to the fact that more than one cut set can usually be found in any given graph. Conceivably, other candidate measurement methods may be identified if different cut sets are adopted in step 2.

Let us next consider a similar problem concerning the temperature alarm on arc *a*. Notice that the connection between the cut sets and the evaluation methods in a mass-flow network is still valid in an energy-flow network. On the basis of this insight, a systematic procedure (algorithm II) has been developed to identify a set of independent temperature evaluation methods. This procedure and a simple illustration example are presented in the appendix. Finally, it should be noted that, without reactors and splitters, the equation form of component-flow constraint in a process network is also identical to that of mass-flow or energy-flow constraint. Consequently, a second modified version of algorithm I can be developed to identify indirect independent concentration (or component flow) evaluation methods.[12] For the sake of brevity, the description of this last procedure is not repeated in this paper.

Application Example

Let us consider the process network in Figure 1 representing a simplified ammonia process.[13] This network consists of six nodes and eight arcs, with node E denoting the environment and nodes 1−5 the major units. Under normal operating conditions, it is assumed that the system is at its original steady state and can be described with the parameters presented in Tables 1 and 2. The statistical parameters associated with the true temperature of every stream can be found in the second and third columns of Table 1, and those of the true flow rate are presented in Table 2. As mentioned previously, the difference between the true value of a process variable and its design value is treated in this work as a random variable with zero mean during normal operation. Thus, the mean values listed in Tables 1 and 2 are also used as the design values in the present example. In this process network, the temperature and flow rate of every arc are measured except T_8, m_3, and m_5. Each of them is measured with one sensor only. The variance of each measurement error can also be found in Tables 1 and 2.

Table 1. Statistical Parameters of Temperatures under Normal Operating Conditions

stream no. (j)	$E[T_j^t]$	$\mathrm{Var}[T_j^t]$	$\mathrm{Var}[e_{T_j}]$
1	268.0	18.6	4.65
2	268.0	18.6	4.65
3	268.0	18.6	4.65
4	238.1	15.1	3.77
5	312.9	116.2	29.0
6	229.8	23.6	5.91
7	119.0	7.41	1.85
8	357.1	55.3	NA

Table 2. Statistical Parameters of Flow Rates under Normal Operating Conditions

stream no. (j)	$E[m_j^t]$	$\mathrm{Var}[m_j^t]$	$\mathrm{Var}[e_{m_j}]$
1	34.3	0.17	0.042
2	34.3	0.17	0.042
3	34.3	0.17	NA
4	20.6	0.051	0.013
5	13.7	0.16	NA
6	24.0	0.14	0.034
7	10.3	0.025	0.006
8	10.3	0.024	0.006

The reconciled values of temperatures and flow rates can be obtained with eq 16. Because the unknown parameters in $\boldsymbol{\eta}$ do not exist in this case, the vector and matrices used to compute $\mathbf{B}$ and $\boldsymbol{\delta}_1$ in this equation were chosen to be

$$\boldsymbol{\delta} = \begin{bmatrix} m_8 + m_6 - m_1 \\ m_1 - m_2 \\ m_2 - m_3 \\ m_3 - m_4 - m_5 \\ m_4 - m_7 - m_8 \\ m_8Cp_8T_8 + m_6Cp_6T_6 - m_1Cp_1T_1 \\ m_1Cp_1T_1 - m_2Cp_2T_2 \\ m_2Cp_2T_2 - m_3Cp_3T_3 \\ m_3Cp_3T_3 - m_4Cp_4T_4 - m_5Cp_5T_5 \\ m_4Cp_4T_4 - m_7Cp_7T_7 - m_8Cp_8T_8 \end{bmatrix} \quad (50)$$

$$\mathbf{A}^{(11)} = \begin{bmatrix} \mathbf{A}_{11}^{(11)} & \mathbf{0} \\ \mathbf{A}_{21}^{(11)} & \mathbf{A}_{22}^{(11)} \end{bmatrix} \quad (51)$$

and

$$\mathbf{T} = \begin{bmatrix} \mathbf{T}_{11} & \mathbf{0} & \mathbf{T}_{13} \\ \mathbf{T}_{21} & \mathbf{T}_{22} & \mathbf{T}_{23} \end{bmatrix} \quad (52)$$

where **0** denotes a zero matrix of appropriate dimension and

$$\mathbf{A}_{11}^{(11)} = \begin{bmatrix} 0 & +1 & +1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \\ +1 & 0 & -1 \end{bmatrix} \tag{53}$$

$$\mathbf{A}_{21}^{(11)} = \begin{bmatrix} 0 & Cp_6T_6 & Cp_8T_8 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -Cp_4T_4 & 0 & 0 \\ Cp_4T_4 & 0 & -Cp_8T_8 \end{bmatrix} \tag{54}$$

$$\mathbf{A}_{22}^{(11)} = \begin{bmatrix} 0 & 0 & Cp_6m_6 \\ 0 & 0 & 0 \\ -Cp_3m_3 & 0 & 0 \\ Cp_3m_3 & -Cp_5m_5 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{55}$$

$$\mathbf{T}_{11} = \begin{bmatrix} -1 & 0 & 0 \\ +1 & -1 & 0 \\ 0 & +1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{56}$$

$$\mathbf{T}_{13} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \\ +1 & -1 & 0 \\ +1 & 0 & 0 \end{bmatrix} \tag{57}$$

$$\mathbf{T}_{21} = \begin{bmatrix} -Cp_1T_1 & 0 & 0 \\ Cp_1T_1 & -Cp_2T_2 & 0 \\ 0 & Cp_2T_2 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -Cp_7T_7 \end{bmatrix} \tag{58}$$

$$\mathbf{T}_{22} = \begin{bmatrix} -Cp_1m_1 & 0 & 0 & 0 \\ Cp_1m_1 & -Cp_2m_2 & 0 & 0 \\ 0 & Cp_2m_2 & 0 & 0 \\ 0 & 0 & -Cp_4m_4 & 0 \\ 0 & 0 & Cp_4m_4 & -Cp_7m_7 \end{bmatrix} \tag{59}$$

$$\mathbf{T}_{23} = \begin{bmatrix} 0 & 0 & Cp_8m_8 \\ 0 & 0 & 0 \\ -Cp_3T_3 & 0 & 0 \\ Cp_3T_3 & -Cp_5T_5 & 0 \\ 0 & 0 & -Cp_8m_8 \end{bmatrix} \tag{60}$$

In the present example, it is assumed that an alarm system must be installed on arc 1 to protect against the detrimental outcomes caused by high temperature. The threshold limit T_1^{U} selected in the simulation studies is 277.9 K. The first independent evaluation method is naturally associated with the temperature sensor for directly measuring arc 1; i.e., when $s = 1$,

$$T_1^{(1)} = \Psi_{T_1}^{(1)}(T_1) = T_1 \tag{61}$$

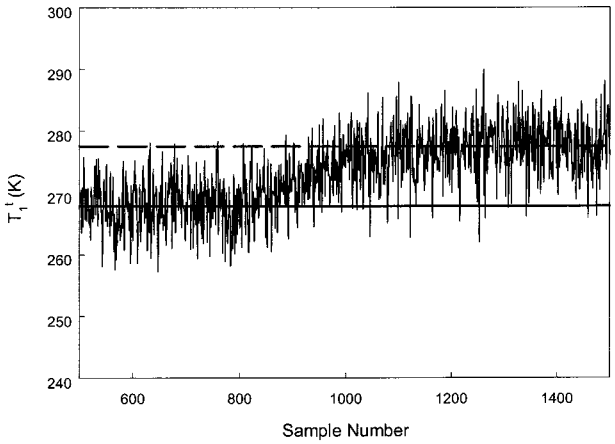

Figure 2. Simulation results of the true temperature T_1^{t} in the application example.

The other independent evaluation methods were identified with the proposed search procedure. Specifically,

$$T_1^{(2)} = \Psi_{T_1}^{(2)}(T_4,T_6,T_7;m_1,m_4,m_6,m_7) = \frac{m_4Cp_4T_4 + m_6Cp_6T_6 - m_7Cp_7T_7}{m_1Cp_1} \tag{62}$$

$$T_1^{(3)} = \Psi_{T_1}^{(3)}(T_2) = Cp_2T_2/Cp_1 \tag{63}$$

To synthesize the optimal alarm logic online, one must be able to evaluate the conditional probabilities, a_s and b_s, with eqs 40 and 41. Consequently, the parameters in the probability density function (pdf) of estimation error d_{T_1} and the joint pdf's of d_{T_1} and $d_{T_1} - \epsilon_{T_1}^{(s)}$ ($s = 1-3$) should be determined in advance. Notice that the means and variances of d_{T_1} and $\epsilon_{T_1}^{(s)}$ can be estimated according to eqs 20 and 44, respectively. As a result, the parameters of the above pdf's can be determined with the data given in Tables 1 and 2.

The effectiveness and resilience of the proposed alarm-generating strategy can be demonstrated with simulation studies. The variation in temperature T_1^{t} due to an unknown fault was first simulated. Initially, $E[T_1^{\mathrm{t}}]$ was kept at its design value, i.e., 268.0 K. The fault occurs at time $800\Delta t$, and Δt is the sampling interval. As a result, the mean temperature of arc 1 increases gradually and reaches a new steady level of 278.0 K after time $1200\Delta t$. The random number generator RONNA in IMSL was used for producing the values of $(T_1^{\mathrm{t}} - E[T_1^{\mathrm{t}}])$. The mean of this random variable is zero, and its variance is the same as $\mathrm{Var}[T_1^{\mathrm{t}}]$ (see Table 1). Finally, the values of true temperatures of arc 1 were computed by adding the $E[T_1^{\mathrm{t}}]$'s and $(T_1^{\mathrm{t}} - E[T_1^{\mathrm{t}}])$'s. The simulation data of all other temperatures and flow rates were generated in such a way that the constraint of material and energy balances is always maintained at each node in the process network. A total of 2000 sets of data have been generated in this case. Only half of them, i.e., from sample no. 500 to no. 1500, are shown in Figure 2.

Next, the measurement values were simulated. This was done by adding the measurement errors to the corresponding true temperatures and flow rates. The values of measurement errors were again created with a random number generator. Using the measurement

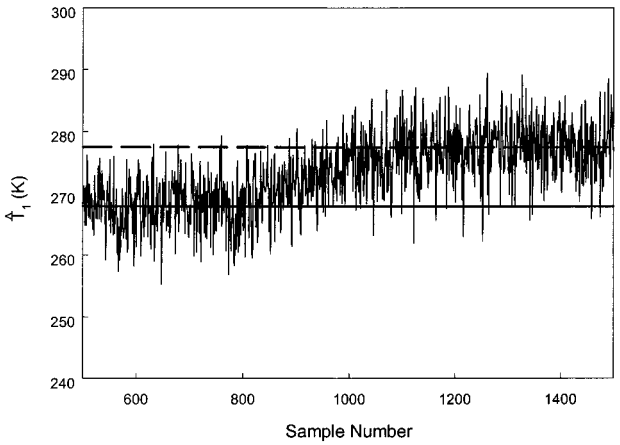

Figure 3. Simulation results of the reconciled temperature $\hat{T}_1$ in the application example.

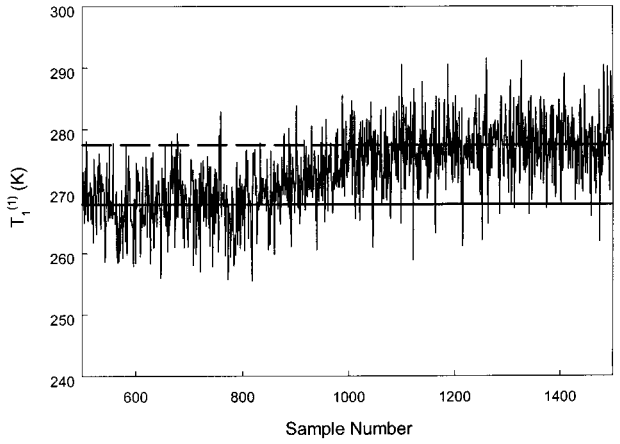

Figure 4. Simulation results of the temperature $T_1^{(1)}$ obtained with the first evaluation method.

data, one can then compute the reconciled temperatures and flow rates with eq 16. A sample of the simulation results of $\hat{T}_1$ is presented in Figure 3.

In this study, the traditional approach, i.e., using the raw temperature measurement data on arc 1, was taken first to set off the alarm. The proportions of type I and II mistakes in this case were determined to be 0.068 97 and 0.081 82, respectively. On the other hand, by adoption of the reconciled data as the basis for alarm generation, it was found that the chances of making these mistakes can be lowered significantly to 4.138% (type I) and 3.636% (type II).

Each time a new set of measurement data and the corresponding reconciled temperatures and flow rates are obtained, an optimal alarm logic can be constructed online with the proposed synthesis procedure. The time needed to construct one of these logics was found to be less than 1 s on a Pentium PC. A sample of the alarm function $f(\mathbf{y})$ for the present example is presented in Table 3. To implement the implied alarm-generation strategy, values of $T_1^{(s)}$ ($s = 1-3$) must also be determined with the three independent methods given in eqs 61−63. These values are plotted in Figures 4−6, respectively. The results of adopting the proposed alarm policy (approach B) with different C_b/C_a ratios are summarized in Table 4. In particular, the proportions of type I and II mistakes are presented in this table. For comparison purpose, the results of using only the direct measurement data (the traditional approach) and the reconciled data (approach A) are also included. From

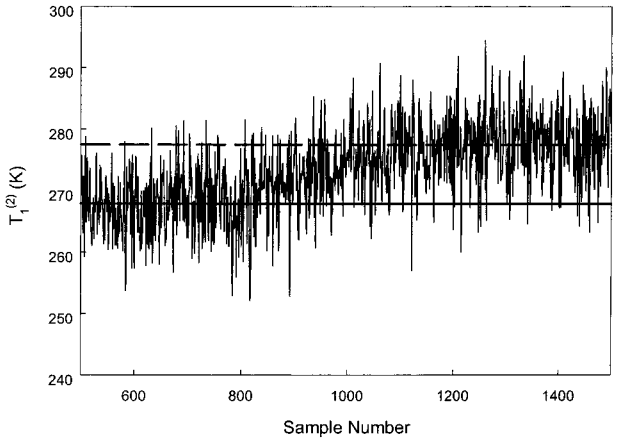

Figure 5. Simulation results of the temperature $T_1^{(2)}$ obtained with the second evaluation method.

Table 3. Optimal Alarm Functions at Different Time Intervals

sample no.	$f(0,0,0)$	$f(0,0,1)$	$f(0,1,0)$	$f(1,0,0)$	$f(0,1,1)$	$f(1,0,1)$	$f(1,1,0)$	$f(1,1,1)$
810	0	0	0	0	0	0	0	1
820	0	0	0	0	1	0	0	1
830	0	0	0	0	0	0	0	1
840	0	0	0	0	0	0	0	1
850	0	0	0	0	0	0	0	1
860	0	0	0	0	0	0	0	1
870	0	0	1	0	1	0	1	1
880	0	0	1	0	1	0	1	1
890	0	0	0	0	0	0	0	1
900	0	0	1	0	1	0	1	1
910	1	1	1	1	1	1	1	1
920	0	0	0	0	0	0	0	1
930	0	0	0	0	0	0	0	1
940	0	0	0	0	0	0	0	1
950	0	0	1	0	1	0	1	1
960	0	0	0	0	0	0	0	1
970	1	1	1	1	1	1	1	1
980	0	0	0	0	1	0	0	1
990	1	1	1	1	1	1	1	1
1000	1	1	1	1	1	1	1	1
1010	0	0	0	0	0	0	0	1
1020	0	0	0	0	0	0	0	1
1030	0	0	0	0	0	0	0	1
1040	0	0	0	0	0	0	0	1
1050	0	0	0	0	1	0	0	1
1060	1	1	1	1	1	1	1	1
1070	1	1	1	1	1	1	1	1
1080	0	0	0	0	1	0	0	1
1090	1	1	1	1	1	1	1	1
1100	1	1	1	1	1	1	1	1
1110	1	1	1	1	1	1	1	1
1120	0	0	0	0	0	0	0	1
1130	0	0	0	0	0	0	0	1
1140	0	0	1	0	1	0	1	1
1150	1	1	1	1	1	1	1	1
1160	0	0	0	0	1	0	0	1
1170	1	1	1	1	1	1	1	1
1180	0	0	0	0	0	0	0	1
1190	0	0	0	0	0	0	0	1
1200	1	1	1	1	1	1	1	1

these results, it is clear that the proposed alarm system is superior in the sense that the corresponding loss due to misjudgment reaches a minimum. Notice also that type II mistakes can be reduced to a negligible level by increasing the C_b/C_a ratio. This is usually the first priority in most cases because the purpose for installing an alarm is almost always to protect against certain catastrophic consequences.

On the other hand, it should be noted that the impact of sensor malfunctions has not yet been assessed in the above simulation studies. Because of the fact that more independent evaluation methods are adopted for alarm-generation purposes, we believe the proposed strategy should still outperform the traditional practice even

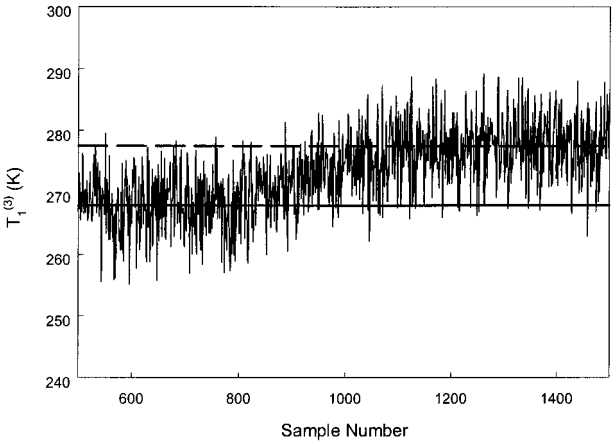

Figure 6. Simulation results of the temperature $T_1^{(3)}$ obtained with the third evaluation method.

Table 4. Performance of the Optimal Temperature Alarm Strategy When All Sensors Function Properly

		proportion of mistake	
alarm strategies	C_b/C_a	type I	type II
traditional		0.068 97	0.081 82
approach A		0.041 38	0.036 36
	10.0	0.093 10	0.018 18
	25.0	0.127 59	0.009 09
approach B	40.0	0.141 38	0.009 09
	60.0	0.141 38	0.009 09
	100.0	0.155 17	0.000 00

Table 5. Performance of the Optimal Alarm Strategy under the Influence of Sensor Malfunctions

case no.	alarm strategy	mistake type	proportion
	traditional	II	0.081 82
1	approach A	II	0.327 27
	approach B ($C_b/C_a = 25$)	II	0.081 82
	traditional	II	0.945 45
2	approach A	II	0.327 27
	approach B ($C_b/C_a = 25$)	II	0.081 82
	traditional	I	0.711 39
3	approach A	I	0.101 27
	approach B ($C_b/C_a = 25$)	I	0.154 43

under the influence of gross errors. Additional simulation studies were thus carried out to verify this assertion. The results of several typical scenarios concerning the temperature alarm on stream 1 are summarized in Table 5. A brief description of these case studies is presented in the sequel:

Case 1: The measurement value T_3 remained "normal" when the true value of alarm variable T_1^t exceeded the threshold limit.

The simulated true values of the process variables associated with Figure 2 have been adopted in this case. Here we assumed that the temperature sensor installed on arc 3 was out of order. More specifically, the mean of the measurement value T_3 stayed at 268.0 K when disturbance occurred. Naturally, the traditional approach was unaffected because the temperature sensor on stream 1 still functioned properly. On the other hand, the estimated temperature $\hat{T}_1$ should be inaccurate because of the fact that all measurements were used in the reconciliation calculation. Thus, the proportion of type II mistakes became unacceptably high in the results obtained by implementing approach A. Finally, notice that the measurement value T_3 was not used in the three independent evaluation methods, i.e., eqs 61−63. Consequently, the chance of type II mistakes was reduced to a very low level.

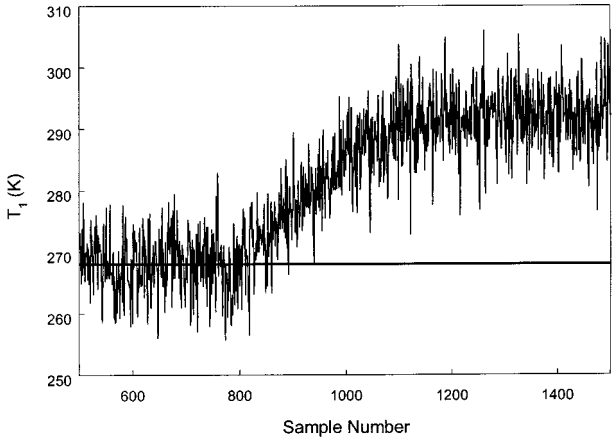

Figure 7. Simulation results of the temperature measurement T_1 due to sensor malfunction.

Case 2: The measurement value T_1 remained "normal" when the true value of alarm variable T_1^t exceeded the threshold limit.

The simulated true values of the process variables adopted in the present case were the same as before. The temperature sensor on a different arc, i.e., arc 1, was assumed to be malfunctioning. Because the direct measurement value T_1 was incorrect, the estimated temperature of stream 1, $\hat{T}_1$, became inaccurate. As a result, the alarm systems built by the first two methods, i.e., the traditional approach and approach A, performed poorly. On the other hand, notice that T_1 was used in only one of the three independent evaluation functions, i.e., $\Psi_{T_1}^{(1)}$. Consequently, type II mistakes were still significantly lowered by implementing approach B.

Case 3: The measurement value T_1 was biased when the true value of alarm variable T_1^t was still within the acceptable range.

In this case, all process variables were assumed to be within acceptable ranges. During operation, a failure in the temperature sensor on stream 1 developed and the corresponding measurement values increased significantly as shown in Figure 7. Consequently, the proportion of type I mistakes was exceedingly high in the simulation results obtained by applying the traditional approach. Notice that a large number of type I mistakes can be removed by implementing either approach A or approach B. The results associated with approach A were somewhat better. This is due to the fact that a high C_b/C_a ratio (25) was adopted in the alarm system synthesized by approach B. In other words, the main emphasis of the corresponding logic was placed upon avoiding type II errors.

Finally, case studies designed to test the flow alarm on stream 1 were also performed. The design value of the alarm variable, m_1^d, and its lower limit, m_1^L, were chosen to be 34.3 and 33.5 kg/h, respectively. Three independent evaluation methods were adopted, i.e.

$$m_1^{(1)} = m_1 \tag{64}$$

$$m_1^{(2)} = m_6 + m_8 \tag{65}$$

$$m_1^{(3)} = m_4 + m_5 \tag{66}$$

A similar procedure was followed to generate the simulation data. For the sake of brevity, a detailed

Table 6. Performance of the Optimal Flow Alarm Strategy under the Influence of Sensor Malfunctions

case no.	alarm strategy	proportion of type II mistakes
	traditional	0.936 17
1	approach A	0.851 06
	approach B ($C_b/C_a = 25$)	0.329 79
	traditional	0.255 32
2	approach A	0.553 19
	approach B ($C_b/C_a = 25$)	0.117 02

description of this procedure is not provided in this paper. Two scenarios were considered:

Case 1: The measurement values of m_1 and m_3 remain "normal" when the true value of alarm variable m_1^t exceeds the threshold limit.

Case 2: The measurement values of m_3 remain "normal" when the true value of alarm variable m_1^t exceeds the threshold limit.

The results of these case studies are presented in Table 6. One can see clearly that approach B performs better than the other two methods in all scenarios.

Conclusions

From the above discussions, it is clear that the proposed alarm-logic design strategy is indeed superior to any of the existing techniques. The resulting alarm system is not only optimal but also resilient. These desirable features are brought about mainly by integrating the intrinsic characteristics of process network, i.e., the hardware and spatial redundancy, into system design. Furthermore, from experiences obtained in solving the example problem, one can also conclude that the demand for online computation is reasonable especially when the sampling interval is in the range of seconds or longer.

Acknowledgment

This work is supported by the National Science Council of the ROC government under Grant NSC88-2214-E006-013.

Appendix: Algorithm II

The detailed steps in algorithm II are described in the sequel:

1. Let temperature be the *current variable* and arc *a* be the *current arc*.

2. Consider the original process network. Merge the input and output nodes of every arc on which the current variable is unmeasured. Let the resulting graph be the *current digraph* and also $i = 1$.

3. Find a cut set $\mathbf{K}_T^{(i)}$ of the current digraph which contains the current arc.

4. Merge the input and output nodes of every arc in $\mathbf{K}_T^{(i)}$ except those of the current arc. Let the result be the current digraph and $i = i + 1$.

5. Repeat steps 3 and 4 until the current arc itself forms a loop. Let $N_T = i - 1$.

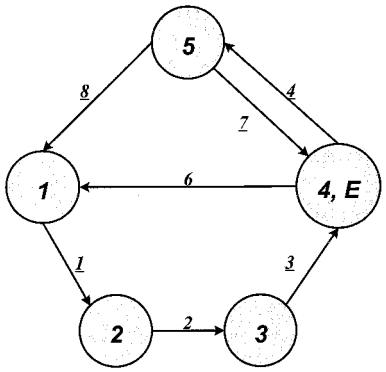

Figure 8. Example to illustrate the procedure for identifying independent temperature evaluation methods: the current digraph obtained in step 2.

6. Let $i = 1$ and $j = 1$. Let flow rate be the current variable.

7. Consider the original process network again. Merge the input and output nodes of every arc in $\mathbf{K}_T^{(k)}$ ($k = 1, 2, ..., i - 1, i + 1, ..., N_T$) and in $\mathbf{K}_M^{(p,q)}$ ($p = 1, 2, ..., i - 1$ and $q = 1, 2, ...$). Then merge the input and output nodes of the arcs without flow sensors. Let the result be the current digraph.

8. If $i > 2$ and $j = 1$, let arc *a* be the current arc and go to step 10.

9. Select an arc in $\mathbf{K}_T^{(i)}$ on which the flow sensor is not installed. Let it be the current arc.

10. Recover the current arc from the current digraph; i.e., expand the corresponding merged nodes to their original configuration. Let the result be the new current graph.

11. Identify from the current digraph a cut set $\mathbf{K}_M^{(i,j)}$ that contains the current arc.

12. Merge the input and output nodes of every arc in $\mathbf{K}_M^{(i,j)}$. Let the result be the current digraph and $j = j + 1$.

13. Repeat steps 9−12 until all arcs with unmeasured flow rate in $\mathbf{K}_T^{(i)}$ are exhausted.

14. Let $i = i + 1$ and $j = 1$. Repeat steps 7−13 until $i = N_T$.

The following example is prepared to facilitate understanding of the above procedure:

Example. The ammonia process is presented in Figure 1. It is assumed in this example that all process variables are measured except the temperature of arc 5 and the flow rates of arcs 4 and 6. Let us also assume that the temperature of arc 1 is the alarm variable in this problem.

First of all, one can obviously determine this temperature with the temperature sensor installed on arc 1, i.e.

$$T_1^{(1)} = \Psi_{T_1}^{(1)}(T_1) = T_1 \qquad \text{(A1)}$$

where T_1 is the measurement value of T_1^t and $\Psi_{T_1}^{(1)}$ denotes an evaluation function which represents the first method to evaluate the temperature of arc 1. On the other hand, several independent *indirect* methods can be identified with the following steps:

Step 1: Let the temperature be the current variable and arc 1 be the current arc.

Step 2: Merge the input and output nodes of arc 5 to form the current digraph in Figure 8. Let $i = 1$.

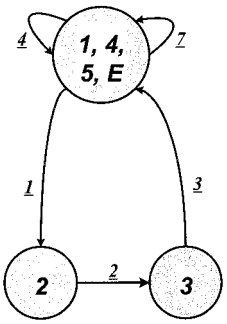

Figure 9. Example to illustrate the procedure for identifying independent temperature evaluation methods: the first current digraph obtained in step 4.

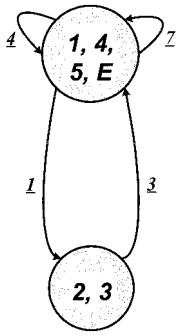

Figure 10. Example to illustrate the procedure for identifying independent temperature evaluation methods: the second current digraph obtained in step 4.

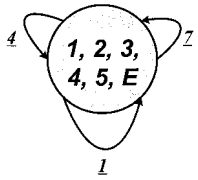

Figure 11. Example to illustrate the procedure for identifying independent temperature evaluation methods: the third current digraph obtained in step 4.

Step 3: Identify a cut set from Figure 8 which contains arc 1, i.e., $\mathbf{K}_T^{(1)} = \{1, 6, 8\}$.

Step 4: Merge the input and output nodes of every arc in $\mathbf{K}_T^{(1)}$ except those of arc 1. Let the result be the current digraph (see Figure 9) and $i = 2$.

Step 5:

(a) Repeat step 3 and find the cut set $\mathbf{K}_T^{(2)} = \{1, 2\}$.

(b) Repeat step 4 to form a new current digraph (Figure 10). Let $i = 3$.

Step 5:

(a) Repeat step 3 and find the cut set $\mathbf{K}_T^{(3)}$.

(b) Repeat step 4 to form a new current digraph (Figure 11). Let $i = 4$.

Because arc 1 itself forms a loop in Figure 11, step 6 should be executed next. The value of N_T should be 3.

Step 6: Let $i = 1$, $j = 1$, and the flow rate be the current variable.

Step 7: Merge the input and output nodes of arcs 1−4 and 6 to form the current digraph presented in Figure 12.

Step 8: Because $i = 1$ and $j = 1$, the present step should be skipped.

Step 9: Because the flow rate of arc 6 in $\mathbf{K}_T^{(1)}$ is not measured, assign it to be the current arc.

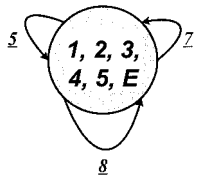

Figure 12. Example to illustrate the procedure for identifying independent temperature evaluation methods: the first current digraph obtained in step 7.

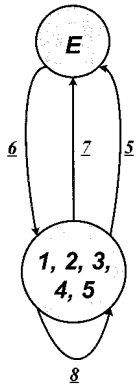

Figure 13. Example to illustrate the procedure for identifying independent temperature evaluation methods: the first current digraph obtained in step 10.

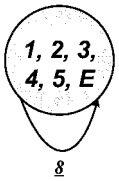

Figure 14. Example to illustrate the procedure for identifying independent temperature evaluation methods: the current digraph obtained in step 12.

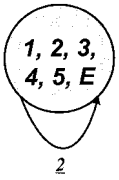

Figure 15. Example to illustrate the procedure for identifying independent temperature evaluation methods: the second current digraph obtained in step 7.

Step 10: Recover arc 6 to form the current digraph in Figure 13.

Step 11: Identify a cut set from Figure 13, i.e., $\mathbf{K}_M^{(1,1)}$.

Step 12: Merge the input and output nodes of every arc in $\mathbf{K}_M^{(1,1)}$. Let the result be the current digraph (Figure 14) and $j = 2$.

Step 13: Because there are no other unmeasured arcs in $\mathbf{K}_T^{(1)}$, step 14 should be carried out next.

Step 14: $i = 2$ and $j = 1$.

(a) Repeat step 7. Merge arcs 1 and 3−8 to form Figure 15.

(b) Repeat step 8. Arc 1 should be the current arc.

(c) Repeat step 10. Recover arc 1 to form the current digraph in Figure 16.

(d) Repeat steps 11 and 12. A cut set can be identified: $\mathbf{K}_M^{(2,1)} = \{1, 2\}$.

Step 12: $i = 3$ and $j = 1$.

(a) Repeat step 7. Merge arcs 1, 2, and 4−8 to form Figure 17.

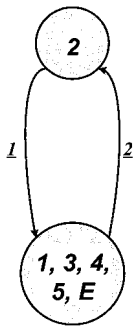

Figure 16. Example to illustrate the procedure for identifying independent temperature evaluation methods: the second current digraph obtained in step 10.

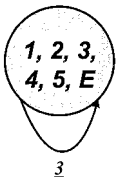

Figure 17. Example to illustrate the procedure for identifying independent temperature evaluation methods: the third current digraph obtained in step 7.

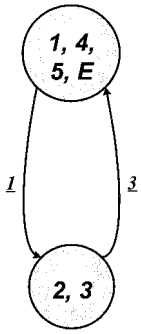

Figure 18. Example to illustrate the procedure for identifying independent temperature evaluation methods: the third current digraph obtained in step 10.

(b) Repeat step 8. Arc 1 should be the current arc.

(c) Repeat step 10. Recover arc 1 to form the current digraph in Figure 18.

(d) Repeat steps 11 and 12. A cut set can be identified: $\mathbf{K}_{\mathrm{M}}^{(2,1)} = \{1, 3\}$.

Because $i = N_{\mathrm{T}} = 3$, the procedure can be terminated.

From cut sets $\mathbf{K}_{\mathrm{T}}^{(1)}$ and $\mathbf{K}_{\mathrm{M}}^{(1,1)}$, one can obtain the second method for computing T_1^{t}, i.e.

$$T_1^{(2)} = \Psi_{T_1}^{(2)}(T_6,T_8;m_1,m_5,m_7,m_8) = T_0 + \frac{1}{m_1 Cp_1}[(m_5 + m_7)Cp_6(T_6 - T_0) + m_8 Cp_8(T_8 - T_0)] \quad \text{(A2)}$$

where T_0 is a reference temperature. In addition, from the sets $\mathbf{K}_{\mathrm{T}}^{(2)}$, $\mathbf{K}_{\mathrm{M}}^{(2,1)}$, $\mathbf{K}_{\mathrm{T}}^{(3)}$, and $\mathbf{K}_{\mathrm{M}}^{(3,1)}$, the third and fourth formulas can be identified:

$$T_1^{(3)} = \Psi_{T_1}^{(3)}(T_2;m_2) = T_0 + \frac{m_2 Cp_2}{m_2 Cp_1}(T_2 - T_0) = T_0 + \frac{Cp_2}{Cp_1}(T_2 - T_0) \quad \text{(A3)}$$

$$T_1^{(4)} = \Psi_{T_1}^{(4)}(T_3;m_3) = T_0 + \frac{m_3 Cp_3}{m_3 Cp_1}(T_3 - T_0) = T_0 + \frac{Cp_3}{Cp_1}(T_3 - T_0) \quad \text{(A4)}$$

Literature Cited

(1) Kresta, J. V.; MacGregor, J. F.; Marlin, T. E. Multivariate Statistical Monitoring of Process Operating Performance. *Can. J. Chem. Eng.* **1991**, *69*, 35.

(2) Mah, R. S. H. *Chemical Process Structures and Information Flows*; Butterworth: Boston, 1990.

(3) Lees, F. P. *Loss Prevention in the Process Industries*, Vol. 1; Butterworth: London, 1980; p 379.

(4) Mah, R. S. H.; Stanley, G. M.; Downing, D. M. Reconciliation and Rectification of Process Flow and Inventory Data. *Ind. Eng. Chem., Process Des. Dev.* **1976**, *15*, 175.

(5) Tsai, C. S.; Chang, C. T. Optimal Alarm Logic Design for Mass Flow Networks. *AIChE J.* **1997**, *43*, 3021.

(6) Inoue, K.; Kohda, T.; Kumamoto, H.; Takami, I. Optimal Structures of Sensor Systems with Two Failure Modes. *IEEE Trans. Reliab.* **1982**, *R-31*, Apr, 119.

(7) Kohda, T.; Kumamoto, H.; Inoue, K. Optimal Shut Down Logic for Protective Systems. *IEEE Trans. Reliab* **1983**, *R-32*, Apr, 26.

(8) Crowe, C. M.; Garcia Campos, Y. A.; Hrymak, A. Reconciliation of Process Flow Rates by Matrix Projection: I. The Linear Case. *AIChE J.* **1983**, *29*, 881.

(9) Crowe, C. M. Reconciliation of Process Flow Rates by Matrix Projection: II. The Nonlinear Case. *AIChE J.* **1986**, *32*, 616.

(10) Ali, Y.; Narasimhan, S. Sensor Network Design for Maximizing Reliability of Linear Processes. *AIChE J.* **1993**, *39*, 820.

(11) Ali, Y.; Narasimhan, S. Redundant Sensor Network Design for Linear Processes. *AIChE J.* **1995**, *41*, 2237.

(12) Chu, S. K. Alarm System Design for Multicomponent Process Networks. M.S. Thesis, National Cheng Kung University, Tainan, Taiwan, 1999.

(13) Kretsovalis, A.; Mah, R. S. H. Observability and Redundancy Classification in Generalized Process Networks: I. Theorems. *Comput. Chem. Eng.* **1988**, *12*, 671.

Received for review February 28, 2000
Accepted September 8, 2000

IE0002780