Published in final edited form as:

J Process Control. 2017 December ; 60: 115–127. doi:10.1016/j.jprocont.2017.04.004.

Hybrid online sensor error detection and functional redundancy for systems with time-varying parameters

Jianyuan Feng^a, Kamuran Turksoy^b, Sediqeh Samadi^a, Iman Hajizadeh^a, Elizabeth Littlejohn^c, and Ali Cinar^{a,b,*}

^aDepartment of Chemical and Biological Engineering, Illinois Institute of Technology, Chicago, IL, United States

^bDepartment of Biomedical Engineering, Illinois Institute of Technology, Chicago, IL, United States

^cBiological Sciences Division, University of Chicago, Chicago, IL 60637, United States

Abstract

Supervision and control systems rely on signals from sensors to receive information to monitor the operation of a system and adjust manipulated variables to achieve the control objective. However, sensor performance is often limited by their working conditions and sensors may also be subjected to interference by other devices. Many different types of sensor errors such as outliers, missing values, drifts and corruption with noise may occur during process operation. A hybrid online sensor error detection and functional redundancy system is developed to detect errors in online signals, and replace erroneous or missing values detected with model-based estimates. The proposed hybrid system relies on two techniques, an outlier-robust Kalman filter (ORKF) and a locally-weighted partial least squares (LW-PLS) regression model, which leverage the advantages of automatic measurement error elimination with ORKF and data-driven prediction with LW-PLS. The system includes a nominal angle analysis (NAA) method to distinguish between signal faults and large changes in sensor values caused by real dynamic changes in process operation. The performance of the system is illustrated with clinical data continuous glucose monitoring (CGM) sensors from people with type 1 diabetes. More than 50,000 CGM sensor errors were added to original CGM signals from 25 clinical experiments, then the performance of error detection and functional redundancy algorithms were analyzed. The results indicate that the proposed system can successfully detect most of the erroneous signals and substitute them with reasonable estimated values computed by functional redundancy system.

Keywords

Functional sensor redundancy; Sensor error detection; Kalman filter; Locally weighted partial least squares regression; Glucose concentrations; CGM

*Corresponding author at: Department of Chemical and Biological Engineering, Illinois Institute of Technology, Chicago, IL, United States. cinar@iit.edu (A. Cinar).

1. Introduction

Sensor failure is a frequent problem in monitoring and control systems due to the characteristics and work conditions of sensors. Erroneous sensor signals can cause an online control system to suggest wrong manipulated variable values that may deteriorate system performance and cause dangerous situations.

Sensor errors can be divided into two main categories: hard failure (complete failure) and soft failure such as bias, drift and outliers [21]. It is easier to detect hard failures while it is difficult to detect and reconcile some soft failures, since it may be difficult to distinguish between sensor values reporting changes in process operation under normal situations and soft faults that appear intermittently or have small deviations. Several approaches have been proposed for sensor fault detection and diagnosis (FDD) and data reconciliation. Technologies for FDD are grouped into three categories: quantitative model based, qualitative model based, and process history based (data-driven) techniques [28]. For model-based techniques, prior knowledge (either quantitative or qualitative) about the process is needed to develop the model for computing residuals between measured and predicted values and compare the residual with a threshold to determine the presence of an error. The models can be first principles, empirical or experiential. For process history based techniques, a large amount of historical process data is needed to create a database of fault patterns, then compute statistical limits that indicate the significance of deviations in sensor readings [22]. Qualitative model-based techniques have also been integrated with data-driven techniques to leverage the power of multivariate statistical approaches and knowledge-based systems [27]. A different paradigm is to develop robust control systems that can tolerate sensor errors [15,20]. A limitation of a robust control approach is the need for prior knowledge of sensor error to build the controller. Error detection methods based on Kalman filter [19,30], artificial neural networks [23,25] (ANN) and principal component analysis [9,29] (PCA) are some of the model-based and data-driven techniques that have been used in many applications.

For online control systems, sensor error detection is only part of the work, the erroneous sensor reading must be modified before it is used in control decisions. Data reconciliation [2] techniques have been proposed to build models for providing estimates of the erroneous or missing sensor signal. Complex systems with time-varying parameters and/or nonlinearities necessitate either fixed reliable models that can represent the process over wide ranges of operating conditions or adaptive models. The development of detailed models can be challenging and adaptive simple models provide an attractive and feasible option. However, the use of erroneous data in updating model parameters would yield a model that includes the effects of sensor errors, and affect error detection and data reconciliation. In order to address this problem, we propose a hybrid online sensor error detection and functional redundancy (SED&FR) system based on two technologies, the outlier-robust Kalman filter [12,26] (ORKF) and the locally-weighted partial least squares [12,18] (LW-PLS).

The two technologies use different approaches for fault detection and analysis. ORKF builds a Kalman model by using signals collected in recent past sampling times. LW-PLS builds a

partial least squares (PLS) model by using the signal samples in the database of historical data that are most similar to current signal samples. The fault reported by ORKF indicates that the current signal is different from the signal trace indicated by recent past values of signals, but it may not be a fault if compared in LW-PLS because the signal samples in the database have similar behavior. Conversely, if a new signal pattern does not match any type of true signal trace in the database, LW-PLS may not be able to provide the correct detection. Hence, the use of both algorithms simultaneously provides a more robust fault detection and diagnosis effort.

The final estimate based on functional redundancy relies on the same two different methods to build the models and provide estimates for erroneous or missing sensor measurements. The hybrid system increases the probability of rapid detection of sensor faults and improves the accuracy of the estimated values. The results from each algorithm can be compared or combined to find the optimal reconciled estimated value. When a continuous sensor error is detected that lasts for a long period, LW-PLS would be more advantageous in making a prediction.

The remainder of the paper is structured as follows. The two technologies (ORKF and LW-PLS) and the nominal angle analysis algorithm for discrimination between sensor readings affected by large changes in process operation and sensor faults are described in Section 2. The results of the system performance with data from clinical experiments are given in Section 3. The discussion of results and conclusions are provided in Section 4 and Section 5, respectively.

2. Methods

For an online control system, the incoming sensor data must be reconciled before being used to update the model equations or calculate a value for manipulate variables. Hence, data reconciliation techniques that require access to future data samples, such as the Kalman smoother [3,17] or Savitzky-Golay filter (SGF) [24] are not appropriate. The techniques that will be used should be capable of both error detection and data reconciliation in order to maintain the function of the control system. The erroneous or missing measurements should be also replaced by appropriate model-based estimates. The ORKF [26] and the LW-PLS [18] provide two complementary technologies. ORKF provides predictions of sensor readings based on the recent behavior of the sensor. LW-PLS uses a model developed by historical data and updates its parameters with current sensor information. Consequently, its predictions take into account the historical information about the expected system performance.

2.1. Outlier-robust Kalman filter

The ORKF method has been used to solve many sensor outlier-related problems in various fields such as global positioning system (GPS) data analysis [1] and robotic systems [26]. Sensor data $\mathbf{y}_k \in \mathbb{R}^{d_1}$ can be described by Kalman filter system equations with hidden states $\mathbf{x}_k \in \mathbb{R}^{d_2}$ where d_1 and d_2 are the dimensions of outputs (number of sensors) and state variables:

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k \quad (1)$$

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{s}_k \quad (2)$$

where $\mathbf{C} \in \mathbb{R}^{d_1 \times d_2}$ is the observation matrix, $\mathbf{A} \in \mathbb{R}^{d_2 \times d_2}$ is the state transition matrix, $\mathbf{v}_k \in \mathbb{R}^{d_1 \times 1}$ is the observation noise at time step k , and $\mathbf{s}_k \in \mathbb{R}^{d_2 \times 1}$ is the state noise at time step k . We assume that $\mathbf{v}_k$ and $\mathbf{s}_k$ are uncorrelated mean-zero Gaussian noise: $\mathbf{v}_k \sim \text{Normal}(0, \mathbf{R})$, $\mathbf{s}_k \sim \text{Normal}(0, \mathbf{Q})$. Covariance matrices $\mathbf{R} \in \mathbb{R}^{d_1 \times d_1}$ and $\mathbf{Q} \in \mathbb{R}^{d_2 \times d_2}$ are diagonal matrices, with $r \in \mathbb{R}^{d_1 \times 1}$ and $q \in \mathbb{R}^{d_2 \times 1}$ on their diagonal, respectively. For the standard Kalman filter, for a data set of length N and $k = 1, \dots, N$, the filter propagation and update equations are:

$$\mathbf{x}'_k = \mathbf{A}\langle \mathbf{x}_{k-1} \rangle \quad (3)$$

$$\Sigma'_k = \mathbf{A} \Sigma_{k-1} \mathbf{A}^T + \mathbf{Q} \quad (4)$$

$$\mathbf{S}'_k = \left(\mathbf{C} \Sigma'_k \mathbf{C}^T + \mathbf{R} \right)^{-1} \quad (5)$$

$$\mathbf{K}'_k = \Sigma'_k \mathbf{C}^T \mathbf{S}'_k \quad (6)$$

$$\langle \mathbf{x}_k \rangle = \mathbf{x}'_k + \mathbf{K}'_k \left(\mathbf{y}_k - \mathbf{C} \mathbf{x}'_k \right) \quad (7)$$

$$\Sigma_k = \left(\mathbf{I} - \mathbf{K}'_k \mathbf{C} \right) \Sigma'_k \quad (8)$$

where $\langle \mathbf{x}_k \rangle$ is the posterior mean vector of the state $\mathbf{x}_k$, Σ_k is the covariance matrix of $\mathbf{x}_k$, and $\mathbf{S}'_k$ is the covariance matrix of the residual prediction error at step k . As a model used to predict the measurements, the data used to train the model has to be robust to outliers. Hence, the coefficient matrices representing the system dynamics ($\mathbf{C}$, $\mathbf{A}$, $\mathbf{R}$ and $\mathbf{Q}$) are unknown and need to be adaptive to fit dynamic changes in a system with time-varying coefficients. Consequently, the standard Kalman filter with constant coefficients that

considers all data samples to be part of the data cloud would not be appropriate and could generate erroneous estimates.

To overcome these limitations, a Bayesian algorithm that treats the weights associated with each data sample probabilistically is introduced. A scalar gamma-distributed weight w_k is assigned for each observed data sample, y_k . Gamma distribution is chosen for the weights to ensure that they remain positive. The resulting prior distributions are Normal, with means and variances as:

$$\begin{aligned} y_k | x_k, w_k &\sim \text{Normal}(\mathbf{C}x_k, \mathbf{R}/w_k) \\ x_k | x_{k-1} &\sim \text{Normal}(\mathbf{A}x_{k-1}, \mathbf{Q}) \\ w_k &\sim \text{Gamma}(a, b) \end{aligned} \quad (9)$$

where a and b are prior scale parameters for w_k . This entire problem can be treated as an Expectation-Minimization-like (EM) learning problem [8,13] and model parameters can be found by maximizing the log likelihood $\log p(\mathbf{x}_{1:k}, \mathbf{y}_{1:k}, w_{1:k})$:

$$\log p(\mathbf{x}_{1:k}, \mathbf{y}_{1:k}, w_{1:k}) = \sum_{i=1}^k \log p(y_i | x_i, w_i) + \sum_{i=1}^k \log p(x_i | x_{i-1}) + \log p(x_0) + \sum_{i=1}^k \log p(w_i) \quad (10)$$

where $\mathbf{x}_{1:k}$, $\mathbf{y}_{1:k}$ and $\mathbf{w}_{1:k}$ denote $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$, $\{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_k\}$, $\{w_1, w_2, \dots, w_k\}$, respectively. Similar to [26], matrices $\mathbf{C}$, $\mathbf{A}$, $\mathbf{R}$ and $\mathbf{Q}$ at time step k can be solved by incorporating the EM update equations to give the outlier-robust Kalman filter (ORKF):

$$\mathbf{x}'_k = \mathbf{A}_{k-1} \langle \mathbf{x}_{k-1} \rangle \quad (11)$$

$$w_k = \frac{a+0.5}{b + \left[(\mathbf{y}_k - \mathbf{C}_{k-1} \mathbf{x}'_k)^T \mathbf{R}_{k-1}^{-1} (\mathbf{y}_k - \mathbf{C}_{k-1} \mathbf{x}'_k) \right]} \quad (12)$$

$$\mathbf{C}_k = \left[\sum_{i=1}^k \lambda^{-i} w_i \mathbf{y}_i \mathbf{x}'_i \right] \left[\sum_{i=1}^k \lambda^{i-k} w_i (\mathbf{x}'_i \mathbf{x}'_i{}^T) \right]^{-1} \quad (13)$$

$$\mathbf{A}_k = \left[\sum_{i=1}^k \lambda^{k-i} \mathbf{x}'_i \mathbf{x}'_i{}^T \right] \left[\sum_{i=1}^k \lambda^{i-k} (\mathbf{x}'_{i-1} \mathbf{x}'_{i-1}{}^T) \right]^{-1} \quad (14)$$

$$r_{km} = \frac{1}{k} \sum_{i=1}^k \lambda^{k-i} w_i \left[y_{im} - C_k(m, :) x'_i \right]^2 \quad (15)$$

$$q_{kn} = \frac{1}{k} \sum_{i=1}^k \lambda^{k-i} \left[x'_{in} - A_k(n, :) x_k \right]^2 \quad (16)$$

$$\sum'_k = Q_k \quad (17)$$

$$S'_k = \left(C_k \sum'_k C_k^T + \frac{R_k}{w_k} \right)^{-1} \quad (18)$$

$$K'_k = \sum'_k C_k^T S'_k \quad (19)$$

$$\langle x_k \rangle = x'_k + K'_k (y_k - C_k x'_k) \quad (20)$$

$$\sum_k = (I - K'_k C_k) \sum'_k \quad (21)$$

where $m = 1, \dots, d_1$; $n = 1, \dots, d_2$; r_{km} is the m th coefficient of the vector r_k ; q_{kn} is the n th coefficient of the vector q_k ; $C_k(m, :)$ is the m th row of the matrix C_k ; $A_k(n, :)$ is the n th row of A_k , and λ is the forgetting factor. In the ORKF, weight w_k is assigned in Eq. (12) to determine the parameters in the model, $a = 1$ and $b = 1$ is set according to reference [26]. Weight w_k can be assigned values from 0 to 1.5 depending on the magnitude of the estimated measurement error ($y_k - C_{k-1} x'_k$) compared to the expected measurement error R_k so that if the measurement error is larger than expected, a small w_k is assigned to this measurement and the model will not be updated to follow the wrong tendency.

Forgetting factor λ is also introduced to the ORKF. This way, only the data close to current time step will be used to generate the model, which enables the model to adapt to variations in a system with time-varying parameters. Smaller λ gives more weight on recent data. Hence, λ needs to be selected to make the trajectory of ORKF ($\langle x_k \rangle$) follow most of the dynamic changes in CGM while robust to abnormal CGM outliers.

The initial values for matrices A and C are usually set to identity matrix since the algorithm is relatively insensitive to the initialization of A and C [26]. Initial values of R and Q are selected based on the initial estimate of the noise in observed data. Each time a measurement is updated, the ORKF will calculate an estimated value ($\hat{\mathcal{Y}}_{O,k} = C\langle x_k \rangle$) that will be used in sensor error detection and data reconciliation.

2.2. Locally weighted partial least squares

Locally weighted regression (LWR) [6], which is also called just in time learning, lazy learning or model-on-demand, constructs a local model by prioritizing samples in a database according to the similarity between them and a query sample. In the database there are many sample sets paired with their prediction values. The LW-PLS compares the current sample with samples in the database and gives a weight for each sample in the database according to their similarity with the query sample and use the weighed samples to generate a PLS model to compute a prediction for the query sample.

Assume that there are E number of completed SP_e ($e \in [1, 2, \dots, E]$) sample steps in each experiment. A database contains measurements in these historical experiments as input $X \in \mathbb{R}^{N \times M}$ and one-step-ahead predictions as output $Y \in \mathbb{R}^{N \times 1}$. The number of pairs of x_n and y_n in matrix X and Y is indicated by N . The number of variables in x_n is indicated by M . Raw data from the database are smoothed with a SGF to reduce noise, and all variables are mean-centered and scaled by their standard deviation. The n th row ($n = 1, 2, \dots, N$) of matrix X and Y are x_n^T and y_n^T , respectively:

$$x_n = [S_{e,k-\alpha}, S_{e,k-\alpha+1}, \dots, S_{e,k-1}, \Delta S_{e,k-\alpha}, \Delta S_{e,k-\alpha+1}, \dots, \Delta S_{e,k-1}, u_{e,k-\alpha}, u_{e,k-\alpha+1}, \dots, u_{e,k-1}]^T \quad (22)$$

$$y_n = S_{e,k} \quad (23)$$

Where $S_{e,k}$ is the sensor signal in e th experiment ($e \in [1, 2, \dots, E]$) at step k ($k \in [\alpha + 1, \alpha + 2, \dots, SP_e]$), $S_{e,k} = S_{e,k} - S_{e,k-1}$, and $u_{e,k}$ is the signal from other related e th experiment at step k (if there exist more than one $S_{e,k}$, their values can also be added at the end of sample x_n), and α denotes the sliding past data window (number of sampling times in the past) of data used to generate sample x_n .

Comparing a query sample in LW-PLS with all the samples in the database is time consuming and not necessary. For example, a query sample that shows rapid increase in the signal does not need to be compared with the samples in the database showing rapid decrease or flat profiles, and those samples not related with query sample would be assigned 0 weights. Hence, a cluster capable of distinguishing the basic characteristic (shape) of the samples and comparing the query sample only with the samples that have the same basic characteristic can improve the efficiency of diagnosis.

A clustering algorithm (xn into five different groups according to the characteristics of sensor signals. The five different groups are increase period (IP), decrease period (DP), increase followed by decrease period (IDP), decrease followed by increase period (DIP) and steady period (SP). Fig. 1 illustrates the characteristics of sensor signals in these five groups. Assume one period of sensor signal data ($S_{e,k-a}, S_{e,k-a+1}, \dots, S_{e,k-1}$) as points in a two-dimensional plot with coordinate $\mathbf{XC} = [1, 2, \dots, a]$, $\mathbf{YC} = [S_{e,k-a}, S_{e,k-a+1}, \dots, S_{e,k-1}]$. $y_{\max}$ and $y_{\min}$ are the maximum and minimum values of $\mathbf{YC}$, the tangents of start point $(1, S_{e,k-a})$ and the end point $(a, S_{e,k-1})$ intersect at (X_{int}, Y_{int}) .

The one-step-ahead prediction $\hat{y}_{LP}$ for one query sample $\mathbf{x}_q$ (have the same format as $\mathbf{x}_n$ using current sensor reading) is calculated by using the following procedure[18]:

1. Determine the number of latent variables R_{LP} and set $r = 1$.
2. Choose the same group of data $(\mathbf{x}_n, \mathbf{y}_n)$ in the database clustering algorithm as $\mathbf{x}_q$ using the clustering algorithm in Fig. 2, assume there are N_c samples of $\mathbf{x}_n$ and $\mathbf{y}_n$ in the database that belong to the same group of $\mathbf{x}_q$.
3. Calculate the similarity matrix $\mathbf{\Omega}$

$$D_n = \sqrt{(\mathbf{x}_n - \mathbf{x}_q)^T (\mathbf{x}_n - \mathbf{x}_q)} \quad (24)$$

$$\omega_n = \exp \left(-\frac{D_n}{\sigma_d \phi} \right) \quad (25)$$

$$\mathbf{\Omega} = \text{diag}(\omega_1, \omega_2, \dots, \omega_{N_c}) \quad (26)$$

where σ_d is the standard deviation of D_n ($n = 1, 2, \dots, N$) and ϕ is a localization parameter; the similarity decreases steeply when ϕ is small and gradually when ϕ is large. The LW-PLS reduces to linear PLS when the similarity matrix $\mathbf{\Omega}$ is an identity matrix.

4. Calculate $\mathbf{X}_r$, $\mathbf{Y}_r$ and $\mathbf{x}_{q,r}$

$$\bar{x}_m = \sum_{n=1}^{N_c} \omega_n x_{nm} / \sum_{n=1}^{N_c} \omega_n \quad (27)$$

$$\bar{y} = \sum_{n=1}^{N_c} \omega_n y_n / \sum_{n=1}^{N_c} \omega_n \quad (28)$$

$$\mathbf{X}_r = \mathbf{X} - \mathbf{1}_{M_c} [\bar{x}_1, \bar{x}_2, \dots, \bar{x}_M] \quad (29)$$

$$\mathbf{Y}_r = \mathbf{Y} - \bar{y} \quad (30)$$

$$\mathbf{x}_{q,r} = \mathbf{x}_q - [\bar{x}_1, \bar{x}_2, \dots, \bar{x}_M]^T \quad (31)$$

where $\mathbf{1}_N \in \mathbb{R}^N$ denotes the unity vector. Eqs. (27) and (28) are calculating the weighted “mean” value for building $\mathbf{X}_r$, $\mathbf{Y}_r$, and $\mathbf{x}_{q,r}$.

5. Derive the r th latent variable of $\mathbf{X}$

$$\mathbf{t}_r = \mathbf{X}_r \mathbf{V}_r \quad (32)$$

:

where $\mathbf{V}_r$ is the maximum eigenvector of $\mathbf{X}_r^T \boldsymbol{\Omega} \mathbf{Y}_r \mathbf{Y}_r^T \boldsymbol{\Omega} \mathbf{X}_r$ which corresponds to the maximum eigen value.

6. Derive the r th loading vector $\mathbf{p}_r$ and regression coefficient vector $\mathbf{q}_r$

$$\mathbf{p}_r = \frac{\mathbf{X}_r^T \boldsymbol{\Omega} \mathbf{t}_r}{\mathbf{t}_r^T \boldsymbol{\Omega} \mathbf{t}_r} \quad (33)$$

$$\mathbf{q}_r = \frac{\mathbf{Y}_r^T \boldsymbol{\Omega} \mathbf{t}_r}{\mathbf{t}_r^T \boldsymbol{\Omega} \mathbf{t}_r} \quad (34)$$

7. Drive the r th latent variable of $\mathbf{x}_q$

$$t_{q,r} = \mathbf{x}_{q,r}^T \mathbf{V}_r \quad (35)$$

8. If $r = R_{LP}$, calculate the estimated output

$$\hat{y}_{LP} = \bar{y} + \sum_{r=1}^R t_{q,r} \mathbf{q}_r \quad (36)$$

and end the estimation. Otherwise, set

$$\mathbf{X}_{r+1} = \mathbf{X}_r - \mathbf{t}_r \mathbf{p}_r^T \quad (37)$$

$$\mathbf{Y}_{r+1} = \mathbf{Y}_r - \mathbf{t}_r \mathbf{p}_r^T \quad (38)$$

$$\mathbf{x}_{q,r+1} = \mathbf{x}_{q,r} - \mathbf{t}_{q,r} \mathbf{p}_r \quad (39)$$

9. Set $r = r + 1$ and go to step 5

There are two tuning parameters; the number of latent variables R_{LP} and localization parameter ϕ . The two parameters can be determined by applying cross-validation to the parameter tuning dataset. The best values of R_{LP} and ϕ are indicated by the minimum model prediction residual (MPR). Note that the parameter tuning dataset should be error free.

2.3. Nominal angle analysis

For the model-based SED&FR system, false positive detections due to the model inaccuracy and the erroneous signal estimates caused by false detections, are important problems. This occurs more frequently when the dynamic changes in the system are too large or too rapid, and the model used in SED&FR system does not have enough prior information to adapt to such large dynamic changes. Since the criterion for error detection is based on the residual between the measurement and model prediction, large number of false positives can occur. A simple threshold for signal change S_k (the variable S_k , S_k and u_k without subscript e denote the online measurement) is not reliable to distinguish between sensor errors and correct sensor readings caused by rapid changes in sensor readings, since large real dynamic changes in sensor readings can also cause large values of S_k . For signal smoothness, a continuous process signal with no bias will be smoother than one with biased signals. Traditionally, when there is a biased signal a specific threshold is designed for sensor signal change difference (Second derivative, $S_k - S_{k-1}$) instead of S_k (First derivative), and the sensor reading is considered to be correct if the difference is less than the threshold. But a constant threshold may not be suitable for all kinds of situations. In order to distinguish between sensor bias and real dynamic changes in sensor readings, the nominal angle analysis (NAA) concept is proposed to add to the SED&FR system.

The false positives are often caused by the model estimation outcomes that cannot follow the rapid dynamic changes. The model is more accurate when the signal is stable. And sensor readings will not change much during a period where the system is not subjected to large disturbances, and a small threshold is needed to detect small magnitude sensor bias or early stages of a drift. Hence, for the initial check, the threshold for sensor signal change difference ($S_k - S_{k-1}$) should be more sensitive when the signal is stable (when the first derivative is small, the second derivative threshold should be large). And the threshold should be less sensitive when the signs of first and second derivative of signal are the same since that may be caused by a large disturbance causing a large dynamic change. In order to

fulfill these characteristics, NAA is used to develop this dynamic threshold with only two turning parameters, the nominal sensor signal change ΔS^N and the maximum nominal angle Φ^{max} . Compared with fixed threshold for second derivatives of the signal, the dynamic threshold of NAA is more accurate and compared with other methods with a large number of parameters for tuning, such as neural network, the NAA is more convenient for parameter identification.

Consider a triangle ($\triangle abc$) illustrated with three examples in Fig. 3. The length ($I_{o,a}$) which is the height of the triangle indicates the nominal sensor signal change (ΔS^N), usually, $|S|_{ave} + |S|_{std}$ is used as ΔS^N ($|S|_{ave}$ is the average value of $|S|_{e,k}$ and $|S|_{std}$ is the standard deviation of $|S|_{e,k}$). The lengths of segments $I_{o,b}$ and $I_{o,c}$ are equal to the absolute value of sensor signal change at step k (ΔS_k) and step $k-1$ (ΔS_{k-1}), respectively, and displayed as the basis of the triangle. The location of $I_{o,a}$ on the right side or left side depends on the sign of the sensor signal change (positive or negative). The nominal angle Φ is generated having a as the vertex and by connecting the point a with b and c .

The signal value at time k is assumed to be without a fault if $\Phi < \Phi^{max}$. By determining the values of the two tuning parameters ΔS^N and Φ^{max} , a threshold for $S_k - S_{k-1}$ that is varying dynamically is provided for sensor readings. The up and down threshold can be determined by Φ^{max} and ΔS^N as:

$$T_{up} = \Delta S^N \left(\tan \left(\tan^{-1} \left(\frac{\Delta S_{k-1}}{\Delta S^N} \right) + \Phi^{max} \right) - \frac{\Delta S_{k-1}}{\Delta S^N} \right) \quad (40)$$

$$T_{down} = -\Delta S^N \left(\frac{\Delta S_{k-1}}{\Delta S^N} - \tan \left(\tan^{-1} \left(\frac{\Delta S_{k-1}}{\Delta S^N} \right) - \Phi^{max} \right) \right) \quad (41)$$

A large threshold is given for variations in sensor signals when sensor readings are changing rapidly ($|S_{k-1}|$ is large) and small threshold is given for variations in sensor signals when sensor readings are stable ($|S_{k-1}|$ is small) (Fig. 4). And a larger upper bound is given when

S_{k-1} is positive, and a larger lower bound is given when S_{k-1} is negative (Fig. 4). The proposed method allows more variation in sensor dynamics when the $S_k - S_{k-1}$ is in the same direction as S_{k-1} . Larger Φ^{max} and ΔS^N will both relax the threshold, and larger ΔS^N will make threshold more sensitive to S_{k-1} . According to the dynamics of sensor data, ΔS^N and Φ^{max} should be tuned to minimize the ratio between false positive and true positive.

Besides providing the initial check for sensor signals to reduce the false positives (detecting errors when the sensor is working correctly) and the false negatives (missing sensor errors when they exist), the NAA can also be used for data reconciliation. When sensor bias is detected with a hybrid method such as proposed in this work, each method provides a reconciled value. These reconciled values may be different for each method. The reconciled sensor reading can be based on ORKF, LW-PLS, or a combination of the two estimates (in this paper, the average of ORKF and LW-PLS estimations is used). If sensor bias is detected, the angles of retuning candidates are compared (Fig. 5). With estimated sensor values ($\hat{y}_{O,k}$

and $\hat{y}_{LP,k}$, ΔS_k^{ORKF} , ΔS_k^{LP} and ΔS_k^{AVE} are defined in Eqs. (42)–(44), the angles Φ_{ORKF} , Φ_{LP} and Φ_{AVE} are generated similarly to the method described earlier. In this way, Φ_{ORKF} , Φ_{LP} and Φ_{AVE} are compared with $\Phi_{Original}$ generated by S_k . If the $\Phi_{Original}$ is the smallest angle, then no error is declared. Otherwise, the estimated value with smallest angle is selected to replace the erroneous sensor signal.

$$\Delta S_k^{ORKF} = \hat{y}_{O,k} - S_{k-1} \quad (42)$$

$$\Delta S_k^{LP} = \hat{y}_{LP,k} - S_{k-1} \quad (43)$$

$$\Delta S_k^{AVE} = (\hat{y}_{O,k} + \hat{y}_{LP,k})/2 - S_{k-1} \quad (44)$$

2.4. Sensor error detection and retuning

Three types of sensor errors are considered in this paper: missing signal, signal stuck to a constant value (measurement signal keeps displaying the same value) and biased signal (spike, step change, drifts, and specified sensor sensitivity deterioration). Missing signals and stuck signals are easy to detect: if there is no signal at step k , then there is a missing signal and if the sensor is repeating the same value, then it is signal stuck. Note that signal stuck is not always an error since a repeated value is possible when the sensor signal is stable. Only when NAA indicates that model estimates have a better value, will signal stuck error be reported. For biased signals, two model prediction values from ORKF and LW-PLS are compared with the real-time sensor signal in order to determine the signal bias.

Fig. 6 shows the process for sensor error detection. If the sensor signal is missing or stuck at a constant value, data reconciliation by functional redundancy is initiated. Otherwise, the signal reading is sent to the NAA algorithm to determine if the change in signal value should be considered as erroneous. Then, ORKF and LW-PLS provide estimates of the current signal value ($\hat{y}_{O,k}$ and $\hat{y}_{LP,k}$). If the differences between these estimates and S_k for two methods are both beyond their threshold (T) and any of the estimates of signal value have smaller angles than $\Phi_{Original}$ in NAA, sensor bias will be reported and the signal value is replaced with the estimate with the smallest angle in NAA. The two methods are looking at different aspect of the signal, ORKF relies more on the smoothness of the current signal (if there is any abnormal rapid change), and LW-PLS is based on historical data (if the current signal tendency is same as the historical reference in the database). If there is a rapid change in sensor signal due to large dynamic change in the process, ORKF may report a false positive because of the large change, and LW-PLS may report false positive if the pattern of change is not stored in the database report. Requiring that both methods report the sensor signal to be beyond threshold will make the error report more reliable and reduce the number of false alarms. Since the performance of LW-PLS is dependent on the accuracy of

the prediction, the horizon parameter α should be determined according to the duration of the effects of previous data on current data. For ORKF, the goal is to make an accurate estimate when the measurement is considered as erroneous and close to measurement when the measurement is considered as correct. ORKF sacrifices some of the accuracy to make the estimation more robust to outliers. The parameters, λ and d_2 in ORKF, are selected by minimizing both type 1 error (false positive) and type 2 error (false negative) by using the objective function:

$$J = \min_{(\lambda, d_2, T, \Delta S^N, \Phi^{\max})} \left(\frac{FP}{RP} + \frac{FN}{RP} \right) \quad (45)$$

where FP, FN, and RP denote false positive, false negative and right positive (correct), respectively. An example of parameters selection for the SED&FR techniques proposed is described in Section 3.

3. Detection of continuous glucose monitoring sensors in an artificial pancreas system

Continuous glucose monitors (CGM) can help people with diabetes monitor their blood glucose concentration (BGC) at a higher frequency and with less manual effort. The CGM sensor is also a critical component of an artificial pancreas (AP) system which uses real-time glucose concentration (GC) information as an input to calculate appropriate real-time insulin infusion rates for people with type 1 diabetes (T1D).

3.1. Data processing

Commercially available glucose sensors and CGM systems have a sampling time of 5 min, and the computation time for SED&FR system is 0.2 s (CPU: Intel i5-3470, RAM: 8GB, Matlab 2015a). Insulin infusion rate data are also used for building the models for LW-PLS analysis.

For LW-PLS, a database contains clinical data from five different subjects with T1D with CGM, CGM change and insulin infusion rate as input X and with one step prediction CGM as output Y . Eqs. (22) and (23) can be rewritten for this data set as Eqs. (46) and (47):

$$\mathbf{x}_n = [G_{e,k-\alpha}, G_{e,k-\alpha+1}, \dots, G_{e,k-1}, \Delta G_{e,k-\alpha}, \Delta G_{e,k-\alpha+1}, \dots, \Delta G_{e,k-1}, ins_{e,k-\alpha}, ins_{e,k-\alpha+1}, \dots, ins_{e,k-1}]^T \quad (46)$$

$$\mathbf{y}_n = G_{e,k} \quad (47)$$

where $G_{e,k}$ is the CGM value in e th historical experiment at time step k , $G_{e,k} = G_{e,k} - G_{e,k-1}$, and $ins_{e,k}$ is the insulin infusion rate in e th historical experiment at step k .

The long-term insulin effect on BGC horizon [16]: the insulin injected one hour earlier may still have effect on blood GC and CGM. Considering the sampling time of 5 min, α , is selected to be 12. The tuning parameter R_{LP} and ϕ are determined by cross-validation. The search range for R_{LP} and ϕ are [1, 2, 3, ..., 20] and [0.1, 0.2, 0.3, ..., 1], respectively. Clinical data from five patients provide 3360 data samples for building the database of LW-PLS. And these data were also used as parameter tuning dataset for LW-PLS. We chose $R_{LP} = 5$ and $\phi = 0.9$ which make the minimum MPR. The cluster threshold, S_{TI} , S_{TD} are valued by 0.05 and -0.05 to make each group have about the same amount of samples in the database.

We set current-step CGM estimation as the output of the model ($d_1 = 1$). As mentioned in Section 2, the parameters selection of λ , d_2 , T , S^N and Φ^{max} are using the objective function in Eq. (45). The optimal value of the forgetting factor λ , d_2 , T , S^N , and Φ^{max} are sought by testing different values in the range [0.5, 0.1, 0.15, ..., 0.95], [1, 2, 3, ..., 8], [1, 2, 3, ..., 15], [5, 6, 7, ..., 15] and [5, 6, 7, ..., 25], respectively. The best λ , d_2 , T , S^N , and Φ^{max} are found to be 0.15, 3, 14, 10, 20. It is assumed that regular sensor noise is not as large as dynamic changes in BG, consequently, R and Q are set as $R = 10I$ and $Q = 10^2I$, where I is the identity matrix.

Clinical CGM and insulin infusion rate data from 25 subjects with T1D are used for testing the proposed SED&FR system. Each subject has 56 h of data that include CGM and insulin infusion rates. The statistical information of the clinical data is displayed in Table 1.

3.2. CGM value jump caused by calibration

The minimally invasive continuous glucose sensors output a signal that is proportional to the glucose concentration value in the subcutaneous interstitial fluid. A calibration algorithm is used to convert the raw sensor signal, typically in nanoamperes to a glucose estimation (mg/dl in the United States) [5].

In most cases, the conversion can be done by using a simple linear equation:

$$y = mx + b \quad (48)$$

where y is the glucose value and x is the sensor current. m and b are model parameters that need to be determined by using one or multiple data sets of y and x (one-point calibration or multi-point calibration). When sensor calibration occurs (it must be done at least twice per day with the current CGM sensors), the CGM signal may jump from one value to another if the model parameters m and b have changed significantly. Fig. 7C and D illustrate two examples of CGM value jump caused by CGM calibration. The CGM signal jump caused by calibration (SJCC) may cause two problems if the CGM signal is directly used by the AP system. First, for an AP system using a recursively updated data-driven model, SJCC is not a real change caused by a body dynamic change, thus the abnormal signal jump will mislead the model identification and decrease model accuracy. Second, jump of CGM calibration

will make the model used in the AP system give a wrong CGM prediction, and consequently the insulin suggestion based on erroneous glucose prediction may cause potential hypo or hyperglycemia.

The calibration time is treated as announced information. In order to smooth SJCC before the subsequent GC readings are used to updating the GC dynamic model and bolus calculation, the SJCC was detected and reconciled by the algorithm:

At each time step k :

$$\Delta G(k) = G(k) - G(k-1)$$

If calibration was done

$$\begin{aligned} \text{flag}_{ca} &= 1 \\ \text{if } \Phi(k) > \Phi^{max} \&\& \lambda_{ca} = 0 \&\& \text{flag}_{ca} = 1 \\ \lambda_{ca} &= \max \left(\left| \frac{\Delta G(k)}{\min\{0.1G(k), 10\}} \right|, 10 \right) \\ \sigma_{ca} &= \Delta G(k) / \lambda_{ca} \\ \hat{G}(k) &= \lambda_{ca} + G(k-1) \\ \sigma_{ca} &= \sigma_{ca} - 1 \end{aligned}$$

Else if $\lambda_{ca} = 0$

$$\begin{aligned} \hat{G}(k) &= \lambda_{ca} + \hat{G}(k-1) + \Delta G(k) \\ \sigma_{ca} &= \sigma_{ca} - 1 \end{aligned}$$

End if

If $\lambda_{ca} = 0$ or $|\hat{G}(k) - G(k)| < 5$ or $((\hat{G}(k) - G(k)) \hat{G}(k-1) - G(k-1)) < 0$ & $\lambda_{ca} = 0$

$$\begin{aligned} \text{flag}_{ca} &= 0 \\ \lambda_{ca} &= 0 \end{aligned}$$

End if

where $G(k)$ and $\hat{G}(k)$ stand for the original CGM value and the smoothed value, respectively. SJCC is announced if NAA shows that $\Phi(k) > \Phi^{max}$ when there is a calibration. The objective for this method is to separate the large CGM deviation at the calibration point into small increments according to the current CGM level ($G(k)$) and magnitude of CGM jump ($G(k) - G(k-1)$). (Fig. 7C and D) With SJCC detection and smoothing procedure the flow chart in Fig. 6 is modified as displayed in Fig. 8. For each data set, on average there are 9 calibrations. In total, 255 calibrations have been done and 21 of them have been detected as SJCC and smoothed with the proposed algorithm.

3.3. Sensor error generation

Since the raw CGM data may have unknown noise, two different data sets are used to evaluate the performance of the methods proposed. In the first data set, raw data are filtered by SGF to generate the noise free data. In the second data set, the raw data are used without any pretreatment.

A sensor error generator introduces additional errors to the two CGM data sets with a specified error appearance percentage (EAP). EAP means the possibility of the sensor error generator adding an error to the data for each sample. Sensor errors include missing signal, signal stuck, spike, drifts, step change, and pressure-induced sensor attenuations (PISA) [10]. The duration, direction and magnitude of errors are noted as Du_e , Di_e and M_e respectively. The following relations are used for error generation:

Missing signal:

$$\begin{aligned} [G_e(k), G_e(k+1) \dots G_e(k+Du_e-1)] &= [NaN, NaN, \dots, NaN] \\ s.t. Du_e &\in [1, 2, 3, 4] \end{aligned} \quad (49)$$

Signal stuck:

$$\begin{aligned} [G_e(k), G_e(k+1) \dots G_e(k+Du_e-1)] &= [G(k), G(k), \dots, G(k+Du_e-1)] \\ s.t. Du_e &\in [1, 2, 3, 4] \end{aligned} \quad (50)$$

Spike:

$$G_e(k) = G(k) + Di_e M_e G(k) \quad (51)$$

Drift change:

$$\begin{aligned} &[G_e(k), G_e(k+1) \dots G_e(k+Du_e-1)] \\ &= [G(k), G(k+1), \dots, G(k+Du_e-1)] + Di_e M_e [1, 2, \dots, Du_e] / Du_e \\ &s. t. Du_e \in [2, 3, 4, 5] \end{aligned} \quad (52)$$

Step change:

$$\begin{aligned} &[G_e(k), G_e(k+1) \dots G_e(k+Du_e-1)] \\ &= [G(k), G(k+1), \dots, G(k+Du_e-1)] + Di_e M_e G(k) \\ &s. t. Du_e \in [2, 3, 4, 5] \end{aligned} \quad (53)$$

PISA [10]:

$$G_e(k+t) = \begin{cases} G_e(k+t) & \text{if } t \leq \frac{D}{5} \\ (k+t)+P \left(1 - \exp\left(\frac{-5t+D}{\tau}\right)\right) - P \left(1 - \exp\left(\frac{-5t}{\tau}\right)\right) & \text{if } \frac{D}{5} < t < Du_e \end{cases}$$

s. t. $Du_e = D + 3\tau$, $t \in [1, Du_e]$, $P \in [10.948.4]$, $D \in [3.921.7]$ (54)

Where $G_e(k)$ is the faulty CGM measurement generated by the sensor error generator using the original CGM value $G(k)$. Failures of CGM sensor are mainly due to sensor-receiver connection problem (missing signal and signal stuck) and biomechanical issues of the sensor-tissue interface [14], such as motion of the subjects (spike and step change), scar tissue growth or degradation of sensor materials (drift change), and PISA [4,7,10,11]. All types of sensor errors are added to data randomly except PISA that is only added when the subject is at a resting period since PISA usually is caused by a mechanical pressure made on the sensor by the subject (e.g., sleeping position causing pressure on body region where sensor is located) inducing a temporary loss of sensitivity with consequent distortion of the CGM trace [5]. All Du_e , D , P , and D are randomly picked within their range [10]. Small amplitude (± 2 mg/dl) of white noise is also added to CGM data in order to test if the system can distribute real sensor errors with normal signal fluctuation (very small signal fluctuation is not considered as signal error). Different magnitudes of M_e range from [10% 20% 30% 40%] and error frequency from [0.1% 0.5% 1% 5%] of the total samples are tested to illustrate the performance of the proposed system.

3.4. Sensor fault detection and data reconciliation results

Examples of online CGM signal reconciliation by functional redundancy for all kinds of faults introduced are illustrated in Fig. 9. The sensor fault detection results for SGF filtered data in different test conditions are displayed in Tables 2 and 3, and the results for unfiltered data are summarized in Tables 4 and 5.

CGM signals with different types of errors are reconciled by using functional redundancy (Fig. 9). Although the difference between model estimations and the CGM signal is large, no sensor error will be reported in many cases with NAA. With a hybrid sensor error detection technology only when both methods report the error at the same time (large difference between the real CGM value and estimations), the system will consider there exists sensor error. In this way, the number of false positive is reduced. When signal values are estimated with two different techniques (ORKF and LW-PLS) and the results are averaged, better estimates are obtained compared to estimates from only one model.

The formula for computing percentages, sensitivity (S), successful reconciliation rate (SRR), false detection ratio (FDR) are

$$S = \frac{EDRS + EDRF}{EDRS + EDRF + FN} \quad (55)$$

$$SSR = \frac{EDRS}{EDRS + EDRF} \quad (56)$$

$$FDR = \frac{FP}{EDRS + EDRF + FP} \quad (57)$$

where EDRS denotes error detected and reconciled successfully and EDRF denotes error detected but reconciliation failed. EDRS include cases during one continuous sensor error episode where the error is detected and the reconciled (estimated) value is closer to the noise free data for sensor bias or signal stuck, and the absolute difference between the estimated value and the noise free CGM is smaller than 10 mg/dl for the signal missing error. If the estimated value does not satisfy these criteria, the error is classified as EDRF. If no error is reported while there is a sensor error, it is classified as missed error (false negative), and if a sensor error is reported when no error was added to data, it is classified as error reported erroneously (false positive).

4. Discussion

As expected, it is easier to detect sensor signal bias with larger magnitude (Tables 2–5). But when EAP increases, the sensitivity may decrease and FDR may increase because two or more errors may be generated close to each other. Interaction between errors will make the errors more difficult to detect. Additionally, the reconciled data may become less accurate since the models in SED&FR system do not have enough noise free prior data to train the model. One extreme case of four errors close to each other is displayed in Fig. 10, where 2 step changes and 2 drift changes were introduced continuously, the first step change was detected and reconciled properly but the following errors were either undetected or reconciled with poor estimation.

Missing signal values can be detected easily since they have a very clear error behavior (Tables 3 and 5). Spikes and step changes with large magnitude also have better sensitivity. The sensor drift is most difficult to detect due to its similarity to normal CGM behavior like in a meal or exercise condition (continuous slow increase or decrease in CGM signal). Most errors can be reconciled with values closer to the real value. For signal stuck the sensitivity is not equal to 1, it is because if the error is detected when the CGM is stable, the repeated value may not necessarily be considered as an error when estimations from the two models also think the CGM is stable.

In tests with unfiltered data, raw data that may already have unknown sensor errors that have been considered as normal conditions. Although most of the generated sensor errors are detected and reconciled successfully, comparison of Tables 2 and 3 with Tables 4 and 5 shows that the average FDR increased from 0.20 to 0.36, and the average sensitivity and the average SSR decreased by 15% and 8%, respectively. This is expected since the data with unknown sensor errors is considered as the standard to judge the reconciliation and the

unknown data may interfere with the error introduced by the noise generator (i.e., the unknown sensor error may increase the measurement but the generated error may decrease it and reduce the magnitude of the deviation). As EAP is increasing the difference between results with and without SGF filtered data becomes smaller because the number of unknown errors are much less than the number of errors introduced by the sensor error generator.

A hybrid error detection and analytical redundancy system can detect errors more accurately since the errors are reported by both methods that interpret data in different ways, thus provides fewer missed errors and false alarms. The detections based on only ORKF or LW-PLS, the average FDR for SGF filtered data are 1.42 and 2.53, and for unfiltered data are 2.12 and 3.32, respectively. It is much higher compared with the average FDR using both methods (0.198 for SGF filtered data and 0.524 for unfiltered data).

This paper focus on errors in CGM sensors. Many other errors related to AP systems like insulin pump leakage, and controller performance deterioration also have effect on CGM readings because of changes in the GC. In these cases, after errors are reported, additional fault diagnosis should be made to determine the source course of the errors and fix the system accordingly.

5. Conclusion

The proposed hybrid SED&FR system combined two technologies, ORKF and LW-PLS, that together successfully detected most of the sensor errors and reconciled the sensor readings for almost all of the detected errors with values closer to the real value. At the same time, the number of false alarms are relatively low. Such models improve the operation of the AP system by providing reliable estimates of erroneous sensor readings.

Acknowledgments

This work is supported by the National Institutes of Health (NIH) grant 1DP3DK101077-01 and the Juvenile Diabetes Research Foundation International (JDRF) grant 17-2013-472.

Abbreviations

ANN	artificial neural networks
BGC	blood glucose concentration
CGM	continuous glucose monitoring
DIP	decrease followed by increase period
DP	decrease period
EAP	error appearance percentage
EDRF	error detected but reconciliation failed
EDRS	error detected and reconciled successfully
FDD	fault detection and diagnosis

FDR	false detection ratio
GC	glucose concentration
IDP	increase followed by decrease period
IP	increase period
LW-PLS	locally-weighted partial least squares
LWR	locally weighted regression
MPR	model prediction residual
NAA	nominal angle analysis
ORKF	outlier-robust Kalman filter
PCA	principal component analysis
PLS	partial least square
PISA	pressure-induced sensor attenuations
S	sensitivity
SED&FR	sensor error detection and functional redundancy
SGF	Savitzky-Golay filter
SJCC	signal jump caused by calibration
SRR	successful reconciliation rate
SP	steady period

References

1. Agamennoni, Gabriel, Nieto, Juan I., Nebot, Eudardo Mario. An outlier-Robust Kalman filter. 2011 IEEE International Conference on Robotics and Automation (ICRA) IEEE; 2011; p. 1551-1558.
2. Bagajewicz, Miguel J., Jiang, Qiyu. Comparison of steady state and integral dynamic data reconciliation. *Comput Chem Eng.* 2000; 24:2367–2383.
3. Yaakov Bar-Shalom, X., Li, Rong, Kirubarajan, Thiagalingam. *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software.* John Wiley & Sons; 2004.
4. Baysal, Nihat, Cameron, Fraser, Buckingham, Bruce A., Wilson, Darell M., Wayne Bequette, B. Detecting sensor and insulin infusion set anomalies in an artificial pancreas. 2013 American Control Conference (ACC) IEEE; 2013. p. 2929-2933.
5. Wayne Bequette B. Continuous glucose monitoring: real-time algorithms for calibration, filtering, and alarms. *J Diabetes Sci Technol.* 2010; 4:404–418. [PubMed: 20307402]
6. Cleveland, William S. Robust locally weighted regression and smoothing scatterplots. *J Am Stat Assoc.* 1979; 74:829–836.
7. Del Favero, Simone, Monaro, Marco, Facchinetti, Andrea, Tagliavini, Alessia, Sparacino, Giovanni, Cobelli, Claudio. Real-time detection of glucose sensor and insulin pump faults in an artificial pancreas. 2014 International Federation of Automatic Control World Congress IFAC; 2014. p. 1941-1946.

8. Dempster, Arthur P., Laird, Nan M., Rubin, Donald B. Maximum likelihood from incomplete data via the em algorithm. *J R Stat Soc Ser B (Methodological)*. 1977; 39:1–38.
9. Dunia, Ricardo, Joe Qin, S. Joint diagnosis of process and sensor faults using principal component analysis. *Control Eng Pract*. 1998; 6:457–469.
10. Facchinetti, Andrea, Del Favero, Giovanni Sparacino Simone, Cobelli, Claudio. Modeling transient disconnections and compression artifacts of continuous glucose sensors. *Diabetes Technol Ther*. 2016; 18:264–272. [PubMed: 26882463]
11. Facchinetti, Andrea, Favero, Simone, Sparacino, Giovanni, Cobelli, Claudio. Anonline failure detection method of the glucose sensor-insulin pump system: improved overnight safety of type-1 diabetic subjects. *IEEE Trans Biomed Eng*. 2013; 60:406–416. [PubMed: 23193300]
12. Feng, Jianyuan, Turksoy, Kamuran, Samadi, Sediqeh, Hajizadeh, Iman, Cinar, Ali. Hybrid online sensor error detection and functional redundancy for artificial pancreas control systems. *IFAC-PapersOnLine*. 2016; 49:753–758.
13. Ghahramani, Zoubin, Beal, Matthew J. *Graphical Models and Variational Methods Advanced Mean Field Method—Theory and Practice*. MIT Press; 2000.
14. Helton, Kristen L., Ratner, Buddy D., Wisniewski, Natalie A. Biomechanics of the sensor-tissue interface—effects of motion, pressure, and design on sensor performance and the foreign body response—part I: theoretical framework. *J Diabetes Sci Technol*. 2011; 5:632–646. [PubMed: 21722578]
15. Herrero, Pau, Calm, Remei, Vehí, Josep, Armengol, Joaquim, Georgiou, Pantelis, Oliver, Nick, Tomazou, Christofer. Robust fault detection system for insulin pump therapy using continuous glucose monitoring. *J Diabetes Sci Technol*. 2012; 6:1131–1141. [PubMed: 23063040]
16. Hovorka, Roman. Continuous glucose monitoring and closed-loop systems. *Diabetic Med*. 2006; 23:1–12.
17. Jazwinski, Andrew H. *Stochastic Processes and Filtering Theory*. Courier Corporation; 2007.
18. Kim, Sanghong, Okajima, Ryota, Kano, Manabu, Hasebe, Shinji. Development of soft-sensor using locally weighted pls with adaptive similarity measure. *Chemom Intell Lab Syst*. 2013; 124:43–49.
19. Kobayashi, Takahisa, Donald Simon, L. Hybrid Kalman filter approach for aircraft engine in-flight diagnostics: sensor fault detection case. *J Eng Gas Turbines Power*. 2007; 129:746–754.
20. McFarlane, Duncan C., Glover, Keith. *Robust Controller Design Using Normalized Coprime Factor Plant Descriptions*. Springer; 1990.
21. McIntosh, Ian Blair Dwight. *A Model-Based Fault Detection and Diagnosis Methodology for Hvac Subsystems*. University of Wisconsin; Madison: 2000.
22. Raich, Anne, Cinar, Ali. Statistical process monitoring and disturbance diagnosis in multivariable continuous processes. *AIChE J*. 1996; 42:995–1009.
23. Romessis, C., Mathioudakis, K. *ASME Turbo Expo 2002 Power for Land, Sea, and Air*. American Society of Mechanical Engineers; 2002. Setting up of a probabilistic neural network for sensor fault detection including operation with component faults ing up of a probabilistic neural network for sensor fault detection including operation with component faults; p. 101-108.
24. Savitzky, Abraham, Golay, Marcel JE. Smoothing and differentiation of data by simplified least squares procedures. *Anal Chem*. 1964; 36:1627–1639.
25. Talebi, Heidar A., Khorasani, Khashayar, Tafazoli, Siamak. A recurrent neural-network-based sensor and actuator fault detection and isolation for nonlinear systems with application to the satellite's attitude control subsystem. *IEEE Trans Neural Netw*. 2009; 20:45–60. [PubMed: 19068428]
26. Ting, Jo-Anne, Theodorou, Evangelos, Schaal, Stefan. Learning an outlier-RobustKalman filter, in: *Machine Learning: Ecml 2007*. Springer; 2007. p. 748-756.
27. Ündey, Cenk, Tatara, Eric, Çınar, Ali. Real-time batch process supervision by integrated knowledge-based systems and multivariate statistical methods. *Eng Appl Artif Intell*. 2003; 16:555–566.
28. Venkatasubramanian, Venkat, Rengaswamy, Raghunathan, Yin, Kewen, Kavuri, Surya N. A review of process fault detection and diagnosis: part I: quantitative model-based methods. *Comput Chem Eng*. 2003; 27:293–311.

29. Wang, Shengwei, Cui, Jingtian. Sensor-fault detection, diagnosis and estimation for centrifugal chiller systems using principal-component analysis method. *Appl Energy*. 2005; 82:197–213.
30. Wei, Xiukun, Verhaegen, Michel, van Engelen, Tim. Sensor fault detection and isolation for wind turbines based on subspace identification and kalman filter techniques. *Int J Adapt Control Signal Process*. 2010; 24:687–707.

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

Fig. 1. Characteristics of sensor signals in different clusters. (A: IDP, B: DIP, C: IP, D: DP, E: SP; blue (dashed) lines in A and B indicate the tangents of start and end step of sensor signal; blue star indicates the intersection of the two tangents). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Fig. 2. Flow diagram of clustering algorithm for LW-PLS. S_{TI} and S_{TD} are the threshold for IP and DP clusters.

Fig. 3.
Examples for nominal angle Φ construction.

Fig. 4.

Threshold change for $S_k - S_{k-1}$ for different S^N and Φ^{max} (A. $S^N = 10$, $\Phi^{max} = [10: 5: 30]$; B. $S^N = [5: 5: 25]$, $\Phi^{max} = 15$).

Fig. 5.
Sensor signal retuning candidate selection using NAA.

Fig. 6.
Flow diagram of SED&FR algorithm.

Fig. 7. Examples of SJCC online smoothing (In cases A and B, the CGM signal is not smoothed because the CGM signal is very close to the calibration BG, In cases C and D, the CGM signal is detected as SJCC and smoothed).

Fig. 8.
Flow diagram of SED&FR algorithm with SJCC smoothing procedure.

Fig. 9.
Examples of online CGM signal reconciliation (A. Missing signal, B. Signal stuck, C. Spike, D. Drift change, E. Step change, F. PISA).

Fig. 10.
Example of online CGM signal reconciliation with four errors close to each other.

Table 1

Statistical information of the clinical data.

G_{ave} mg/dl	G_{std} mg/dl	G_{max} mg/dl	G_{min} mg/dl	G_{lave} mg/dl	G_{lstd} mg/dl/sample	G_{lmax} mg/dl/sample	G_{lmin} mg/dl/sample	ins_{ave} U/hour	ins_{std} U/hour	ins_{max} U/hour	ins_{min} U/hour
146	53	394	39	5.15	5.28	82	0	1.65	4.58	30	0

Table 2

Summary of sensor fault detection and reconciliation results (SGF filtered data) M_e : fault magnitude, EAP: error appearance percentage, EDRS: error detected and reconciled successfully, EDRF: error detected but reconciliation failed, FN: false negative detection (missed errors), FP: false positive detection, S: sensitivity, SRR: successful reconciliation rate, FDR: false detection ratio.

EAP(%)	0.1				0.5				1				5			
	10	20	30	40	10	20	30	40	10	20	30	40	10	20	30	40
EDRS	61	64	92	100	286	376	446	460	550	764	909	941	2087	3073	3551	3780
EDRF	3	7	2	7	26	17	33	25	54	81	55	49	366	361	379	357
FP	46	43	46	40	96	134	125	105	193	243	227	182	637	878	992	806
FN	48	28	27	24	205	138	124	93	413	266	250	186	1656	1028	988	841
S	0.57	0.72	0.78	0.82	0.60	0.74	0.79	0.84	0.59	0.76	0.79	0.84	0.60	0.77	0.80	0.83
SRR	0.95	0.90	0.98	0.93	0.92	0.96	0.93	0.95	0.91	0.90	0.94	0.95	0.85	0.89	0.90	0.91
FDR	0.42	0.38	0.33	0.27	0.24	0.25	0.21	0.18	0.24	0.22	0.19	0.16	0.21	0.20	0.20	0.16

Table 3

Summary of results for different types of errors (SGF filtered data).

Error Type	Spike				Drift				Step change				PISA				Missing Signal	Signal stuck
	10	20	30	40	10	20	30	40	10	20	30	40	>10	10-20	20-30	>30		
M_e (%)																		
EDRS	491	871	1154	1154	208	340	612	903	369	958	1122	1216	57	255	415	534	4494	3440
EDRF	24	18	9	13	45	61	54	51	19	30	19	14	28	23	30	16	614	754
FN	839	288	137	99	1014	920	728	454	867	306	146	92	276	336	248	279	0	1060
S	0.38	0.76	0.89	0.92	0.20	0.30	0.48	0.68	0.31	0.76	0.89	0.93	0.24	0.45	0.64	0.66	1.00	0.80
SRR	0.95	0.98	0.99	0.99	0.82	0.85	0.92	0.95	0.95	0.97	0.98	0.99	0.67	0.92	0.93	0.97	0.88	0.82

Table 4

Summary of sensor fault detection and reconciliation results for errors (unfiltered data).

EAP(%)	0.1				0.5				1				5			
	10	20	30	40	10	20	30	40	10	20	30	40	10	20	30	40
EM(%)	40	58	82	98	224	330	397	417	430	601	713	814	1882	2683	3130	3340
EDRS	7	9	20	6	58	51	69	50	141	108	99	96	680	673	608	564
EDRF	529	538	539	541	572	559	550	568	603	630	615	609	824	1011	941	929
FP	78	40	28	20	451	261	167	90	927	502	275	191	4368	2576	1506	924
FN	0.38	0.63	0.78	0.84	0.38	0.59	0.74	0.84	0.38	0.59	0.75	0.83	0.37	0.57	0.71	0.81
S	0.85	0.87	0.80	0.94	0.79	0.87	0.85	0.89	0.75	0.85	0.88	0.89	0.73	0.80	0.84	0.86
SRR	0.92	0.89	0.84	0.84	0.67	0.59	0.54	0.55	0.51	0.47	0.43	0.40	0.24	0.23	0.20	0.19
FDR																

Table 5

Summary of results for different types of errors (unfiltered data).

Error Type	Spike				Drift				Step change				PISA				Missing Signal	Signal stuck
	10	20	30	40	10	20	30	40	10	20	30	40	<10	10-20	20-30	>30		
M_e (%)																		
EDRS	435	873	1091	1218	132	334	526	841	416	956	1106	1212	53	242	335	536	3446	2383
EDRF	59	47	30	22	83	93	75	56	136	88	44	27	13	48	59	38	1577	1004
FN	972	451	233	175	1206	1054	830	547	918	386	207	169	283	421	350	235	0	1234
S	0.34	0.67	0.83	0.88	0.15	0.29	0.42	0.62	0.38	0.73	0.85	0.88	0.19	0.41	0.53	0.71	1.00	0.78
SRR	0.88	0.95	0.97	0.98	0.61	0.78	0.88	0.94	0.75	0.92	0.96	0.98	0.80	0.83	0.85	0.93	0.69	0.77