Estimating State Probability Distributions from Noisy and Corrupted Data

Lloyd P. M. Johnston
Dept. of Chemical Engineering, University of New South Wales, Sydney, 2052, Australia

Mark A. Kramer
Gensym Corporation, 125 Cambridge Park Drive, Cambridge, MA 02140

The method of recursive state density estimation (RSDE) is developed for determining the probability distribution of the states of a system from measurements that contain both random noise and gross errors. The technique is based on the expectation maximization algorithm and is iterative in nature. Similar to EM, at each iteration the likelihood of the distribution estimated by the RSDE algorithm is guaranteed to increase, thus arriving at the most likely distribution of the true states, given the measurement data set and the algorithm initial conditions. Convergence of the algorithm to the correct solution, for a simple case where an analytical answer can be derived for comparison, is shown. Two chemical process examples that have more complex distributions are also shown. Once the probability distribution of the states has been determined, many monitoring and statistical process and quality control functions can be performed using the more accurate distributions of the process states, avoiding corruption of the distribution due to faulty measurements.

Introduction

In chemical and manufacturing plants, numerous measurements of state variables are made to obtain a picture of what is happening in the plant. The measurement systems used to observe the process introduce "small" random errors when operating normally (measurement noise) and can introduce "large" systematic errors when a failure of the measurement system occurs (gross errors). Historical plant data contain both types of errors. If we wish to derive the probability distribution of the true states from historical data, a way to remove these errors is required. Such a probability distribution can be used for various tasks, including statistical process control (Owen, 1989), quality control (Montgomery, 1985), and more recently data rectification (Johnston and Kramer, 1995). If reasonable assurances can be made that the fundamental processes that have led to the historical probability distribution of the states is unchanged, then the probability distribution of the states can be used for future data rectification, statistical process control, quality control, modeling, and prediction.

Casting the model of the plant in a probabilistic framework facilitates the use of all the standard tools of statistical and Bayesian analysis to make sound probabilistic statements about the validity of the solutions to the various problems being solved. This includes deciding when the system has deviated from the fundamental processes that were driving the historical distribution of the states, resulting in a change in the underlying probability distribution of the states. This ability to detect when the historical model is no longer valid is extremely important, as it warns the user that the results obtained from the model are unlikely to be correct.

Any parametric or nonparametric density estimation technique allows for the calculation of the probability density function for a given set of data. This article focuses on the estimation of the probability density function of the true plant states, given a data set that includes measurement errors. This leads to a dilemma, in that all that is available are the corrupted measurements, but what is required is the probability distribution of the states of the measured variables. The goal of this article is to introduce a technique in which the probability distribution of the states can be estimated from the noisy and corrupted historical measurement sets. We use expecta-

Correspondence concerning this article should be addressed to L. P. M. Johnston.

tion maximization (EM) (Dempster et al., 1977) to link probability density function estimation with data rectification to arrive at the probability distribution of the states using the measurements.

Probability Density Function Estimation

There are many methods for finding the probability distribution function for a set of data. The methods fall into two basic classes, parametric and nonparametric. Parametric methods assume a form for the probability distribution, and the parameters of the distribution are fit to the data. The most frequently used parametric model is the Gaussian distribution, with the mean vector and covariance matrix calculated from the data using maximum likelihood estimates or the method of moments (Ross, 1987). Nonparametric distributions are structured such that the data themselves determine the form of the probability distribution model (Devroye, 1984). A common nonparametric probability density estimator is a mixture of Gaussian distributions (Everitt and Hand, 1981). A particular case of this type of estimator is Parzen windows (Parzen, 1962), which centers a Gaussian distribution at each point in the "training" data set. The probability distribution of the data is calculated as the average of all the Gaussian values:

$$P\{\boldsymbol{x}\} = \frac{1}{K} \sum_{k=1}^{K} \psi(\boldsymbol{x} - \boldsymbol{m}_k, \sigma^2 \boldsymbol{I}), \tag{1}$$

where ψ is a Gaussian distribution. The center for each Parzen window, $\boldsymbol{m}_k$, is coincident with the kth point in a training data set. The width of each window is set by the adjustable parameter σ, which is the same for all Gaussian units, and $\sigma^2 \boldsymbol{I}$ is the diagonal covariance matrix for the Gaussian distribution, which is the window function.

Other nonparametric density estimators that are more efficient than Parzen window estimators, in a computational sense, use a smaller number of Gaussian basis functions to cover the data set, rather than putting an identical Gaussian distribution at each data point. Such density estimators are classed as mixture models (Everitt and Hand, 1981). The calculation of the necessary coefficients, such as mean vectors and covariance matrices for each basis function in mixture models, is much more difficult for nonparametric estimators than for parametric estimators (e.g., a single Gaussian distribution) because there are many more adjustable parameters, including parameters that determine the structure of the model. Johnston and Kramer (1994) present an elliptical basis function (EBF) estimator that has some advantages over other mixture model methods.

The EBF estimator is composed of H elliptical units, and the density is given by

$$P\{\boldsymbol{x}\} = \sum_{h=1}^{H} \rho_h a_h(\boldsymbol{x}), \tag{2}$$

where ρ_h is the local density estimate associated with unit h and a_h is the activation of unit h

$$a_h(\boldsymbol{x}) = \exp\left[-(\boldsymbol{x} - \boldsymbol{m}_h)^T \boldsymbol{Q}_h^{-1} (\boldsymbol{x} - \boldsymbol{m}_h)\right]. \tag{3}$$

The calculation of ρ_h, $\boldsymbol{m}_h$, and $\boldsymbol{Q}_h$ from a training data set and the optimal choice of the number of units H are detailed in Johnston and Kramer (1994).

The EBF estimator is similar to a weighted sum of Gaussian shaped units, each with its own covariance matrix $\boldsymbol{Q}_h$. Everitt and Hand (1981) use the EM algorithm to find the means and covariances of a distribution composed of a mixture of Gaussian units. The density estimator is

$$P\{\boldsymbol{x}\} = \sum_{i=1}^{N} q_i \psi_i(\boldsymbol{x}), \tag{4}$$

where $\psi_i(\boldsymbol{x})$ is a Gaussian distribution with parameters mean $\boldsymbol{m}_i$ and covariance $\boldsymbol{Q}_i$, and q_i is the weighting value for the ith Gaussian. In this application, their goal was to fit the mixture of Gaussians to distributions that were truly mixtures of Gaussian distributions and the number of Gaussians was known *a priori*. In this case each $\boldsymbol{x}$ vector in the training set belonged to one of the underlying Gaussians, and thus once the membership of a point is established the distribution is linear in the sum of the $\boldsymbol{x}$ values and the sum of the squared $\boldsymbol{x}$ values. This results because once it is known which distribution the $\boldsymbol{x}$ data vector was drawn from, the problem reduces to finding the parameters for a single Gaussian distribution from all of the data vectors that belong to it. Everitt and Hand (1981) use the EM algorithm to iteratively estimate the membership function of the data and the parameters of the individual Gaussian units until convergence is reached. They report good results for cases where the underlying distribution is a mixture of Gaussians that are well spaced apart. In this case the membership function is very clear. The results deteriorate and poor fits to the underlying distribution result when the underlying distribution is a mixture of Gaussians that have a high degree of overlap. In these cases it is much more difficult to conclusively assign a data vector to a particular Gaussian unit, and the fit of the mixture model to the underlying distribution suffers because of this.

In using the EBF estimator to fit a probability distribution to the underlying data, we are not assuming that the true underlying distribution is a mixture of Gaussians. The EBF estimator is used because it gives good nonparametric density estimates that fit a wide variety of data sets regardless of the underlying distribution. Parzen windows achieve the same general fit property as EBF estimators, but the EBF estimators overcome some of the spiky and erroneously discontinuous distributions that can occur with Parzen windows, and they are more computationally efficient because fewer basis functions are used. Because it is not assumed that the data are truly being drawn from an underlying mixture of Gaussian distributions, there is no longer a 0/1 membership function that breaks the mixture of Gaussians (or Gaussian-type units in the EBF case) down into estimating the parameters of a single Gaussian a number of times, as in the Everitt and Hand application. Thus the EBF method can accurately estimate distributions even when there is a high degree of overlap between the basis functions.

Data Rectification

The goal of data rectification is to find the true state of the plant by removing both random noise and gross errors from

the measured data. Most research on data rectification has been in the specific area of data reconciliation. Data rectification literally means to find the true values of the plant states, whereas data reconciliation implies an adjustment to the measured values to conform to model constraints, such as mass and energy balances. The usual assumption is that the measurement errors follow a Gaussian distribution, which leads to least squares objective function (Mah, 1987):

$$\min_{\hat{x}} (y-\hat{x})^T Q^{-1}(y-\hat{x})$$
$$\text{s.t.} \quad h(\hat{x}) = 0$$
$$g(\hat{x}) \le 0, \tag{5}$$

where y are the measurements, $\hat{x}$ are the estimated plant states, and Q is the measurement noise covariance matrix. For the case of independent linear constraints the problem is formulated as

$$\min_{\hat{x}} (y-\hat{x})^T Q^{-1}(y-\hat{x})$$
$$\text{s.t.} \quad A\hat{x} = 0, \tag{6}$$

which has the closed form solution (Kuehn and Davidson, 1961):

$$x = [I - QA^T(AQA^T)^{-1}A]y. \tag{7}$$

When the constraints are not linear then there is no closed-form solution to Eq. 5, and a numerical optimization is required to find the solution.

There is the possibility that sensors will fail and that gross errors, along with the random noise errors characterized by Q, will be present in the measured data. A large number of auxiliary gross-error detection and identification schemes have been developed for systems with linear constraints to remove these gross errors from the data set (Hlavacek, 1977; Mah and Tamhane, 1982; Tamhane and Mah, 1985; Rosenberg et al., 1987). The general procedure of these tests is to use the measurements and the model constraints to form a test statistic. Then a hypothesis test is performed to conclude if the estimated errors in the measurements are likely to have come from the Gaussian distribution characterized by Q. If not, then it is concluded that a gross error exists and a procedure, such as serial elimination (Nogita, 1972), is used to identify which measurement contains the gross error. The measurement with the gross error is removed from the data set and the reduced data set is reconciled using the standard data-reconciliation scheme. More recently, Tjoa and Biegler (1991) and Johnston and Kramer (1995) use a compound Gaussian objective function instead of the least-squares objective function, which consolidates the gross error detection, removal, and data rectification tasks into one step.

Johnston and Kramer (1995) formulated the data-rectification problem in a probabilistic framework and showed that traditional data reconciliation yields the maximum likelihood states of the plant only when two restrictive assumptions are made. The first assumption is that the errors must be normally distributed and no gross errors can be present in the data set. The second assumption is that all the feasible states

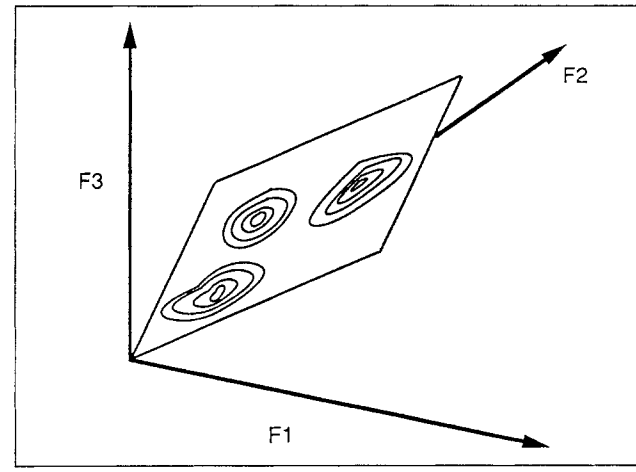

Figure 1. Plant state subspace for a mixing tee where $F1+F2=F3$.

Historical data from the process forms probability contours on the feasible plane, implying that not all points on the rectification surface are equally likely.

of the plant, that is, those satisfying known constraints, are equally likely. That is, the probability distribution of the plant states is assumed to be 1 if the model constraints are satisfied and 0 otherwise. This is termed the binary assumption. In reality this is not true. A plot of the plant operating states over its history would reveal that they do not uniformly cover the state subspace, but preferentially lie in certain areas. Consider the case of a mixing-T. The constraint is that $F_1 + F_2 = F_3$, which defines a plane. A plot of the historical plant state values of F_1, F_2, and F_3 would all lie in the plane, but would not uniformly cover the plane. The states would partially cover the plane, as is shown by the probability contours in Figure 1. The probability distribution of the plant states, $P\{x\}$, describes how the states cover the plane.

Johnston and Kramer (1995) proposed the maximum likelihood rectification (MLR) method for performing data rectification. The method maximizes the probability of the plant states given the measurements. By applying Bayes theorem, the MLR objective function is:

$$\max_x P\{x \mid y\} = \max_x P\{y \mid x\} \frac{P\{x\}}{P\{y\}}, \tag{8}$$

which reduces to

$$\text{MLR} = \max_x P\{y \mid x\} P\{x\} \tag{9}$$

since $P\{y\}$ is a normalizing factor that does not depend on x.

The measurement, y, of the state, x, can be generally represented by

$$y = x + \delta, \tag{10}$$

where δ is the error introduced in the measurement process. In general δ can be a function of x, but for additive errors the $P\{y \mid x\}$ term in the MLR is $P\{\delta\}$:

$$P\{y \mid x\} = P\{\delta\}. \tag{11}$$

The $P\{\delta\}$ term is the probability distribution of the measurement error. If $P\{x\}$ is modeled with the binary assumption that $P\{x\} = 1$ if the constraints are satisfied and $P\{x\} = 0$ otherwise, and $P\{\delta\}$ is modeled as a Gaussian distribution, then the MLR reduces to the constrained least-squares data-reconciliation formulation in Eq. 6.

To remove the assumption that the measurement errors are normally distributed and allow for gross errors in the distribution, a compound distribution of two Gaussians has been used by Tjoa and Biegler (1991) and Johnston and Kramer (1995). A "narrow" Gaussian represents a normally operating sensors noise, and a "broad" Gaussian represents the failed modes of a sensor, such as miscalibration or a sensor malfunction. The distribution of the measurement error for the ith measurement is

$$P\{\delta_i\} = (1 - p_i)\frac{1}{\sqrt{2\pi}\sigma_i}\exp\left(-\frac{(x_i - y_i)^2}{2\sigma_i^2}\right) + p_i\frac{1}{\sqrt{2\pi}b_i\sigma_i}\exp\left(-\frac{(x_i - y_i)^2}{2b_i^2\sigma_i^2}\right), \quad (12)$$

where σ_i is the standard deviation of the normal noise on the ith sensor; b_i is the ratio of the standard deviation of the broad Gaussian repesenting the gross errors to the standard deviation of the normal noise for sensor i; and p_i is the probability that sensor i will contain a gross error. Both Tjoa and Biegler and Johnston and Kramer found that the solution to the data-rectification problem was robust to a fairly wide range of choices of p_i and b_i. Assuming that the errors are uncorrelated,

$$P\{\boldsymbol{\delta}\} = \prod_{i=1}^{n} P\{\delta_i\}. \quad (13)$$

In addition to the distribution of $P\{\boldsymbol{\delta}\}$ the probability distribution of the states $P\{\boldsymbol{x}\}$ is required to find the maximum likelihood estimate of the state $\boldsymbol{x}$ given the measurements. If we had a data set of the plant states, any one of the previously discussed density estimation methods could be used to fit a probability distribution to the data set, thus finding $P\{\boldsymbol{x}\}$. A database of historical plant states is not available; only a set of plant measurements, which are corrupted by sensor noise and sensor gross errors, is available for use in the estimation of $P\{\boldsymbol{x}\}$.

Expectation Maximization

The expectation maximization (EM) algorithm (Dempster, 1977) is a broadly applicable algorithm for computing maximum likely estimates from incomplete data. In many applications a large number of data vectors are collected for modeling and analysis purposes. Invariably in these large data sets some of the data vectors have "holes" in them, for reasons of a failed sensor, clerical omission, or other data-collection error. The rest of the data vector does hold valuable information that would be lost if the entire data vector were thrown out because of the missing measurements. But to use standard modeling and analysis tools, a value must be supplied for the missing value. EM is used to calculate the most likely estimate of the relevant parameters, such as regression coefficients, means, variances, and other parameters of interest depending on the application, in spite of the missing data.

The EM algorithm consists of an expectation step, the E-step, and a maximization step, the M-step, which are iteratively repeated until convergence. If the model were known, then the missing values could be calculated by finding their expected values given the known data and the model (the E-step). Conversely, if the data set were complete, then well-known complete data techniques for finding the most likely parameter values could be used to find the parameters of the model (M-step). After an initial estimate of the missing values, the EM algorithm iteratively goes through the E- and M-steps until convergence is reached. In the E-step the sufficient statistics of the missing data are estimated, as opposed to the missing data themselves (Little and Rubin, 1987). Stable convergence is achieved as the M-step finds parameters that monotonically increase the likelihood of the parameters given the data at each iteration. This monotonic convergence ensures that the maximum likelihood estimate of the parameters given the data, and initial starting point of the algorithm, are arrived at (Meng and Rubin, 1994). The EM algorithm has been used in a broad range of applications, including estimates of parameters of density functions when there are missing data (Redner and Walker, 1984; Meng and Rubin, 1991; Yuille et al., 1994), in evaluating synaptic transmissions (Stricker and Redman, 1994) as well as in expert systems (Jordan and Jacobs, 1994). In this application we use the EM algorithm to link data rectification and probability density estimation steps to arrive at the distribution of the plant states from the noisy and corrupted measurements.

Recursive State Density Estimation

The proposed application of the EM algorithm is different from previous applications in that in this case the "missing data" are the true states of the plant, not sporadic missing measurements in a data set. We call the application of the EM algorithm for finding the density estimate of the states from noisy and corrupted measurements recursive-state density estimation (RSDE). The best estimate of the true states of the plant at the beginning of the RSDE procedure are the measurements themselves. The probability distribution $P\{\boldsymbol{x}\}$ is found by fitting an appropriate distribution, parametric or nonparametric, to the $\boldsymbol{y}$ data by estimating the sufficient statistics of $P\{\boldsymbol{x}\}$ with $\boldsymbol{x} = \boldsymbol{y}$. This is the M-step. With an estimate of $P\{\boldsymbol{x}\}$ the expected value of the sufficient statistics of the "missing data" (the $\boldsymbol{x}$ data) can be calculated. This is the E-step. The M-step and E-step are then alternated until the convergence of the distribution is obtained. In the examples section, a simple example with a known analytical answer is used to illustrate the procedure and demonstrate its convergence.

The sufficient statistics of the missing data will depend on the form of the distribution used for estimating $P\{\boldsymbol{x}\}$. For example, if a simple one-dimensional Gaussian is used, the sufficient statistics are the sum of the x values and the sum of squared x values. In the general case of process data the distribution is often a multidimensional, multimodal distribution and its form is usually unknown. Nonparametric density estimators are capable of capturing the more complex distri-

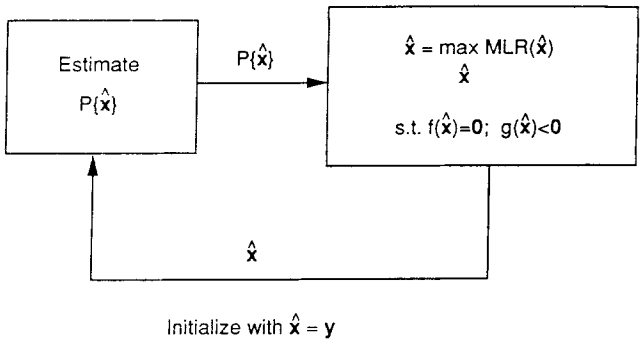

Figure 2. RSDE algorithm.

butions, but these distributions depend on the data themselves for their form. Thus the sufficient statistics of the nonparametric density estimators are the x data themselves.

Casting the problem of estimating the distribution of the x data from the measurements y in terms of the EM algorithm, and using a nonparametric density estimator, leads to the following RSDE algorithm, which is graphically shown in Figure 2:

1. Set initial estimate of the x data equal to the measurements y
2. Estimate $P\{x\}$ using an appropriate method
3. Use the MLR approach in Eq. 9 to find the maximum likely new x values given the measurements and the current estimate of $P\{x\}$
4. Using the new x values from 3 go to step 2 until convergence is reached

Convergence is reached when none of the x values changes upon subsequent iterations of the algorithm. An efficient solution method for solving the MLR problem in step 3 is given in Johnston (1996).

The RSDE methodology applies to any form of parametric or nonparametric density estimator that is appropriate for the given data set. If a parametric distribution for x is used, then the sufficient statistics of the x data, and not the x data themselves, need to be calculated in step 3. Through the MLR formulation, the problem can be implemented with any distribution of the measurement error including ones that allow for sensor failure, making it broadly applicable. Application of the algorithm is shown in the examples section.

Incorporating Linear Constraints

In the data-rectification step of the RSDE algorithm, the MLR is solved subject to known constraints. If linear constraints are imposed on the rectification, then there will be an exact linear correlation among the rectified values that are then to be used to find $P\{x\}$ in the next iteration. If mixture models with Gaussian shaped basis functions are used to model $P\{x\}$, then the exact linear correlation in the data will cause a singularity in the covariance matrix of the Gaussian basis functions. This will make it impossible to find the inverse of the covariance matrix that is needed to find the density estimate. Although the data are observed in n dimensions, the data are truly only $(n\text{-}m)$ dimensional, where m is the number of independent linear model constraints. One way to avoid this singularity problem is to estimate $P\{x\}$ in n dimensions and not enforce the linear equality constraints in the MLR solution during the RSDE algorithm. The linear relationships are "rediscovered" in $P\{x\}$, as the Gaussian distributions will approximate the linear relationships by constructing very thin discs as substitutes for the plane that represent the linearly constrained data. The analytical linear equality constraints are then enforced when the derived $P\{x\}$ distribution is used to rectify future data.

The singularity problem is avoided, and estimation of $P\{x\}$ is improved, if the density estimation is done only in the subspace of the linearly independent dimensions. For a system constrained by m independent linear equations that satisfy the equation $\boldsymbol{Ax} = \mathbf{0}$, a probability distribution is constructed in the linear independent subspace defined by the m by n constraint matrix A,

$$P\{x\} = f(\boldsymbol{Px}), \tag{14}$$

where f is the probability density estimation function. The matrix $\boldsymbol{P}$ is a projection matrix that maps x in n-space to a corresponding vector in m-space. The n-dimensional vector x is recovered by multiplying the m-dimensional vector by $\boldsymbol{P}^T$. The $(n\text{-}m)$-by-n projection matrix $\boldsymbol{P}$ is formed by finding $(n\text{-}m)$ orthonormal n-dimensional basis vectors that span the null space of plane described by $\boldsymbol{Ax} = \mathbf{0}$ (Brown, 1991). Each row of $\boldsymbol{P}$ corresponds to a basis vector. In general, the projection of an n-dimensional vector $\boldsymbol{u}$ to an m-dimensional vector $\boldsymbol{v}$ by multiplying by $\boldsymbol{P}$, then projection of $\boldsymbol{v}$ back to n-space by multiplication by $\boldsymbol{P}^T$, does not result in the recovery of $\boldsymbol{u}$. That is, for

$$\boldsymbol{v} = \boldsymbol{Pu} \quad \text{and} \quad \boldsymbol{u}' = \boldsymbol{P}^T\boldsymbol{v} \tag{15}$$

$\boldsymbol{u}'$ will not equal $\boldsymbol{u}$ generally. This is because the projection matrix $\boldsymbol{P}$ projects all vectors to the plane $\boldsymbol{Ax} = \mathbf{0}$. For the case where $\boldsymbol{Au} = \mathbf{0}$, then the transformation in Eq. 15 will result in $\boldsymbol{u} = \boldsymbol{u}'$, because if the vector already lies on the plane, then the projection does not move the vector but only changes its basis from n dimensions to m dimensions. Thus for all valid rectified states (i.e., those for which $\boldsymbol{Ax} = \mathbf{0}$) the matrix projection does not move $\boldsymbol{x}$ but only temporarily changes its dimension for calculation of $\boldsymbol{P}\{x\}$ in a lower dimension than $\boldsymbol{x}$.

Performance Measures

Three performance measures: log probability (J value), log likelihood, and mean squared error (MSE) are used to assess the performance of the RSDE algorithm in the Examples section. The most important measure is the J value, as it measures the goodness of fit of the calculated probability density function. The log likelihood and MSE measures should monotonically increase and monotonically decrease, respectively, at subsequent iterations of the RSDE algorithm.

Log probability (J value)

The measure of how well a particular density estimator approximates a density function is the product of the probabilities of sample points independently and randomly drawn from the true underlying distribution (Traven, 1991). For a test set drawn from the true underlying distribution of the states of size n_t, the overall fit measure is given by

$$J' = \prod_{i=1}^{n_t} P\{x_i\}. \tag{16}$$

For convenience of calculation, usually the normalized log of Eq. 16 is used:

$$J = \frac{1}{n_t}\log(J') = \frac{1}{n_t}\sum_{i=1}^{n_t}\log(P\{x_i\}). \tag{17}$$

For each of the process examples, a training set of the states x is simulated and the measurements y are created by adding noise and possibly gross errors to the x data. A probability distribution density function estimator is then fit to the x training data, the raw y training data, and by the RSDE method for each of the three examples. The J value for a separate 1,000-point test set of x data is calculated for each estimator. The largest possible J value for an estimator is that of the estimator calculated from the x training data because we cannot expect to produce an estimator that is any better than one that is fit to the true plant states. We also want the J value of the estimator to be better than the J value of the estimator fit to the raw y data. Thus the range of J values for a good estimator is between the J value of the estimator fit to the y data (on the low side) and the J value of the estimator fit to the x data (on the high side). A good estimator will be closer to the J value of the estimator fit to the x data. The examples given are all from simulated chemical plants, and thus the true plant states are known and an estimator fit to the true plant states can be generated for comparison purposes. In practice the true plant states are never known and a method to estimate the distribution of the true plant states from the measurement data, such as the proposed RSDE technique, needs to be used to find an estimate of the true plant states.

Log likelihood

The likelihood of the rectified state for an individual measurement in the training set is the product of the probability of the rectified state $P\{x_k\}$ and the probability the adjustment made to the measurement $P\{\delta_k\}$:

$$L'_k(x_k, y_k) = P\{x_k\}P\{\delta_k\}. \tag{18}$$

For the entire training set the likelihood is the product of the individual likelihoods:

$$L'(X,Y) = \prod_{k=1}^{K} P\{x_k\}P\{\delta_k\}, \tag{19}$$

where X and Y are K by n matrices containing the K individual x and y row vectors. For convenience usually the normalized log of Eq. 19 is used:

$$L(X,Y) = \frac{1}{K}\sum_{k=1}^{K}[\log(P\{x_k\}) + \log(P\{\delta_k\})]. \tag{20}$$

In the EM algorithm the log likelihood function being maximized, in this case Eq. 20, is guaranteed to monotonically increase until convergence (Dempster, 1977). Monotonic increase of the log likelihood function at each iteration of the RSDE algorithm is shown for each of the examples.

Mean squared error

One application of the probability distribution of the plant states estimated by the RSDE methodology is data rectification (Johnston and Kramer, 1995), and data rectification is one of the steps of the RSDE algorithm. Thus the RSDE algorithm estimates the true plant states for the training set of data used to calibrate $P\{x\}$, along with the desired probability density function. A measure of the accuracy of the rectified states is the mean squared error (MSE)

$$\text{MSE} = \frac{1}{N^*K}\sum_{k=1}^{K}\sum_{n=1}^{N}\frac{(\hat{x}_{n,k} - x_{n,k})^2}{\sigma_n^2}, \tag{21}$$

where N is the dimension of the measurement vector, K is the number of measurement vectors used to calculate the MSE, $\hat{x}$ is the rectified state, x is the true plant state, and σ_n is the standard deviation of the Gaussian noise on sensor n. The MSE is structured such that if there are no sensor failures (i.e., the measurements only contain random Gaussian noise with standard deviations σ_n), and the data is not rectified (that is, $\hat{x} = y$), then the MSE = 1.0. If there are gross errors, the MSE can be greater than 1.0 for the unrectified data. If the rectification is perfect (i.e., the true state of the plant was found for every measurement vector), then the MSE = 0.0. Thus the goal is to drive the MSE toward zero. It is expected that the MSE will monotonically decrease at each RSDE iteration, because we expect each subsequent RSDE iteration to have a more accurate $P\{x\}$, which should lead to more accurate rectified values and this would be reflected by a lower MSE. Plots of the MSE as a function of the RSDE iteration showing the monotonic decrease are shown for the two chemical engineering system examples.

Examples

The RSDE methodology was first applied to a simple one-dimensional distribution of states where the parameters of the state distribution can be analytically derived from the parameters of the measurement distribution, which can be easily calculated using standard techniques. The RSDE algorithm is then applied to find the probability distribution of the states for two chemical engineering processes from noisy and corrupt measurements.

Example 1

The distribution of the states of a one-dimensional system are known to follow a Gaussian distribution with unknown mean μ and unknown variance σ_x. The system is observable through measurements that are recorded as y and are related to x by

$$y = x + \epsilon, \tag{22}$$

where ϵ is a normally distributed random variable with mean zero and known variance σ. Further, ϵ is independent of x. Given the set of y data, the mean and variance of y can be

calculated using standard techniques. From the y data, the distribution of the x data can be analytically calculated. The mean of x is equal to the mean of y, and the variance of x is equal to the variance of y minus the variance of ϵ (σ^2). This is because the distribution of y is formed by the sum of two independent Gaussian random variables, and thus the mean of y is the sum of the mean of x and mean of ϵ, and the variance of y is the sum of the variance of x and the variance of ϵ. This same result can be achieved by applying the principles of EM in the RSDE algorithm. The advantage of using the RSDE algorithm is that it is not only applicable to this simple case, but it is also applicable to more complex systems where the deconvolution of the distribution of x from the distribution of y is extremely difficult or impossible to do.

To apply the EM algorithm, the complete data set is considered to be both x and y. The "missing" data in the EM sense are the x values. The joint distribution of x and y, given the mean and variance of x, is

$$f(x_i, y_i \mid \mu, \sigma_x) = f(y_i \mid x_i, \mu, \sigma_x) f(x_i \mid \mu, \sigma_x) \tag{23}$$

$$f(x_i, y_i \mid \mu, \sigma_x) = \frac{\beta_i}{\sqrt{2\pi}\,\sigma_T} \exp\left[-\frac{1}{2} \frac{\left(x_i - \dfrac{\sigma_x^2 y_i + \sigma^2 \mu}{\sigma_x^2 + \sigma^2} \right)}{\sigma_T^2} \right]. \tag{24}$$

Thus the joint distribution for the complete data set of n measurements is

$$f(\boldsymbol{X}, \boldsymbol{Y} \mid \mu, \sigma_x) = \prod_{i=1}^{n} \left[\frac{\beta_i}{\sqrt{2\pi}\,\sigma_T} \exp\left[-\frac{1}{2} \frac{\left(x_i - \dfrac{\sigma_x^2 y_i + \sigma^2 \mu}{\sigma_x^2 + \sigma^2} \right)^2}{\sigma_T^2} \right] \right], \tag{25}$$

which simplifies to

$$f(\boldsymbol{X}, \boldsymbol{Y} \mid \mu, \sigma_x) = \exp\left[-\frac{1}{2} \sum_{i=1}^{n} \frac{\left(x_i - \dfrac{\sigma_x^2 y_i + \sigma^2 \mu}{\sigma_x^2 + \sigma^2} \right)^2}{\sigma_T^2} \right] \prod_{j=1}^{n} \left[\frac{\beta_j}{\sqrt{2\pi}\,\sigma_T} \right], \tag{26}$$

where,

$$\beta_i = \frac{1}{\sqrt{2\pi}\sqrt{\sigma_x^2 + \sigma^2}} \exp\left[-\frac{1}{2} \frac{\sigma_x^2 y_i^2 + \sigma^2 \mu^2}{\sigma_x^2 \sigma^2} \right] \times \exp\left[-\frac{1}{2} \left(\frac{\sigma_x^2 + \sigma^2}{\sigma_x^2 \sigma^2} \right) \left(\frac{\sigma_x^2 y_i + \sigma^2 \mu}{\sigma_x^2 + \sigma^2} \right)^2 \right] \tag{27}$$

and

$$\sigma_T = \frac{\sigma_x \sigma}{\sqrt{\sigma_x^2 + \sigma^2}}. \tag{28}$$

In the EM algorithm the missing data themselves are not estimated, but the sufficient statistics of the missing data are estimated. The sufficient statistics for the x data are the sum of the x values and the sum of the square of the x values, as the joint distribution is linear in these two quantities. Thus we want to estimate:

$$E\left[\sum_{i=1}^{n} (x_i \mid y_i, \mu, \sigma_x) \right] = \sum_{i=1}^{n} E[(x_i \mid y_i, \mu, \sigma_x)] \tag{29}$$

and

$$E\left[\sum_{i=1}^{n} (x_i^2 \mid y_i, \mu, \sigma_x) \right] = \sum_{i=1}^{n} E[(x_i^2 \mid y_i, \mu, \sigma_x)]. \tag{30}$$

For the ith measurement, the expected value of the corresponding state and state squared are

$$E[(x_i \mid y_i, \mu, \sigma_x)] = \left(\frac{\sigma_x^2 y_i + \sigma^2 \mu}{\sigma_x^2 + \sigma^2} \right) \tag{31}$$

and

$$E[(x_i^2 \mid y_i, \mu, \sigma_x)] = \sigma_T + \left(\frac{\sigma_x^2 y_i + \sigma^2 \mu}{\sigma_x^2 + \sigma^2} \right)^2, \tag{32}$$

where we have used the fact that

$$\mathrm{Var}(x_i) = E[x_i^2] - (E[x_i])^2; \tag{33}$$

thus

$$E[x_i^2] = \sigma_T^2 + (E[x_i])^2. \tag{34}$$

Therefore, the expected values of the relevant sums are

$$E\left[\sum_{i=1}^{n} (x_i \mid y_i, \mu, \sigma_x) \right] = \sum_{i=1}^{n} \left(\frac{\sigma_x^2 y_i + \sigma^2 \mu}{\sigma_x^2 + \sigma^2} \right) \tag{35}$$

and

$$E\left[\sum_{i=1}^{n} (x_i^2 \mid y_i, \mu, \sigma_x) \right] = n\sigma_T + \sum_{i=1}^{n} \left(\left(\frac{\sigma_x^2 y_i + \sigma^2 \mu}{\sigma_x^2 + \sigma^2} \right)^2 \right). \tag{36}$$

This completes the E-step. The M-step calculates the usual maximum likelihood estimates of the mean and variance given the sufficient statistics. Given the values of μ and σ_x from the tth iteration, the values of μ and σ_x at the $(t+1)$th iteration are found by

$$\mu^{(t+1)} = \frac{1}{n} E\left[\sum_{i=1}^{n} (x_i \mid y_i, \mu^{(t)}, \sigma_x^{(t)})\right] = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\sigma_x^{(t)^2} y_i + \sigma^2 \mu^{(t)}}{\sigma_x^{(t)^2} + \sigma^2}\right), \quad (37)$$

which converges to

$$\hat{\mu} = \frac{\sum_{i=1}^{n} y_i}{n} \quad (38)$$

and

$$\sigma_x^{(t+1)} = \frac{1}{n}\left(E\left[\sum_{i=1}^{n} (x_i^2 \mid y_i, \mu^{(t)}, \sigma_x^{(t)})\right] - \left(E\left[\sum_{i=1}^{n} (x_i \mid y_i, \mu^{(t)}, \sigma_x^{(t)})\right]\right)^2\right), \quad (39)$$

which simplifies to

$$\sigma_x^{(t+1)^2} = \frac{\sigma_x^{(t)^2}\sigma^2}{\sigma_x^{(t)^2} + \sigma^2} + \frac{1}{n}\sum_{i=1}^{n}\left(\left(\frac{\sigma_x^{(t)^2} y_i + \sigma^2 \mu^{(t)}}{\sigma_x^{(t)^2} + \sigma^2}\right)^2\right) - \frac{1}{n^2}\left(\sum_{i=1}^{n}\left(\frac{\sigma_x^{(t)^2} y_i + \sigma^2 \mu^{(t)}}{\sigma_x^{(t)^2} + \sigma^2}\right)\right)^2 \quad (40)$$

and converges to

$$\hat{\sigma}_x^2 = \sigma_y^2 + \sigma^2. \quad (41)$$

The details of the convergence of Eq. 37 to Eq. 38 and of Eq. 40 to Eq. 41 are given in the Appendix. This shows that the correct result for the mean and variance of x given the y data are achieved using the EM algorithm. This demonstrates that the RSDE algorithm, which casts the x values as "missing" data and then applies the principles of EM, converges to the maximum likely result for the mean and variance of the state variables given the measurements. The RSDE algorithm can now be applied to more complex systems, where analytical results, as in the one-dimensional Gaussian distribution case, are not readily available.

Linearly constrained flow network

Figure 3 shows the flow network from Mah (1987). The total flow of each stream is measured. The linear plant constraints are $A\mathbf{x} = 0$, where A is composed of the three independent mass balances that constrain the process:

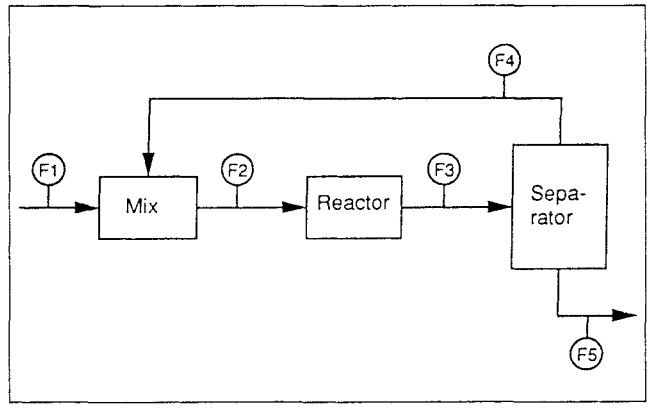

Figure 3. Flow network.

$$A = \begin{bmatrix} 1 & -1 & 0 & 1 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 \end{bmatrix}$$

and $\boldsymbol{x}$ is a 5-by-1 vector containing the five flow rates. The plant was assumed to operate at a low and high flow rate with probabilities 0.45 and 0.55, respectively. At the low flow rate F_1 was drawn from $N(100, 10)$ distribution, and at the high flow rate F_1 was drawn from $N(150, 10)$ distribution. The ratio of F_4 to F_3 was drawn from a $N(0.6, 0.05)$ distribution. Having randomly selected F_1 and the separator split fraction, the remaining flows were calculated, and the y values were generated by adding random, zero-centered Gaussian noise with standard deviations 1.0, 4.0, 4.0, 3.0, and 1.0, respectively, to the five sensors. A 100-point data set was generated for calibration of $P\{\boldsymbol{x}\}$, and a separate 1,000-point data set was generated for testing the accuracy of the fit distributions. The cross-validation methodology of Johnston and Kramer (1994) was used to select the best density estimator, which was an EBF estimator with four basis functions.

The RSDE methodology was applied with $P\{\boldsymbol{x}\}$ fit in the linearly independent two-dimensional subspace of the flow network. The projection matrix $\boldsymbol{P}$ is found by finding the null space of A, and is

$$\boldsymbol{P} = \begin{bmatrix} -0.6124 & -0.2041 & -0.2041 & 0.4082 & -0.6124 \\ 0 & 0.5714 & 0.5714 & 0.5174 & 0 \end{bmatrix}.$$

In addition to the RSDE estimator, a density estimator was fit to the generated $\boldsymbol{x}$ data and to the raw y data. To fit an estimator to the y data it was first projected to the constraint plane by multiplication by $\boldsymbol{P}$. Because y does not lie on the constraint plane, the raw y data vector cannot be recovered by $\boldsymbol{P}^T\boldsymbol{P}y$, as it can for the $\boldsymbol{x}$ values that do satisfy the constraints. Thus the projection to the plane reconciles but does not completely rectify the y data. The J value for each estimator is summarized in Table 1. The RSDE estimator's J value of -7.47 was closer to the maximum expected J value for the estimator fit to the $\boldsymbol{x}$ data of -7.45 than the y estimator J value of -7.62.

Table 1. Log Probability Values for PDF Estimated from the True State Variables, the Measurements, and by the RSDE Algorithm for the Three Examples

PDF Estimated from	Flow Network		Heat-Exchanger Network	
	Noise Only	10% Gross Errors	Noise Only	10% Gross Errors
x data	−7.45	−7.45	−27.38	−27.38
y data	−7.62	−7.74	−31.37	−32.16
RSDE	−7.47	−7.48	−28.73	−29.61

Figure 4 shows that the log likelihood monotonically increases with RSDE iteration as expected. The raw y data do not satisfy the constraints and thus $P\{x\}$ at $x = y$ (the initial condition for the RSDE algorithm) is 0. Therefore the log likelihood at the start of the algorithm (iteration 0) is undefined and is not plotted. Figure 5 shows the monotonic decrease of the MSE from 1 and 0.32. Thus at each RSDE iteration more accurate estimates of the true plant states are being found.

The RSDE algorithm was also applied to a data set that was corrupted with gross errors in the F1 sensor. A gross error ranging from −30 to +30 was added to 10% of the F1 measurements in the 100-point training data set. The J value for the RSDE estimator was −7.48, which was essentially the same as the J value of the RSDE estimator fit to the data without the gross error, and very close to the J value of −7.45 for the estimator fit to the x data. The estimator fit to the raw y data with the gross errors was −7.74. Thus the RSDE algorithm was able to effectively reject the gross errors and find a good estimate of $P\{x\}$. This was also shown by the MSE being reduced from 2.72 to 0.40.

The RSDE methodology was then applied by fitting $P\{x\}$ in five dimensions without forcing the known analytical constraints to be obeyed in the data-rectification step. It is not suggested that $P\{x\}$ be estimated in the full dimension of the measurements when there are known analytical linear model constraints, but this example is shown to demonstrate how

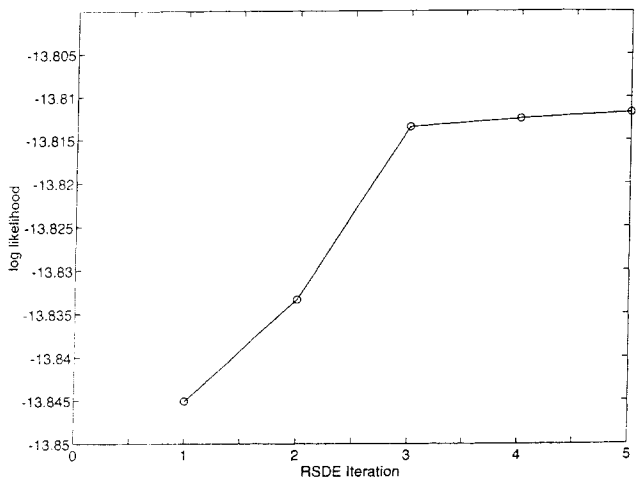

Figure 4. Monotonic increase of the log likelihood function during iterations of the RSDE algorithm for the flow-network data set with no gross errors.

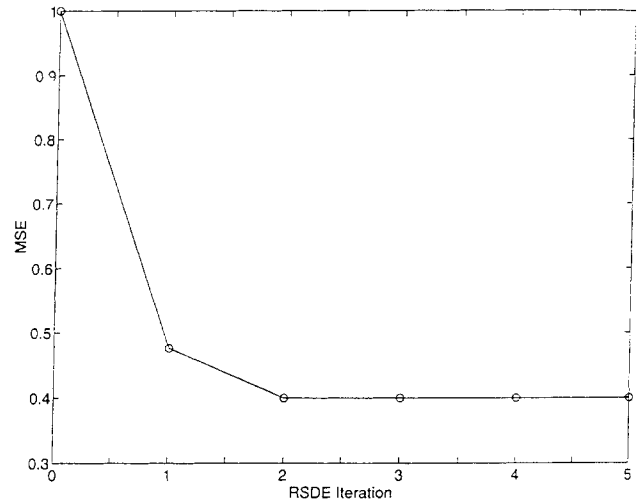

Figure 5. Monotonic decrease of the MSE value during iterations of the RSDE algorithm for the flow-network data set with no gross errors.

"unknown" process correlations can be captured by $P\{x\}$. In this case the linear correlations in the data are "discovered" by the RSDE methodology and incorporated into $P\{x\}$. Figure 6 shows the log likelihood at each RSDE iteration and it monotonically increases. The log likelihood does not level off because the estimated plant states are almost perfectly linearly correlated after six iterations, and thus the covariance matrices of the basis functions are nearly singular. Small changes in the estimated plant states, increasing the linear correlation among them, cause the covariance matrices to become closer to singular, and thus the $P\{x\}$ value drastically rises between iterations, causing the log likelihood to continually increase. If the RSDE algorithm is continued, eventually the estimated plant states become linearly correlated enough to cause the basis function covariance matrices to be-

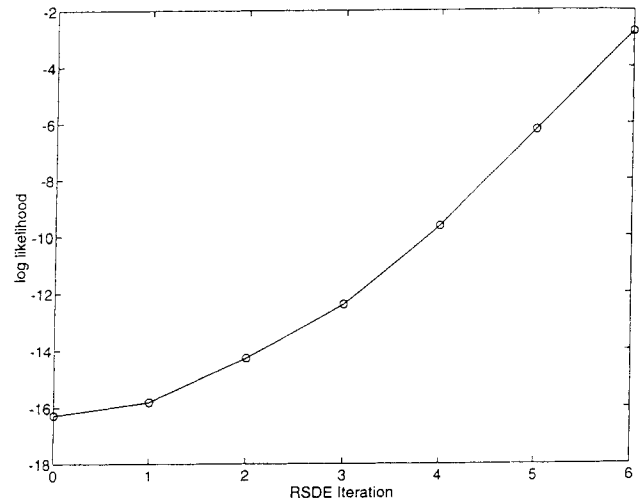

Figure 6. Monotonic increase of the log likelihood function during iterations of the RSDE algorithm for the flow-network data set with no gross errors and with $P\{x\}$ estimated in five dimensions.

come singular, and the algorithm will terminate, as the $P\{x\}$ function can no longer be calculated. Estimating $P\{x\}$ in the linearly independent subspace, as shown previously, eliminates this problem. The MSE was reduced from 1 to 0.37 during the RSDE algorithm, showing that the $P\{x\}$ distribution was capturing the process data correlation without enforcement of the linear constraints. In this manner the $P\{x\}$ distribution will also capture all correlations in the data and then exploit these correlations in the data-rectification process.

Heat-exchanger network

Figure 7 shows a heat-exchanger network that was adapted from Tjoa and Biegler (1991). Process stream A is heated by process stream B and utility streams C and D. The flow of utility stream D is adjusted to maintain an outlet temperature of stream A (TA8) of 615°C. The instrumentation of the network is such that mass and energy balances cannot be completely written for the system. Table 2 shows the operating conditions of the network. Full details of the process simulation can be found in Johnston and Kramer (1995). The RSDE technique was applied to a data set that consisted of 120 points, and an EBF density estimator with 6 units was used to estimate the probability density function. A separate 1,000-point test data set was generated to calculate the J values of density estimators calculated from the x data, raw y data, and by the RSDE algorithm.

The J values for each estimator are summarized in Table 1. The J value for the RSDE estimator was -28.73, which is closer to the J value of -27.38 for the estimator fit to the x data than the J value of -31.37 for the estimator fit to the y data. Thus the implicit process relationships were captured and a fairly accurate $P\{x\}$ was found using the RSDE algorithm. As in the previous example, the log likelihood increased monotonically with each iteration of the RSDE methodology, and thus the maximum likelihood distribution of the states given the measurements was achieved. The MSE decreased monotonically from 1.05 to 0.62, indicating that at each subsequent iteration the estimate of the states was closer to the true states. The data rectification in this case was done without mathematical constraints because the instrumentation on the network did not allow for mass and energy balance to be written. Nonetheless the data rectification was

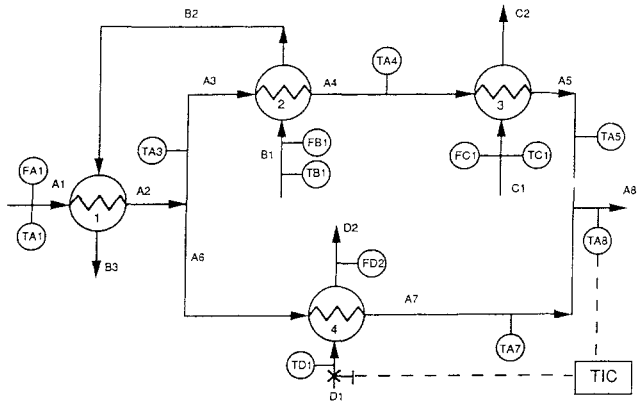

Figure 7. Heat-exchanger network.

Table 2. Operating Conditions for the Heat-Exchanger Network

FC1 = $N(240, 5)$	Probability = .3
TA1 = $N(450, 5)$	FA1 = $N(800, 5)$
TB1 = $N(620, 5)$	FB1 = $N(200, 5)$
TC1 = $N(670, 5)$	Probability = .4
TD1 = $N(690, 5)$	FA1 = $N(900, 5)$
FA3/FA2 = .42	FB1 = $N(250, 5)$
	Probability = .3
	FA1 = $N(990, 5)$
	FB1 = $N(320, 5)$

quite accurate, and thus we have a good deal of confidence in the estimated probability distribution of the plant states.

Another data set for the heat-exchanger network was created with 120 points. For this data set a gross error between -10 and $+10$°C was present in 10% of the measurements of TA4. The J value for the estimator calculated by the RSDE algorithm was -29.61, which was larger than the J value calculated from the raw y data containing the gross errors of -32.16. The log likelihood monotonically increased throughout all iterations of the RSDE methodology, although the starting and final log likelihood values were lower than for the data set containing only noise because of the gross errors present. The gross errors also caused the starting MSE to be approximately 2.27, as opposed to the 1.05 starting value for the data set with no gross errors, but the RSDE methodology still reduced the MSE for the data set with gross errors to the same 0.62 level that was observed for the data set with no gross errors.

During the iterative RSDE process, the gross errors are rejected as being unlikely because they lie in regions of low probability. For an individual measurement vector, the sensor containing the gross error is drawn toward the correct value because by making a large adjustment in the erroneous sensor and small adjustments in the other sensors, which do not contain gross errors, the rectified state is moved into a more likely region of $P\{x\}$ than the measurement. This adjustment lowers the $P\{\delta\}$ contribution to the objective function value (Eq. 9), but increases the $P\{x\}$ contribution to the objective function value. The net result is an increase in the objective function value, and thus the log likelihood value monotonically increases during the algorithm. The RSDE methodology only makes adjustments that are probable, given the information at hand. Figure 8 shows the measurements and the final rectified values plotted against the true values for sensor TA4, the sensor with the gross error. The end result is that gross errors are effectively rejected and the rectified values obtained are close to the true values. This results in the probability distribution estimated by the RSDE methodology being very close to the true probability distribution of the states.

Conclusions

The RSDE methodology is presented as an application of the EM algorithm for finding the maximum likely distribution of the true plant states, given the noisy and corrupted measurements. As an application of the EM algorithm, the convergence properties of the RSDE methodology follow that of the EM algorithm, and thus the maximum likely distribution

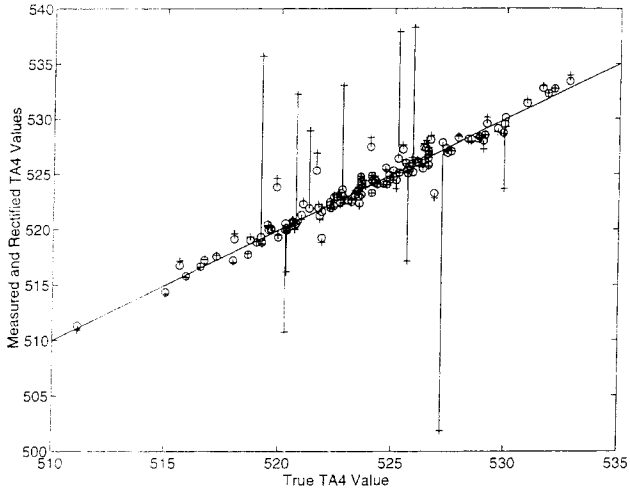

Figure 8. Measurement (+) and final rectified value (○) of TA4 against the true value of TA4 for the data set in which 10% of the TA4 measurements were corrupted by gross errors.

of the states given the measurements and the algorithm initial conditions is ensured. The RSDE methodology was shown to converge to the correct analytic result for a simple one-dimensional Gaussian distribution. The RSDE algorithm was also successfully applied to the more complex distributions of two simulated chemical processes, specifically a flow network and a heat-exchanger network. In all cases the RSDE methodology performed well, even when the data set was corrupted with 10% gross errors. For the examples shown, the probability density functions obtained using the RSDE algorithm were better estimators of the true state probability distribution than a probability density function estimator fit to the raw measurement data, as measured by the log probability of an independent test data set. For a linear-flow network, the RSDE methodology was applied in both the full space of the measurements and in the linearly independent subspace of the processes, a distribution that accurately rectified the data was found in both cases. The RSDE methodology yields the probability distribution of the true plant states, that is most likely given the measurement information, and this distribution can then be applied to safety, quality control, and statistical process control tasks. Using the distribution of the true plant states, and not the distribution of the corrupted measurements, will improve the performance of each of these tasks.

Literature Cited

Brown, W. C., *Matrices and Vector Spaces*, Dekker, New York (1991).

Dempster, A. P., N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *J. Roy. Statist. Soc. B*, **39**, 1 (1977).

Devroye, L., *Nonparametric Density Estimation: The L View*, Wiley, New York (1984).

Everitt, B. S., and D. J. Hand, *Finite Mixture Distributions*, Chapman & Hall, London (1981).

Hlavacek, V., "Analysis of Complex Plant—Steady-State and Transient Behavior," *Comput. Chem. Eng.*, **1**, 75 (1977).

Johnston, L. P. M., and M. A. Kramer, "Probability Density Estimation Using Elliptical Basis Functions," *AIChE J.*, **40**, 1639 (1994).

Johnston, L. P. M., and M. A. Kramer, "Maximum Likelihood Data Rectification: Steady State Systems," *AIChE J.*, **41**, 2415 (1995).

Johnston, L. P. M., "Probability Based Approaches to Process Data Modeling and Rectification," PhD Thesis, Massachusetts Institute of Technology, Cambridge (1996).

Jordan, M. I., and R. A. Jacobs, "Hierarchical Mixtures of Experts and the EM Algorithm," *Neural Comput.*, **6**, 181 (1994).

Kuehn, D. R., and H. Davidson, "Computer Control II: Mathematics of Control," *Chem. Eng. Prog.*, **57**, 44 (1961).

Little, R. J. A., and D. B. Rubin, *Statistical Analysis with Missing Data*, Wiley, New York (1987).

Mah, R. S. H., "Data Screening," *Foundations of Computer Aided Process Operations*, G. V. Reklaitus and H. D. Spriggs, eds., Elsevier, New York (1987).

Mah, R. S. H., and A. C. Tamhane, "Detection of Gross Errors in Process Data," *AIChE J.*, **28**, 828 (1982).

Meng, X., and D. B. Rubin, "Using EM to Obtain Asymptotic Variance-Covariance Matrices: The SEM Algorithm," *J. Amer. Stat. Assoc.*, **86**, 899 (1991).

Meng, X., and D. B. Rubin, "On the Global and Componentwise Rates of Convergence of the EM Algorithm," *Linear Algebra Appl.*, **199**, 413 (1994).

Montgomery, D. C., *Introduction to Statistical Quality Control*, Wiley, New York (1985).

Nogita, S., "Statistical Test and Adjustment of Process Data," *Ind. Eng. Chem. Proc. Des. Dev.*, **11**, 197 (1972).

Owen, M., *SPC and Continuous Improvement*, Springer-Verlag, New York (1989).

Parzen, E., "On Estimation of Probability Density Function and Mode," *Ann. Math. Stat.*, **33**, 1065 (1962).

Redner, R. A., and H. F. Walker, "Mixture Densities, Maximum Likelihood and the EM Algorithm," *SIAM Rev.*, **26**, 195 (1984).

Rosenberg, J., R. S. H. Mah, and C. Iordache, "Evaluation of Schemes for Detecting and Identifying Gross Errors in Process Data," *Ind. Eng. Chem. Res.*, **26**, 555 (1987).

Ross, S. M., *Introduction to Probability and Statistics for Engineers and Scientists*, Wiley, Toronto (1987).

Stricker, C., and S. Redman, "Statistical Models of Synaptic Transmission Evaluated Using the Expectation-Maximization Algorithm," *Biophys. J.*, **67**, 656 (1994).

Tamhane, A. C., and R. S. H. Mah, "Data Reconciliation and Gross Error Detection in Chemical Process Networks," *Technometrics*, **27**, 409 (1985).

Tjoa, I. B., and L. T. Biegler, "Simultaneous Strategies for Data Reconciliation and Gross Error Detection of Nonlinear Systems," *Comput. Chem. Eng.*, **15**, 679 (1991).

Traven, H. G. C., "A Neural Network Approach to Statistical Pattern Classification by 'Semiparametric' Estimation of Probability Density Functions," *IEEE Trans. Neural Networks*, **2**, 366 (1991).

Yuille, A. L., P. Stolorz, and J. Utans, "Statistical Physics, Mixtures of Distributions, and the EM Algorithm," *Neural Comput.*, **6**, 334 (1994).

Appendix

Proof of the convergence of mean and variance in Example 1 to the expected correct results. For the mean of the states, the update equation is (Eq. 37):

$$\mu^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} \frac{\sigma_x^{(t)^2} y_i + \sigma^2 \mu^{(t)}}{\sigma^2 + \sigma_x^{(t)^2}} \tag{A1}$$

$$\mu^{(t+1)} = \frac{\sigma_x^{(t)^2}}{\sigma^2 + \sigma_x^{(t)^2}} \sum_{i=1}^{n} \frac{y_i}{n} + \frac{\sigma^2}{\sigma^2 + \sigma_x^{(t)^2}} \mu^{(t)} \tag{A2}$$

$$\frac{\sigma^2 + \sigma_x^{(t)^2}}{\sigma_x^{(t)^2}} \mu^{(t+1)} - \frac{\sigma^2}{\sigma_x^{(t)^2}} \mu^{(t)} = \sum_{i=1}^{n} \frac{y_i}{n}. \tag{A3}$$

At convergence $\mu^{(t+1)} = \mu^{(t)} = \hat{\mu}$

$$\frac{\sigma^2+\sigma_x^{(t)^2}-\sigma^2}{\sigma_x^{(t)^2}}\hat{\mu}=\sum_{i=1}^{n}\frac{y_i}{n}. \tag{A4}$$

Therefore,

$$\hat{\mu}=\sum_{i=1}^{n}\frac{y_i}{n}, \tag{A5}$$

the expected result for the problem posed in example 1 (Eq. 38).

For the variance of the states, the update equation is (Eq. 40):

$$\sigma_x^{(t+1)^2}=\frac{\sigma^2\sigma_x^{(t)^2}}{\sigma^2+\sigma_x^{(t)^2}}+\frac{1}{n}\sum_{i=1}^{n}\left(\left(\frac{\sigma_x^{(t)^2}y_i+\sigma^2\mu^{(t)}}{\sigma^2+\sigma_x^{(t)^2}}\right)^2\right)-\left(\frac{1}{n}\sum_{i=1}^{n}\frac{\sigma_x^{(t)^2}y_i+\sigma^2\mu^{(t)}}{\sigma^2+\sigma_x^{(t)^2}}\right)^2. \tag{A6}$$

Simplifying the last two terms in Eq. A6:

$$\frac{1}{n}\sum_{i=1}^{n}\left(\left(\frac{\sigma_x^{(t)^2}y_i+\sigma^2\mu^{(t)}}{\sigma^2+\sigma_x^{(t)^2}}\right)^2\right)=\frac{1}{\left(\sigma^2+\sigma_x^{(t)^2}\right)^2}\times\left[\frac{\sigma_x^{(t)^4}}{n}\left(\sum_{i=1}^{n}y_i^2\right)+2\sigma_x^{(t)^2}\sigma^2\mu^{(t)}\left(\sum_{i=1}^{n}\frac{y_i}{n}\right)+\sigma^4\mu^{(t)^2}\right] \tag{A7}$$

$$\left(\frac{1}{n}\sum_{i=1}^{n}\frac{\sigma_x^{(t)^2}y_i+\sigma^2\mu^{(t)}}{\sigma^2+\sigma_x^{(t)^2}}\right)^2=\frac{1}{\left(\sigma^2+\sigma_x^{(t)^2}\right)^2}\times\left[\sigma_x^{(t)^4}\left(\sum_{i=1}^{n}\frac{y_i}{n}\right)^2+2\sigma_x^{(t)^2}\sigma^2\mu^{(t)}\left(\sum_{i=1}^{n}\frac{y_i}{n}\right)+\sigma^4\mu^{(t)^2}\right]. \tag{A8}$$

Substituting Eqs. A7 and A8 into Eq. A6 and simplifying, leads to

$$\sigma_x^{(t+1)^2}=\frac{\sigma^2\sigma_x^{(t)^2}}{\sigma^2+\sigma_x^{(t)^2}}+\frac{\sigma_x^{(t)^4}}{\left(\sigma^2+\sigma_x^{(t)^2}\right)^2}\left[\frac{1}{n}\left(\sum_{i=1}^{n}y_i^2\right)-\left(\sum_{i=1}^{n}\frac{y_i}{n}\right)^2\right]. \tag{A9}$$

Noting that,

$$\left[\frac{1}{n}\left(\sum_{i=1}^{n}y_i^2\right)-\left(\sum_{i=1}^{n}\frac{y_i}{n}\right)^2\right]=E[y_i^2]-(E[y_i])^2=\mathrm{Var}(y)=\sigma_y^2 \tag{A10}$$

simplifies Eq. A9 to

$$\sigma_x^{(t+1)^2}=\frac{\sigma^2\sigma_x^{(t)^2}}{\sigma^2+\sigma_x^{(t)^2}}+\frac{\sigma_x^{(t)^4}}{\left(\sigma^2+\sigma_x^{(t)^2}\right)^2}\sigma_y^2. \tag{A11}$$

Further simplification yields

$$\sigma_x^{(t+1)^2}=\frac{\sigma^4\sigma_x^{(t)^2}+\sigma^2\sigma_x^{(t)^4}+\sigma_x^{(t)^4}\sigma_y^2}{\left(\sigma^2+\sigma_x^{(t)^2}\right)^2}, \tag{A12}$$

which leads to

$$\sigma_x^{(t+1)^2}\left(\sigma^4+2\sigma^2\sigma_x^{(t)^2}+\sigma_x^{(t)^4}\right)=\sigma^4\sigma_x^{(t)^2}+\sigma^2\sigma_x^{(t)^4}+\sigma_x^{(t)^4}\sigma_y^2. \tag{A13}$$

At convergence $\sigma_x^{(t+1)^2}=\sigma_x^{(t)^2}=\hat{\sigma}_x^2$. Therefore Eq. A13 becomes

$$\sigma^4\hat{\sigma}_x^2+2\sigma^2\hat{\sigma}_x^4+\hat{\sigma}_x^6=\sigma^4\hat{\sigma}_x^2+\sigma^2\hat{\sigma}_x^4+\hat{\sigma}_x^4\sigma_y^2. \tag{A14}$$

Simplifying and dividing each side by $\hat{\sigma}_x^2$ results in

$$\hat{\sigma}_x^2+\sigma^2=\sigma_y^2, \tag{A15}$$

which simplifies to the correct result for the variance of the states ($\hat{\sigma}_x^2$) for the system posed in Example 1 (Eq. 40)

$$\hat{\sigma}_x^2=\sigma_y^2-\sigma^2. \tag{A16}$$

Manuscript received July 1, 1996, and revision received Oct. 20, 1997.