Qualitative Path Estimation: A Fast and Reliable Algorithm for Qualitative Trend Analysis

Kris Villez
Eawag, Process Engineering, Spike, Überlandstrasse 133, P. O. Box 611, Dübendorf, CH-8600, Switzerland

DOI 10.1002/aic.14736
Published online February 6, 2015 in Wiley Online Library (wileyonlinelibrary.com)

Fault detection and identification is challenged by a lack of detailed understanding of process dynamics under anomalous circumstances as well as a lack of historical data concerning rare events in a typical process. Qualitative trend analysis (QTA) techniques provide a way out by focusing on a coarse-grained representation of time series data. Such qualitative representations are valid in a larger set of operating conditions and thus provide a robust way to handle the detection and identification of rare events. Unfortunately, available methods fail when faced with moderate noise levels or result in rather large computational efforts. For this reason, this article provides a novel method for QTA. This leads to dramatic improvements in computational efficiency compared to the previously established shape constrained splines method while the accuracy remains high. © 2015 American Institute of Chemical Engineers *AIChE J*, 61: 1535–1546, 2015
Keywords: batch process monitoring, change point detection, fault diagnosis, qualitative trend analysis, segmentation

Introduction

One of the most challenging tasks within the field of process supervision and control is that of fault diagnosis. Amongst others, the successful execution of fault diagnosis is challenged with (1) small amounts of data corresponding to faulty process conditions, (2) limited information about the root causes of recorded faults, and (3) poor understanding of process dynamics and causal relationships under abnormal conditions. Classic approaches to the fault detection and identification challenge have focused on defining normalcy by means of (a) first principles mechanistic models or (b) data mining methods. In principle, one can use such models to detect and interpret deviations from normal operation. This can be challenging, however. For example, typical observer-based methods require observability of a state or signature residual associated with each type of fault to identify its cause.[1] Other observer-based methods assume that the fault symptoms can be described by linear functions of their magnitude.[2] The usefulness of data mining methods is particularly limited when rare events are not present in the data used for fault modeling. In the light of these challenges, the so called qualitative approach to fault diagnosis is very interesting. In this case, one deliberately describes the process and/or its anomalies by means of coarse-grained qualitative simulation models or qualitative features.[3,4] The underlying idea is that such qualitative models and features can be extrapolated much further than a quantitative process model or data characterization. As such, limited assumptions need to be made regarding the process' behavior under previously unseen circumstances. In addition, this also means that a limited number of fault occurrences can still lead to an accurate, though imprecise, description of their behavior.

A popular set of qualitative methods for fault diagnosis is referred to as qualitative trend analysis (QTA) by which a time series is divided into time windows, called episodes, on the basis of the signs of its derivatives. Note that the links between QTA methods and classic segmentation methods have been rather weak so far.[5] The use of the Viterbi algorithm in this article connects the two fields in a stronger fashion. In the case of QTA methods, the sign of a measured signal and/or one or more of its derivatives are evaluated and used for further interpretation.[4,6] The resulting segmentation is referred to as a qualitative representation (QR) and its constituting segments are called episodes. Within such episodes, the sign of the analyzed signal and/or one or more derivatives does not change. Most typically, the first and second derivative are of interest as changes in their sign can be identified easily by the human eye. Indeed, in most engineering applications one attempts to replace a tenuous visual data inspection by an automated algorithm which mimics signal analysis as performed by the human eye. This also explains why many QTA techniques are rooted in artificial intelligence research. Research of the previous century has resulted in a wide variety of QTA methods. A number of techniques is based on archetypal artificial intelligence techniques such as artificial neural networks[7] or popular time series analysis techniques such as wavelet analysis or hidden Markov models (HMMs).[8–11] These methods consist of a two-step procedure (see Figure 1). In the first step, quantitative methods are used to obtain an abstraction of the data series. This can be based on several bases such as the use of a lossless continuous wavelet transformation[8,10]; a classification neural network[7]; or the identification of piece-wise

Corresponding concerning this article should be addressed to K. Villez at kris.villez@eawag.ch.

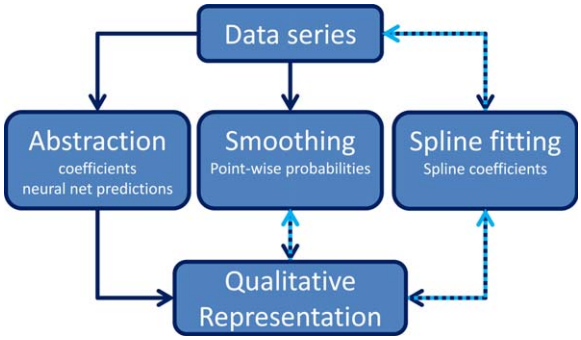

Figure 1. A schematic overview of available techniques for QTA.

Left: Traditional methods provide a one-way path from data to QRs; Right: The SCS method permits simulation of data according to the QR; Center: The newly proposed method has the ability to simulate features but not data. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

polynomial functions.[12,13] In a second step, the quantitative result is abstracted into a qualitative features. For instance, the signs of the wavelet coefficients are interpreted by a heuristic rule,[8] the quantitative neural network outputs are rounded to the closest target class,[7] or the signs of the derivative of piece-wise polynomials are evaluated.[12] This second step, in contrast to the first, is typically based on intuition and is often lacking in terms of statistical rigor or global optimality. In addition, the information flow in these algorithms is one-directional, that is, from the original data via an intermediate quantitative description of the data series to the QR. It is generally impossible to reverse these approaches, for example, to simulate data in accordance with a hypothesized QR. This also means that there is no joint likelihood function available for the QR and the original data. As such, these methods can be classified as discriminative methods.[14,15] Most importantly, this means the quality of the resulting QR cannot evaluated easily, except by comparing to the ground truth which is available in benchmark simulation studies but not in full-scale applications.

The above methodological and empirical observations were the main motivation for the development of a globally optimal method for QTA based on shape constrained splines (SCS).[16] The resulting accuracy was favorably compared to the previously studied wavelet-based method.[17] Interestingly, this method provides a joint likelihood for (1) the QR, (2) a number of spline function coefficients, and (3) the measured data. As a consequence, the applied model allows –in principle– to simulate data in accordance to any hypothesized qualitative segmentation. Because the information flow can be reversed, this method is labeled as a generative method.[14,15] An effective way of sampling the distribution described by the obtained likelihood function is, however, not available yet. For this reason, a maximum a posteriori likelihood optimization has been applied so far.[16]

This SCS method allows to obtain the best segmentation of an univariate time series into so called episodes which are characterized by a specific combination for the signal's sign and one or more of its derivatives. These combinations of signs are known as primitives. The application of the optimization method requires that one knows the sequence of primitives of the analyzed data series. This means that only the locations in the data series where the primitive or shape changes are optimized. When the exact sequence of primitives is not known, one can execute the optimization for every candidate sequence and then select the best sequence based on a measure of fit. This allows to use the technique for batch process diagnosis based on qualitative information alone. While an excellent performance is reported, the SCS method is very slow as it solves the nonlinear segmentation problem by means of a deterministic optimization scheme. For example, up to 20 h were needed on a modern desktop computer for a one-time execution of batch fault diagnosis as studied in this article. The computational requirements do not scale well either with the length of the data series nor the number of identified episodes. As such, the SCS method represents an extreme approach to QTA in the sense that global optimality is traded off against high computational efforts. With the SCS method at one side (globally optimal but slow) and a number of alternative methods at the other (suboptimal yet fast) within the spectrum of the QTA methods, one begs to question whether an intermediate solution is available, possibly trading computational cost off against reasonable accuracy. The author contends that such method can be devised as a two-step procedure by combining an existing algorithm for univariate series smoothing, such as kernel regression, and a path estimation method, such as the Viterbi algorithm. The method based on such combination, further referred to as qualitative path estimation (QPE), has both discriminative and generative properties (see Figure 1). Similar to the SCS method, one is again required to know which sequences of primitives are feasible.

The next section describes the applied methods. This is followed by the description of the data set used for benchmarking. The results of this benchmarking study are presented and discussed in two following sections. The last section summarizes the main conclusions drawn from this study.

Methods

The following paragraphs describe the prerequisite concepts and terminology, the proposed method, and the applied performance metrics. Acronyms, notations, and symbols are listed in Tables 1–3.

General concepts and terminology

Methods for QTA are developed to segment data series, mostly univariate time series, into so called episodes. These episodes are defined as contiguous and consecutive windows over the argument values of the data series within which the signs of the analyzed data series and/or their derivatives are judged constant. These episodes are characterized by a start point, an end point, and a so called primitive. The start point of one episode is the end point of the previous one

Table 1. List of Acronyms

Acronym	Full Wording
HMM	Hidden Markov Model
MAD	Mean Absolute Deviation
NOC	Normal Operation Conditions
QPE	Qualitative Path Estimation
QR	Qualitative Representation
QS	Qualitative Sequence
QTA	Qualitative Trend Analysis
SCS	Shape Constrained Splines
WLS	Weighted Least Squares

Table 2. Notation

Notation	Meaning
a, σ	scalar
$\boldsymbol{a}, \boldsymbol{A}_{.j}, \boldsymbol{\sigma}, \boldsymbol{\Sigma}_{.j}$	column vector
$\boldsymbol{A}, \boldsymbol{\Sigma}$	matrix

(contiguity) and are further referred to as transition points. The primitive represents a unique combination of signs for the analyzed data series and/or its derivative. Most typically, one is concerned with the signs of the first and second derivatives resulting in so called triangular primitives.[8] The primitives relevant in this study (A, B, C, and D) and their corresponding signs for the first and second derivatives are displayed in Figure 2.

A sequence of episodes is also known as a QR. A sequence of episodes for which the primitives and their order are specified but the transition times are unknown is known as a qualitative sequence (QS). A QS corresponding to l episodes can be represented as a vector of primitives, $\mathbf{q}=(q_1, q_2, \ldots, q_t, q_{t+1}, \ldots, q_l)^T$. Transitions are only permitted between the following pairs of primitives: (A,B), (B,C), (C,B), (C,D), (D,A), and (A,D). Any other transition between primitives would imply a discontinuity of the 1st derivative which neither the SCS method nor the proposed QPE method can deal with. The proposed method in this article specifically addresses the search for optimal values for transition points in a QR given one of these QSs. As for the SCS method, this method can also be used to determine the most likely QS following optimization of the transition points in each QR.

Table 3. Symbols

Symbol	Variables
α	Normalization constant
β	Polynomial coefficient
γ	Base-2 logarithm of τ
δ	Kronecker delta
κ	Primitive
λ	Transition likelihood
π_0	Prior likelihood
σ	Variance
τ	Kernel half-width
Λ	Likelihood
$\boldsymbol{\Sigma}$	Variance-covariance matrix
a	Degree of derivative
c	Condition index
d	Polynomial term order
f	Polynomial function
$f^{(a)}$	ath derivative function
h	Data point index
i	Data point index
j	Batch index
k	Kernel weight
l	Length of primitive sequence
m	Number of scenarios
n	Length of data series
o	Order of polynomial
p	Discrete (predecessor) state
p_M	Maximum likelihood predecessor state
q	Primitive
s	Discrete state
s_{path}	Discrete state on maximum likelihood path
t	Discrete (target) state
u	Integrand
x	Argument
y	Measurement
z	Distance
$\mathbf{A}$	Projection matrix
$\mathbf{P}$	Maximum likelihood predecessor states
$\mathbf{T}$	Transition likelihood matrix
$\mathbf{K}$	Kernel weight matrix

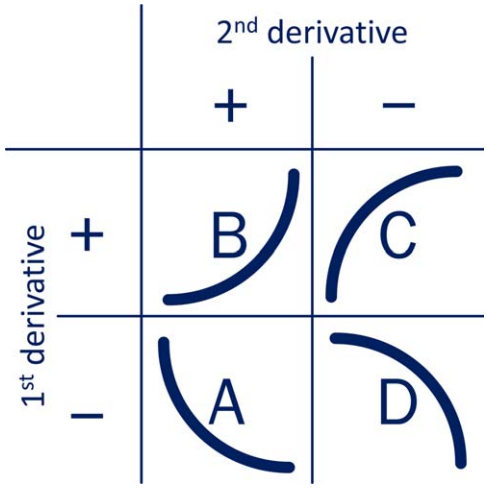

Figure 2. Triangular primitives according to the signs of the 1st and 2nd derivative: A = anti-tonic convex, B = isotonic convex, C = isotonic concave, D = anti-tonic concave.

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

Algorithm: Qualitative path estimation

The newly proposed method is composed of a smoothing step to compute point-wise probabilities for primitives followed by application of the Viterbi algorithm for state path estimation. These two steps are joined by matching each primitive in a QS with a discrete state in a linear Markov chain, further referred to as a qualitative state. In the smoothing step, the information flow is one-directional from the original data to these point-wise probabilities (see Figure 1). This is the discriminative part of the method. The point-wise probabilities are then further interpreted by finding the most likely sequence of so called qualitative states given the point-wise probabilities for these states and a HMM. This last step is optimal in the sense that the found sequence of states is the maximum likelihood sequence for the given point-wise probabilities and conditional to a given HMM. The information flow can be reversed as point-wise probabilities for the primitives can be simulated given a sequence of qualitative states and the HMM. This is thus the generative part of the method. The following paragraphs explain the method in mathematical detail.

Qualitative State Probabilities via Kernel Regression. The first step of the QPE algorithm consists of kernel regression. This method is based on repeated fitting of a polynomial regression model in a moving-window approach. Consider an univariate data series consisting of n measurements, $\mathbf{y}(i)$, obtained at equidistant argument values, $\mathbf{x}(i)$. A standard kernel-based regression scheme[18] is used to smooth this data series. Only the essentials are described in what follows.

Practically, one fits a polynomial of second (quadratic) or higher degree in a window around each data point, i, by means of weighted least squares (WLS)

$$\min_{\beta_i} \; J(\boldsymbol{\beta}_i)=\sum_{1\le h\le n} k_{h,i}\cdot\left(\mathbf{y}(h)-f\left(z_{h,i},\boldsymbol{\beta}_i\right)\right)^2 \tag{1}$$

$$=\left(\mathbf{y}-f\left(\mathbf{z}_{.,i},\boldsymbol{\beta}_i\right)\right)^T\cdot\mathbf{K}_i\cdot\left(\mathbf{y}-f\left(\mathbf{z}_{.,i},\boldsymbol{\beta}_i\right)\right) \tag{2}$$

with

$$z_{h,i}=\mathbf{x}(h)-\mathbf{x}(i) \tag{3}$$

$$f\left(z_{h,i},\boldsymbol{\beta}_i\right)=\sum_{d=1}^{o}\boldsymbol{\beta}_i(d).z_{h,i}^{d-1} \tag{4}$$

$$\mathbf{K}_i(h,i)=\begin{cases} k_{h,i}, & \text{if } h=i \\ 0, & \text{otherwise}\end{cases} \tag{5}$$

In the above, $z_{h,i}$, is a distance measure between a given data point, $\mathbf{x}(h)$, and a reference data point, $\mathbf{x}(i)$. Naturally, this distance is zero when $h=i$. The polynomial is represented by $f(z,\boldsymbol{\beta})$ with z the independent variable and $\boldsymbol{\beta}$ the vector of polynomial coefficients. In this study, a quadratic polynomial is used ($o=3$). The weights, $k_{h,i}$, are fixed *a priori* and decrease with increasing absolute values for the distances, $z_{h,i}$. To this end, so called kernel functions are popular. In this study, the tri-cube kernel is used

$$k_{h,i}=\begin{cases}\left(1-\left|\dfrac{z_{h,i}}{\tau}\right|^3\right)^3, & |z_{h,i}|\le\tau \\ 0, & \text{otherwise}\end{cases} \tag{6}$$

The tri-cube kernel is symmetrical and leads to zero-valued weights for any absolute distance larger than a critical value, τ, which is further referred to as the kernel half-width. Its application results in a moving window approach to the regression problem.

The minimization in Eq. 2 is executed for every point, i, in the data series. As such, n vectors of polynomial coefficients result. The WLS-optimal coefficient values are obtained analytically as follows

$$\hat{\boldsymbol{\beta}}_i=\left(\mathbf{z}_{.,i}^T\cdot\mathbf{K}_i\cdot z_{.,i}\right)^{-1}\cdot\mathbf{z}_{.,i}^T\cdot\mathbf{K}_i\cdot\mathbf{y}=\mathbf{A}_i\cdot\mathbf{y} \tag{7}$$

The corresponding derivatives of the estimated polynomial functions are evaluated in the corresponding window centers as follows

$$\hat{f}_i^{(a)}=\left.\frac{\partial^a f(z,\beta)}{\partial z^a}\right|_{z=0,\;\beta=\hat{\beta}_i}=a!\cdot\hat{\boldsymbol{\beta}}_i(a+1) \tag{8}$$

Assuming that the measurements, $\mathbf{y}(i)$, are characterized by independent and identically distributed measurement errors drawn from a multivariate Gaussian distribution with zero mean and covariance matrix, $\boldsymbol{\Sigma}_y$, then the above estimates for the polynomial coefficients and derivatives are distributed normally. The estimate of their mean corresponds to the above computed values while the expected variance-covariance matrix of the estimated polynomial coefficients in point i is computed as follows

$$\boldsymbol{\Sigma}_{\boldsymbol{\beta},i}=\mathbf{A}_i\cdot\boldsymbol{\Sigma}_y\cdot\mathbf{A}_i^T \tag{9}$$

Without loss of generality, the measurement error covariance matrix is assumed diagonal and its diagonal elements, $\sigma_{y,i}$, are assumed to be invariant

$$\boldsymbol{\Sigma}_y(h,i)=\begin{cases}\sigma_{y,i}=\sigma_y, & h=i \\ 0, & \text{otherwise}\end{cases} \tag{10}$$

The diagonal elements of the coefficient covariance matrix, $\boldsymbol{\Sigma}_{\boldsymbol{\beta},i}$, correspond to point-wise variances of the polynomial coefficients

$$\sigma_{\boldsymbol{\beta}(d),i}=\boldsymbol{\Sigma}_{\boldsymbol{\beta},i}(d,d) \tag{11}$$

Based on Eqs. 8 and 11, the variance of the above-computed derivative estimates (Eq. 8) can thus be computed as follows

$$\sigma_{f^{(a)},i}=(a!)^2\cdot\boldsymbol{\Sigma}_{\boldsymbol{\beta},i}(a+1,a+1) \tag{12}$$

The estimated distributions for the derivatives in $\mathbf{x}(i)$ can now be written as follows

$$f_i^{(a)}\sim N\left(\hat{f}_i^{(a)},\sigma_{f^{(a)},i}\right) \tag{13}$$

The probability that a derivative is positive (resp., negative) is obtained by integrating the probability mass from zero to infinity (resp., minus infinity to zero). As long as the measurement variances, $\sigma_{y,i}$, are nonzero, one can assume that the likelihood for zero-valued derivatives can safely be assumed equal to zero

$$\Lambda\left(f_i^{(a)}=0\right)=0 \tag{14}$$

Then, one can write the likelihoods for a positive, resp. negative, value for the derivative as follows

$$\Lambda_i^{(a)}(+)=\Lambda\left(f_i^{(a)}>0|\mathbf{y}\right)=\int_{u=0}^{+\infty}\exp\left(-\frac{\left(u-\hat{f}_i^{(a)}\right)^2}{\sigma_{f^{(a)},i}}\right) \tag{15}$$

$$\Lambda_i^{(a)}(-)=\Lambda\left(f_i^{(a)}<0|\mathbf{y}\right)=\int_{u=-\infty}^{0}\exp\left(-\frac{\left(u-\hat{f}_i^{(a)}\right)^2}{\sigma_{f^{(a)},i}}\right) \tag{16}$$

The probability for a particular primitive, $\boldsymbol{\kappa}(i)$, in a given point, $\mathbf{x}(i)$, is then computed by computing the product of probabilities for individual derivatives. The relevant probabilities in this work (cfr. Figure 2) are computed as follows

$$\Lambda(\boldsymbol{\kappa}(i)=A|\mathbf{y})=\Lambda_i^{(1)}(-)\cdot\Lambda_i^{(2)}(+) \tag{17}$$

$$\Lambda(\boldsymbol{\kappa}(i)=B|\mathbf{y})=\Lambda_i^{(1)}(+)\cdot\Lambda_i^{(2)}(+) \tag{18}$$

$$\Lambda(\boldsymbol{\kappa}(i)=D|\mathbf{y})=\Lambda_i^{(1)}(-)\cdot\Lambda_i^{(2)}(-) \tag{19}$$

$$\Lambda(\boldsymbol{\kappa}(i)=C|\mathbf{y})=\Lambda_i^{(1)}(+)\cdot\Lambda_i^{(2)}(-) \tag{20}$$

These probabilities are computed in each point, $\mathbf{x}(i)$, leading to a series of probabilities for each possible primitive in a point i conditional to the entire series of data, $\mathbf{y}$. These probabilities offer the advantage of a statistical assessment of qualitative states and, subsequently, QRs. The computation of these probabilities is discriminative in nature. Indeed, the applied models do not permit simulation of data according to these probabilities. Note that Eqs. 17–20 assume (erroneously) that the derivatives of different degree in a single point are uncorrelated. This could be improved by computing the qualitative state probabilities as integrals of multivariate Gaussian integrals rather than the product of univariate Gaussian

integrals. However, this deliberate approximation is more straightforward in most software packages and does not stand in the way of an effective algorithm, as will be shown below.

Maximum Likelihood Path Estimation via the Viterbi Algorithm. To optimize the transition locations in a QS, the Viterbi algorithm is applied. This algorithm is an optimal method to estimate the most likely sequence of discrete process states given a series of uncertain and indirect observations generated by a stochastic discrete-time process. It is based on an HMM which is generative in nature as one can simulate feasible state sequences and corresponding (uncertain) measurements.[19,20] Once more, only the essential elements are discussed here.

The Viterbi algorithm is theoretically optimal for segmentation when the process state evolves in time according to a first-order Markov process. Concretely, the expected likelihood for the process state at time i given likelihoods for each possible state at time $i-1$ is written as follows

$$\Lambda(\mathbf{s}(i)=t \mid i-1)=\sum_{p=1}^{q}\mathbf{T}_i(t,p)\cdot\Lambda(\mathbf{s}(i-1)=p \mid i-1) \quad (21)$$

In words, the likelihood that the process is in target state t at point i is a linear combination of the likelihoods of each possible predecessor state at time $i-1$. This linear combination is defined by the transition likelihoods, $\mathbf{T}_i(t,p)$, which determine the likelihood that the process will be in a target state t at time i conditional to the process being in the predecessor state p at time $i-1$. Equation 21 thus delivers a one step ahead prediction for the likelihoods, which are only dependent on the likelihoods for the directly preceding time point.

The above predictive model is completed with a sensor model. To make this possible, each Markov process state is matched with a primitive in the qualitative sequence, $\mathbf{q}$, so that $\mathbf{q}(t)$ indicates the primitive associated with the t-th discrete state of the linear Markov chain. This results in the following equivalence for (a) the likelihood of observed data conditional to the Markov state and (b) the likelihood of the same data conditional to the primitive associated with the considered Markov state

$$\Lambda(\mathbf{y}(i) \mid \mathbf{s}(i)=t)=\Lambda(\mathbf{y}(i) \mid \boldsymbol{\kappa}(i)=\mathbf{q}(t)) \quad (22)$$

In addition, the likelihood of a data point, $\mathbf{y}(i)$, conditional to the likelihood of a primitive at time i is set equal to the likelihood of said primitive conditional to the likelihood of the data point. More specifically, one writes

$$\Lambda(\mathbf{y}(i) \mid \boldsymbol{\kappa}(i)=\mathbf{q}(t))=\Lambda(\boldsymbol{\kappa}(i)=\mathbf{q}(t) \mid \mathbf{y}(i)) \quad (23)$$

The above equation inverts the dependence relationship between likelihoods. Indeed, the likelihood of a data point conditional to a qualitative state is considered equal to the computed likelihood of the qualitative state conditional to the observed data point. As such, this implicitly assumes that the prior likelihood for any measurement is uniform and that each qualitative state is equally likely *a priori*.

To make the Viterbi algorithm application possible, it is necessary to equal the conditional likelihood of a primitive to a single data point equal to the above-computed probability of this primitive to the whole data series as found in Eqs. 17–20

$$\Lambda(\boldsymbol{\kappa}(i)=\mathbf{q}(t) \mid \mathbf{y}(i))=\Lambda(\boldsymbol{\kappa}(i)=\mathbf{q}(t) \mid \mathbf{y}) \quad (24)$$

This approximation is rather severe and therefore deserves extra attention. By means of Eq. 24, one deliberately ignores autocorrelation effects on the estimates of derivatives and the subsequent point-wise probabilities for the primitives. In addition, one assumes that the probabilities for the primitives are independent of each other while, in reality, they are not. Ignoring such autocorrelation is necessary, however, for the Viterbi algorithm to be applicable. Despite this approximation, the resulting method works remarkably well as will be shown below.

The sensor equations, Eqs. 22–24, can now be summarized as

$$\Lambda(\mathbf{y}(i) \mid \mathbf{s}(i)=t)=\Lambda(\boldsymbol{\kappa}(i)=\mathbf{q}(t) \mid \mathbf{y}) \quad (25)$$

Given the above model, consisting of a first-order Markov process (Eq. 21) and an –assumed memoryless– Markov sensor (Eq. 25), one can compute the most likely sequence of states by means of the two-pass Viterbi algorithm. The first forward pass, consists of computing the most likely predecessor states for each possible target state, t, at each time point, i. One finds the most likely predecessor state, p, which maximizes the sum of the product of (1) the corresponding transition likelihood and (2) the probability of most likely path of states leading to state p at time $i-1$

$$\max_{p} \quad \mathbf{T}_i(t,p)\cdot\Lambda_{\text{path}}(\mathbf{s}(i-1)=p) \quad (26)$$

Consider p_{M} the selected most likely predecessor, then the most likely path leading to state t at time instant i has the following likelihood

$$\Lambda_{\text{path}}(\mathbf{s}(i)=t)=\alpha\cdot\Lambda(\mathbf{y}(i) \mid \mathbf{s}(i)=t)\cdot\mathbf{T}_i(t,p_{\text{M}})\cdot\Lambda_{\text{path}}(\mathbf{s}(i-1)=p_{\text{M}}) \quad (27)$$

with α a normalization factor. The maximizing value of p, p_{M}, is recorded for each time instant i and target state t, resulting in an $n \times l$ matrix, $\mathbf{P}$

$$\mathbf{P}(i,t)=p_{\text{M}}|_{i,t} \quad (28)$$

The forward pass of the Viterbi algorithm is initiated by providing a preset likelihood for each state at point $i=0$

$$\Lambda_{\text{path}}(\mathbf{s}(0)=t)=\boldsymbol{\pi}_0(t) \quad (29)$$

with $\boldsymbol{\pi}_0(t)$ representing the *a priori* probabilities for the state, t, at point $i=0$.

The backward pass of the Viterbi algorithm starts by selecting the final state in the estimated path, $\mathbf{s}_{\text{path}}(n)$, as the value for t which maximizes the path likelihood at the end of the time series, $\Lambda_{\text{path}}(\mathbf{s}(n)=t \mid \mathbf{y})$. Having selected this final state, the backward pass can begin. At this time, one traces the most likely predecessor states by going back in time, each time selecting the most likely predecessor of the currently selected state as follows

$$\mathbf{s}_{\text{path}}(i-1)=\mathbf{P}(k,\mathbf{s}_{\text{path}}(i)) \quad (30)$$

This Viterbi algorithm is completed when the first time instant is reached ($i=0$).

Modifications and requirements for data series segmentation with a known qualitative sequence

To enable the use of the above algorithm for segmentation, the following modifications and restrictions are implemented in this work:

1. The sequence of primitives and the corresponding Markov process states are assumed to be known.

2. The Markov process is constrained to be a linear chain without cycles by setting all elements in $\mathbf{T}_i$ equal to zero

except on the diagonal and the elements just right of this diagonal, for example

$$\mathbf{T}_i(p,t)=\begin{cases}1-\lambda_i(p), & p=t\\ \lambda_i(p), & p=t-1\\ 0, & \text{otherwise}\end{cases} \tag{31}$$

3. Without loss of generality the implemented transition likelihoods, $\lambda_i(p)$, are considered invariant with respect to time, process state and selected Markov chain. In addition, the value for λ is set to 1, so that

$$\forall p=1\ldots q,\ \forall i=1\ldots n\ :\ \lambda_i(p)=\lambda=1 \tag{32}$$

As a consequence, the transition likelihood matrices, $\mathbf{T}_i$, and the corresponding HMM, are also invariant

$$\forall i=1\ldots n\ :\ \mathbf{T}_i=\mathbf{T}_{i-1}=\mathbf{T} \tag{33}$$

Importantly, this particular choice for the transition likelihoods means that the Markov process model defines the chronological order of the qualitative states but does not hold prior information about the location of the state transitions. This also means that the *a priori* likelihood for any path generated by any Markov process with transition likelihoods as above is the same. Indeed, the values for $\mathbf{T}_i(t,p)$ in Eq. 21 are always one (1) for any feasible path. As such, the fault diagnosis exercise is executed in an uninformative Bayesian setting, apart from the *a priori* definition of the qualitative sequence, $\mathbf{q}$, and the associated linear Markov chain.

4. The Viterbi algorithm is modified by constraining the selected path so that the first and last states in the sequence are equal to the first and last state in the linear Markov chain. Practically, this means the backward pass is initiated with the last state in the chain rather than the state corresponding to the maximum value for the corresponding path likelihood. In addition, the likelihood at time zero (0) for the first state in the linear chain is set to one ($\boldsymbol{\pi}_0(1)=1$) while all other likelihoods are set to zero ($\forall j>0:\boldsymbol{\pi}_0(j)=0$). This approach ensures that the likelihood associated with the entire qualitative sequence is computed and not a likelihood corresponding to only a part of this sequence.

The ultimate QR now results by finding those segments in the state path $\mathbf{s}_{\text{path}}$ where the selected state does not change. Each time a pair of subsequent selected states is different from each other, a corresponding transition point is set halfway between the argument values corresponding to this pair and further referred to as $\hat{\mathbf{x}}_{\text{trans}}$. This completes the execution of the QPE algorithm.

Modifications and requirements for batch fault diagnosis

The QPE algorithm can also be applied for batch process fault diagnosis. To do so, one needs to associate each fault condition with a unique qualitative sequence *a priori*. Practically, the following setup is used:

1. Each possible condition, c, is associated with a specific qualitative sequence, $\mathbf{q}_c$, and associated Markov process described by a corresponding transition likelihood matrix, $\mathbf{T}_{c,i}$. As before, each primitive in each sequence, $\mathbf{q}_c(t)$, corresponds to a single state, t, in the corresponding Markov chain. Eqs. 31–33 hold for each transition likelihood matrix.

2. The QPE algorithm is executed for each of the available qualitative sequences, $\mathbf{q}_c$. The resulting transition points are referred to as $\hat{\mathbf{x}}_{c,\text{trans}}$ and the associated path likelihoods as $\Lambda_{\text{c,path}}$.

3. The fault diagnosis result is obtained by selecting the fault c with the highest likelihood for $\Lambda_{c,\text{path}}$.

Additional algorithm parameters

Two parameters defining the algorithm have been left undefined so far (τ and σ_y). To study the effect of these parameters, the following settings were applied:

Kernel Half-Width (τ). The kernel half-width was varied between 2 and 1024. More precisely, the applied values were set factors $2^{1/10}$ (= 1.072) apart as follows

$$\tau=2^{\gamma} \tag{34}$$

$$\text{with}:\ \gamma\in\{1.0,1.1,\ldots,9.9,10.0\} \tag{35}$$

Measurement Variance (σ_y). The above method requires knowledge of the measurement error variance, σ_y. In practice this is seldom available. For this reason, the method is tested in two settings:

1. Setting 1: Known variance. In the first setting, the measurement error variance is simply assumed known.

2. Setting 2: Estimated variance. In the second setting, the measurement error variance is replaced by its maximum likelihood estimate which is obtained as follows

$$\hat{\sigma}_y=\frac{1}{n}\cdot\sum_{i=1}^{n}\left(y_i-f(z_{h,i},\beta)|_{h=i,\beta=\hat{\beta}_i}\right) \tag{36}$$

$$=\frac{1}{n}\cdot\sum_{i=1}^{n}\left(y_i-f(0,\hat{\beta}_i)\right) \tag{37}$$

Performance evaluation

The following paragraphs describe the criteria used to evaluate the QPE method by means of the benchmark batch process simulation study. The QPE method is evaluated on the basis of its segmentation accuracy, fault diagnosis accuracy, and computational requirements.

Segmentation accuracy

The QPE algorithm is aimed at the identification of the most likely transition points given a single, predetermined qualitative sequence. An overall measure of accuracy is determined as the mean absolute deviation (MAD) between the ground truth transition points and their optimized values. The ground truth values, $\mathbf{x}_{\text{trans}}(t)$, are obtained by simple differentiation of the noiseless signals. Their estimates, $\hat{\mathbf{x}}_{\text{trans}}(t)$, are given by the QPE algorithm. The accuracy is measured as follows

$$\text{MAD}=\frac{1}{l-1}\sum_{t=1}^{l-1}|\hat{\mathbf{x}}_{\text{trans}}(t)-\mathbf{x}_{\text{trans}}(t)| \tag{38}$$

Importantly, the ground truth qualitative sequence needs to be known for the computation of this measure.

Classification accuracy

A second but no less important objective of this study is to evaluate the QPE algorithm as a tool for fault diagnosis. The fault diagnosis accuracy is evaluated as the fraction of correctly classified batches ($j=1\ldots m$)

$$\text{Fault Diagnosis Accuracy} = \frac{1}{m} \cdot \sum_{j=1}^{m} \delta(\mathbf{c}(j), \hat{\mathbf{c}}(j)) \tag{39}$$

with δ the Kronecker delta to indicate equality

$$\delta(\mathbf{c}(j), \hat{\mathbf{c}}(j)) = \begin{cases} 1, & \text{if } \mathbf{c}(j) = \hat{\mathbf{c}}(j) \\ 0, & \text{otherwise} \end{cases} \tag{40}$$

In addition to the above overall classification accuracy, condition-specific classification accuracies are also studied. To compute these measures, a predefined set of process conditions (c, normal and faulty operations) are assumed to be known *a priori* and are matched one-to-one with a qualitative sequence (q_c) and associated Markov process transition likelihood matrix (T_c). Furthermore, for each batch (j) the ground truth or reference process condition needs to be known ($\mathbf{c}(j)$).

Computational requirements

One of the main characteristics of the QPE method is that both of its algorithmic steps are of linear time complexity. As such, a favorable comparison with the SCS method is expected. The computational requirements for the QPE method are evaluated by tracking the time requirements for the complete execution of fault diagnosis as well as the portion associated with the kernel regression and Viterbi algorithm step. To this end, all computations were executed on a dedicated desktop machine (Intel^{R} Core^{TM} i7–4770 CPU, 3.40 GHz, 16.0 GB RAM).

Materials

Data set

The newly proposed algorithm is evaluated by means of a data set used previously for benchmarking of QTA methods.[16,17] This data set consists of simulated univariate batch time series. The use of simulations allows effective benchmarking against the ground truth instead of a subjective reference assessment. The use of such a benchmark data set was necessary to demonstrate and validate the rather poor performance of the wavelet-based algorithm studied in the first effective benchmarking study on QTA.[17] The continued development of new algorithms benefits from testing with the same data set because comparison between methods is straightforward in spite of the idealized features of simulated data sets. The analyzed data set consists of 150 noiseless data series for the Penicillin concentration obtained by simulation of a highly nonlinear batch fermentation process model created for benchmarking.[21] These data were originally simulated for the study of more conventional fault detection and diagnosis methods.[22] Each of these 150 batches are simulated according one to three process conditions (see also Figure 3). Batch 1–50 correspond to normal operation conditions (condition 1), batch 51–100 are simulated to a reduced saturation constant (condition 2), and for batch 101–150 the substrate feed rate is reduced (condition 3). Each simulated batch lasts 400 h and results in a noiseless vector of 5001 equidistant measurements (one measurement per 4.8 min). Each of the simulated conditions results in a distinct and unique QS for the (noiseless) time series, as indicated in Table 4. As such, fault diagnosis can be performed by evaluating which QS –and its corresponding process condition– is the most likely given a time series.

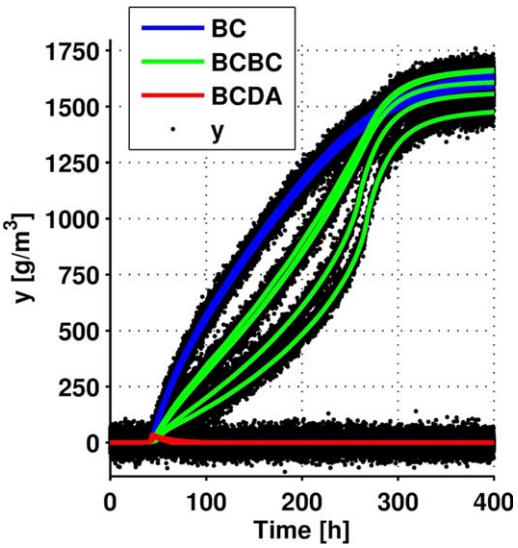

Figure 3. Noiseless simulations of every 10th batch (lines) and noisy data obtained with a measurement error variance of 10^3 $(g/m^3)^2$ (dots, standard deviation: 31.6 g/m^3).

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

The noiseless time series are corrupted by independent and identically distributed measurement errors from a zero mean univariate Gaussian distribution with five different measurement error variances, σ_y, namely: $0, 10, 100, 10^3$, and 10^4 $(g/m^3)^2$. The measurement error sequences for all batches indexed as j, $j+50$ and $j+100$ for $j = 1...50$, are the same up to a constant factor, namely the applied measurement standard deviations. This results in a total of 750 simulated time series (150×5). For each of these series, the QPE method is applied to identify the optimal transition points as well as to identify the most likely QS.

Implementation

All computations are implemented in the QPE toolbox for Matlab/Octave which is released as supplementary material to this manuscript together with an exemplary analysis of data as executed for this study.

Results

Data set

Figure 3 displays the measurement profiles for 15 batches, namely every 10th batch of the simulated Penicillin fermentation process (i.e., batch 10, 20,... 150) and with a measurement error variance of 10^3 $(g/m^3)^2$ (standard deviation: 31.6 g/m^3). Visual inspection easily confirms three different conditions of the process with distinct QSs as identified in Table 1. It remains to be evaluated whether the QPE method enables automated identification of these conditions.

Table 4. Overview of Simulated Conditions and Corresponding Qualitative Sequences (QSs)

Condition	Description	Batch Cycles	QS
1	NOC: Normal operation conditions	1–50	BC
1	Fault 1: Reduced saturation constant	51–100	BCBC
1	Fault 2: Reduced substrate feed	101–150	BCDA

Detailed example

The proposed method is demonstrated by means of a single batch simulation. The data of batch 51 in particular, with a measurement error variance of 10^3 $(g/m^3)^2$ (standard deviation: 31.6 g/m^3) were selected to this end. The noisy data are shown in the top panel of Figure 4.

In the first step of the algorithm, kernel regression is applied as a smoother to these data. A quadratic polynomial and a kernel half-width (τ) of 512 are applied to produce the results in Figure 4. The shown results are computed with the measurement error variance assumed known. The estimate of the first derivative is close to zero as well as rather uncertain at the beginning and end of the batch. In between, the first derivative is positive and more precise. Similarly, the estimate of the second derivative is uncertain and close to zero at the beginning and end of the batch and more precise in between. However, the pattern of its signs during the batch length is more complex. Roughly speaking, one identifies a positive segment, a segment where the second derivative hovers around zero, another positive segment and a negative segment. Based on the computed estimates for the derivatives and the associated variances, the point-wise probabilities for each of the primitives are calculated. The computed likelihoods for the primitives A, B, C, and D (see Figure 2) are shown in the top panel Figure 5. As expected from the visual inspection of the smoothed derivatives, the probabilities for the B and C primitives (increasing trends) are generally higher than those for A and D primitives (decreasing trends). An exception to this is observed at the beginning of the batch cycle where the A primitive appears to dominate.

The sequence of probabilities for each primitive are then interpreted by means of Viterbi path estimation. For demonstration purposes, the HMM corresponding to the (true) BCBC sequence is used. Practically, this means that the probabilities for A and D primitives are ignored. Indeed, with this HMM model, A and D primitives are considered impossible to achieve. The resulting state path, $\mathbf{s}_{path}$, is shown in the bottom panel of Figure 5 together with the corresponding QR. The optimized transition points are indicated in all panels of Figures 4 and 5 and show a pleasing match between visual interpretation of the data and the computed result.

The above steps were executed for all qualitative sequences (BC, BCBC, and BCDA) and for all considered kernel half-widths (τ). The resulting path likelihoods (Λ_{path}) are shown in Figure 6. It can be observed that the likelihood for

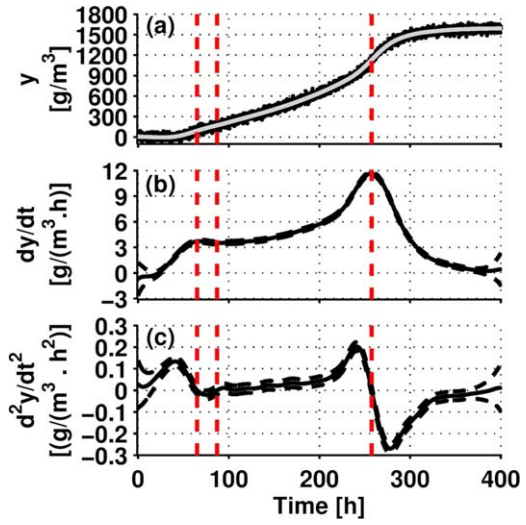

Figure 4. Smoothing of the concentration data by means of quadratic polynomial kernel regression: (a) simulated noisy data and smoothed kernel regression estimate; (b) Estimate and 3–σ point-wise confidence intervals for the 1st derivative; (c) Estimate and 3–σ point-wise confidence intervals for the 2nd derivative. Red dashed lines indicate the location of the identified transition point for the BCBC sequence.

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

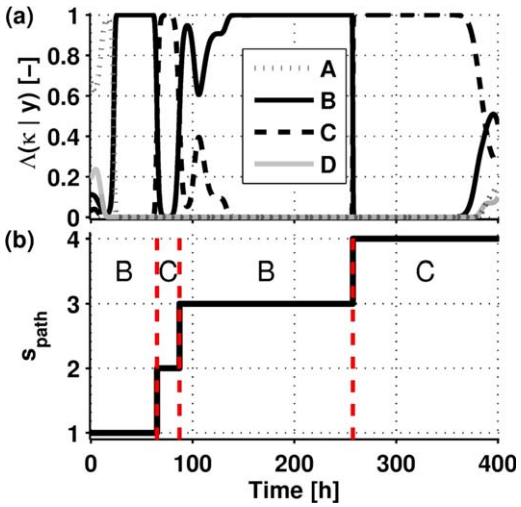

Figure 5. Viterbi path estimation for segmentation on the basis of qualitative features: (a) Point-wise probabilities for the primitives A, B, C, and D; (b) Identified state path. Red dashed lines indicate the location of the identified transition point for the corresponding BCBC sequence.

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

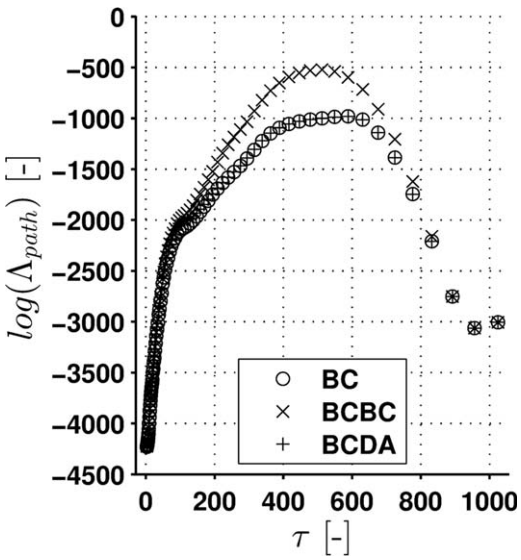

Figure 6. Log-likelihood as function of the kernel half-width (τ) for all three considered qualitative sequences.

The BCBC shape is clearly identified as the most likely over a wide range of kernel half-widths.

the BCBC sequences are generally higher than those for the BC and BCDA sequence, thus leading to a positive identification of the true sequence. Very low and very high kernel half-widths result in likelihoods which are lower and closer to each other. Inspection of the corresponding results leads to the conclusion that lower kernel half-widths lead to ineffective denoising and highly oscillating values for the derivatives, further leading to a reduced distinction between the maximum likelihoods obtained for each of the three QSs. High kernel half-widths lead to a rather high smoothing level, bringing all derivatives close to zero, once more leading to reduced discrimination between the resulting likelihoods.

Performance evaluation

Having demonstrated the proposed QPE algorithm with an exemplary time series, the results obtained for all time series are summarized in the next paragraphs. First the accuracy of the identified transition points, given the true QS, is discussed. Then the accuracy of the QPE method for fault diagnosis is evaluated. Finally, the required time for computation is studied.

Segmentation accuracy

The transition point accuracy for the QPE method was evaluated by means of the MAD (Eq. 38) for (1) five different noise levels, (2) ninety-one (91) different kernel half-widths, and (3) two ways of defining the standard deviation: (a) known and (b) estimated as in Eq. 36. Figure 7 displays the MAD values averaged over all batches and the batches with a single specific simulated condition. As indicated above, the correct fault condition or class and associated QS is considered known and only the transition points are sought for. The differences between the cases with known and estimated measurement variances are limited and hardly visible. However, both the noise variance and kernel half-width affect the MAD substantially. In the absence of noise ($\sigma_y = 0$) and kernel half-widths (τ) up to 776, the accuracy of transitions appears to increase linearly with the kernel half-width. This is the case regardless whether the overall MAD is considered or the MAD is inspected for each condition separately. Kernel half-widths of 832 and higher break this linear trend by delivering higher MAD values. For higher noise levels, the MAD curve appears roughly convex with an apparent minimum within the range of evaluated kernel half-widths. A reasonable performance can still be obtained for measurement error variances as high as 10^3 $(\mathrm{g/m^3})^2$ as the minimum overall MAD is 140.15 (corresponding to 11.21 h or 2.8% of the batch cycle length). At the highest noise level ($\sigma_y = 10^4$ $(\mathrm{g/m^3})^2$), the minimum overall MAD is 574.11, which amounts to 46 h simulated time or 11.5% of the batch cycle length. It is noted that the best performance for condition 1 remains good for all noise levels (minimal MAD below 100, equivalent to 8 h or 2% of the data series length). For condition 2 and 3, similar observations can be made when excluding the highest noise level (best MAD below 125, corresponding to 10 h or 2.5% of batch length). In general, increasing noise levels lead to increasing values for the optimal kernel half-width. As such, the negative impact of increasing noise levels on the accuracy can be compensated to some extent by stronger smoothing, which is not too surprising.

Classification accuracy

The fault diagnosis accuracy is simply computed as the fraction of batches to which the correct condition is associated by virtue of the most likely QS (Eq. 39). This fraction is computed for all 150 batches as well as for each set of 50 batches corresponding to a single simulated condition. All computed fractions are displayed in Figure 8. Interestingly, using a known or estimated standard deviation has almost no effect on the diagnosis accuracy. The fault accuracy is, however, sensitive to the applied kernel half-width (τ). For the three lowest noise variances (i.e., up to 10^2 $(\mathrm{g/m^3})^2$), a maximum accuracy of 100% can be achieved. For low noise

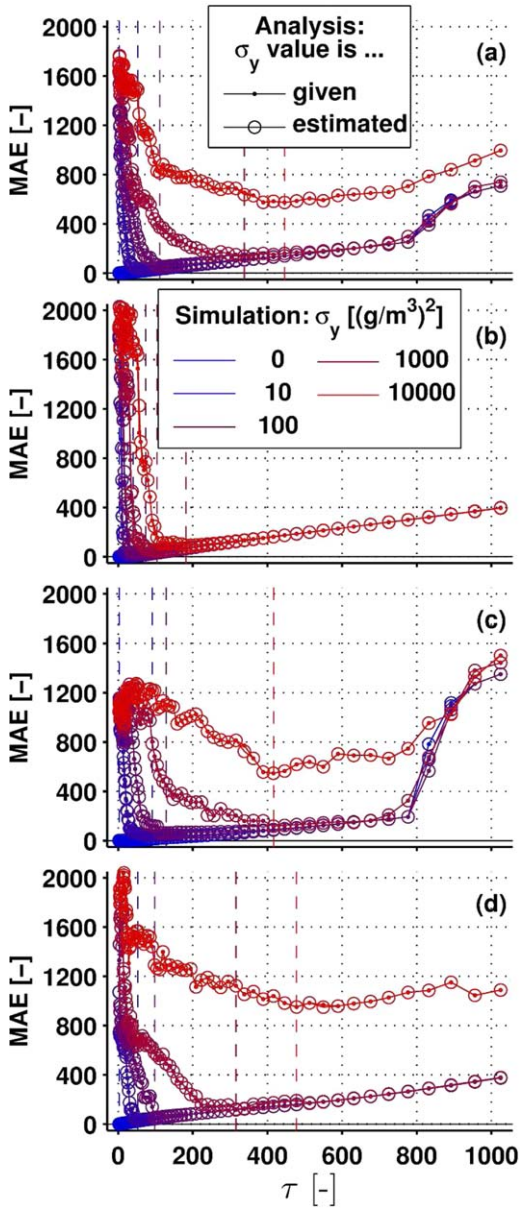

Figure 7. Accuracy of the transition locations as function of the kernel half-width for (a) all conditions (all batch cycles), (b) condition 1 (batch cycles 1–50), (c) condition 2 (batch cycles 51–100), and (d) condition 3 (batch cycles 101–150).

Results are shown for increasing noise levels (blue to red) and for different approaches to the estimation of the noise standard deviation. Vertical dashed lines indicate the minimum MAD values for each noise variance and corresponding kernel half-widths. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

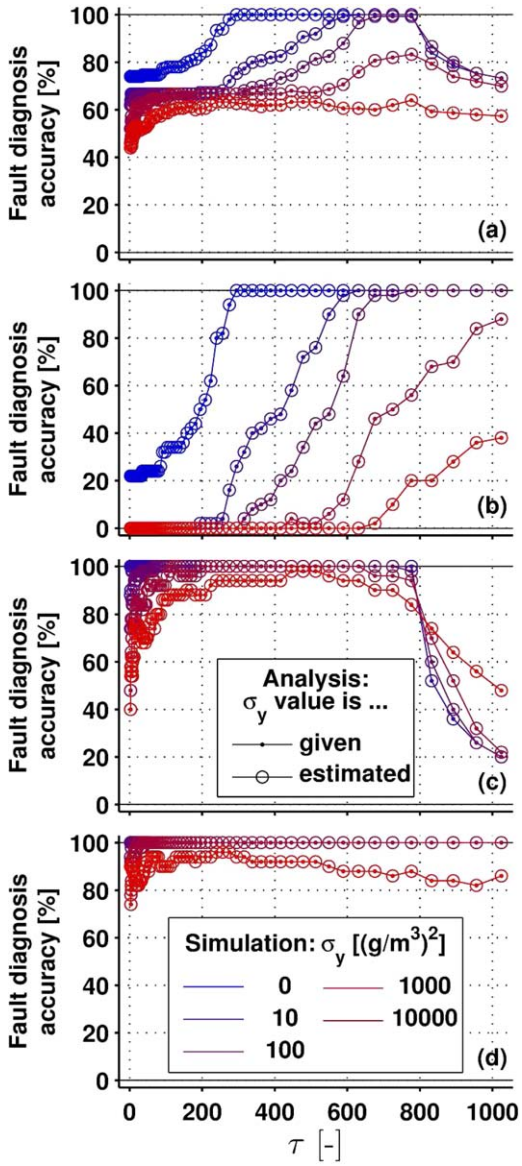

Figure 8. Observed fault diagnosis accuracy as a function of the kernel half-width for (a) all conditions (all batch cycles), (b) condition 1 (batch cycles 1–50), (c) condition 2 (batch cycles 51–100), and (d) condition 3 (batch cycles 101–150).

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.] Results are shown for increasing noise levels (blue to red) and the two settings for the noise standard deviation.

levels, this is also a robust result as a wide selection of possible kernel half-widths lead to this perfect classification. For higher noise variances the maximum overall accuracy is 83.34% ($\sigma=10^3$ $(g/m^3)^2$) and 64% ($\sigma=10^4$ $(g/m^3)^2$). Low kernel half-widths impact the overall accuracy most by increased misclassification of batches belonging to condition 1. Higher kernel half-widths impact the overall accuracy foremost by increased misclassification of batches belonging to condition 2. The classification performance for batches 101–150 (condition 3) remains 100% for all kernel half-widths as long as the noise level is low ($\sigma_y=0$ or 10 $(g/m^3)^2$). The same performance for condition 3 can be achieved at all noise levels, except the highest ($\sigma_y=10^4$ $(g/m^3)^2$). In this case, the maximal accuracy is 96% (for $\tau = 256$).

Computational requirements

The computational requirements of the proposed QPE method are a linear function of the number of data points. Indeed, both the kernel regression step and the Viterbi algorithm are algorithms with linear running time ($\mathcal{O}(n)$). In addition, the associated computational load does not depend on factors such as the noise variance or the considered QS. Inspection of the registered time needed to compute the QPE results does not challenge these expectations. Increasing the kernel half-width does lead to rather dramatic increases in computational requirements, however. As can be seen in Figure 9, this effect on the computational requirements is attributed to the kernel regression step as the computational demand for the Viterbi step is unaffected by the kernel half-width. More importantly, however, the total time requirement remains below 30 s in all cases.

Discussion

In this work, a new method for QTA, that is, segmentation of data series on the basis of shapes, has been proposed. The method can be compared favorably against other methods, such as the recently developed SCS method. The following paragraphs discuss identified strengths, weaknesses, opportunities, and threats.

Strenghts

Speed. Above all, the most important benefit of the proposed QPE method is its speed. Whereas complete fault diagnosis can requires up to 20 h for the SCS method,[16] the QPE method delivers fault diagnosis results in under 30 s for all considered cases. In addition, the required computational load was found independent of the noise level or the data values. This permits prediction of the required time for fault diagnosis with the QPE method, in contrast to the SCS

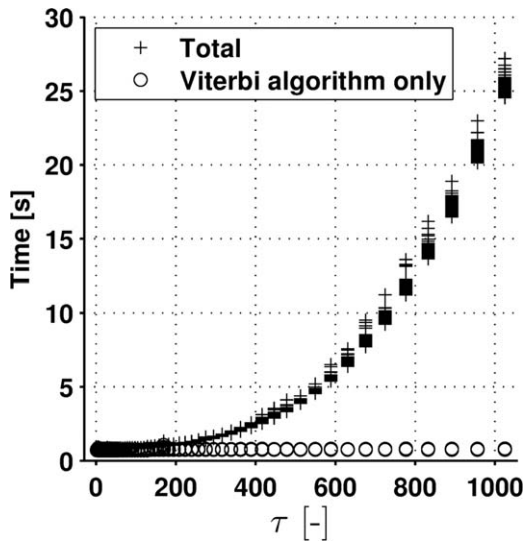

Figure 9. Time requirements for execution of fault diagnosis with the QPE algorithm.

For each kernel half-width, 1500 points are shown (150 batches × 5 noise variances × 2 approaches to measurement variance).

method where severe dependencies of the computational time on the simulated condition and noise level were found.

Accuracy. Despite a greatly advanced speed, the reported fault diagnosis performance for the QPE method remains high. For example, at a noise variance of $10^4(g/m^3)^2$, the overall accuracy for the QPE method is 83.3% (64.0%) whereas the SCS method resulted in a 85.2% (64.0%) accuracy. It is noted that the QPE method outperforms the wavelet-based method studied earlier[17] in both fault diagnosis accuracy and speed. Indeed, the wavelet-based method delivered, at its best, a classification performance of only 60% while the computational demand rises up to 2.5 min (150 s), about five times more than the worst case for the QPE method. Also, the need to estimate the measurement error variance hardly affects the transition location and diagnosis performance with the QPE method.

Ease of Implementation. The method is based on the combined application of kernel regression as a smoother and the Viterbi algorithm as a path estimation method. While this combination is unique and novel, the fact that smoothers and path estimation methods have been developed and studied extensively, means that the method is straightforward to implement, either from scratch or based on pre-existing software. This is considered an important advantage over the SCS method, which is less intuitive and requires specialized software for second order cone programming and branch-and-bound optimization.

Weaknesses

The QPE method is characterized by a few drawbacks. These are important to keep in mind even though these do not prevent application of the method:

Lack of Statistical Optimality. Even if the constituting tools of the proposed method (smoothing and path estimation) are optimal by themselves and for their original purposes, the proposed combination requires approximations and assumptions which are questionable in the light of statistical theory. Most importantly, the applied sensor equation in the HMM erroneously assumes independence of the estimates of the derivatives obtained through smoothing and the resulting probabilities of the primitives. In reality, the smoothing operation leads to unavoidable correlation between derivatives of different order and at different locations in the data series. The observed robustness of the method to this lack of theoretical optimality is likely application-dependent. To a lesser extent, the absence of a joint likelihood function and associated generative properties can also be considered a drawback of the method.

Necessity of Tuning. The proposed QPE method thanks its excellent performance due to an inherent flexibility obtained using a smoother. Indeed, by selecting the kernel support half-width (τ) one can fine-tune the method for the intended application. However, such tuning is necessary for every new application. As demonstrated by the benchmarking study in this work, even a change in measurement noise warrants adjustment of the kernel half-width. In contrast, the SCS method does not require such tuning. One should thus trade (inexpensive) computing time for the SCS method against (human, expensive) time required to fine-tune the QPE algorithm. Once more, this trade-off is expected to be case-specific.

Opportunities

Having established the QPE method for univariate data series, a few opportunities arise:

Multivariate Data Series. So far, only one QTA method can deal explicitly with multivariate trends.[9] The QPE method is expected to lend itself to a multivariate application setting as well.

Zero-Valued Derivatives. In the above, it was assumed that the noiseless signal does not exhibit segments with zero-valued derivatives (linear and steady-state trends) or, alternatively, that one does not care to identify them as such. This was found sufficient, as before, for fault diagnosis of a simulated batch process. Should recognition of such features be warranted, then the QPE method should be extended for this.

Alternative Applications. So far, the SCS and QPE methods have been studied primarily in a fault diagnosis application context where they are applied to time series data. It remains to be evaluated whether these methods are also applicable for other data or even other goals such as data reconciliation, data mining,[23] and control.[24]

On-Line and Real-World Application. Both the SCS and QPE method have been used for off-line diagnosis of a simulated batch process. However promising, the ultimate test of such method lies with their on-line and real-world application. To this end, a modification of the QPE algorithm, called qualitative state estimation, has been proposed for on-line control of the full-scale Hard wastewater treatment plant in Winterthur (Switzerland).[25]

Threats

The QPE method cannot be applied if the following requirements cannot be met:

Library of Qualitative Sequences and Associated Process Conditions. The QPE method assumes that one or more qualitative sequences (QSs) are given for data series segmentation. For fault diagnosis, it is an additional requirement that each QS is associated uniquely with a single process condition. These requirements are met when long term experience with a process and its malfunctions can be expressed this way. Note that this does not mean that every possible qualitative sequence should have been experienced. It is, however, necessary that an expert or operator assigns a likely cause or process condition to each feasible qualitative sequence.[24] If this cannot be met, then the QPE algorithm cannot be applied in its current form. Most of the existing QTA techniques, except the SCS and QPE method, deal with this effectively already and allow to obtain new QSs and QRs with limited restrictions. Thus, the SCS and QPE methods, while high-performing, are limited in their application range. For example, these algorithms cannot be applied for data mining or to continuous-flow systems. The development of the qualitative state estimation algorithm mentioned above partly addresses this challenge by permitting the use for continuous-flow systems.[25] Methods which enable data mining on the basis of modified SCS or QPE methods are not established yet.

Discontinuities. As indicated, the QPE method cannot deal with discontinuities in the 1st derivative. For example, a CA sequence would imply a discontinuity, which cannot be handled efficiently within the kernel regression framework. Quite critically, to the best of the author's knowledge, the QPE method cannot be extended for discontinuities. The pre-existing SCS method has been extended for discontinuous behaviors, however.[26] In the mean time, some of the existing piece-wise polynomial methods reported before offer opportunities for such cases.[12,27]

Conclusions

An analysis of the spectrum of data analytic methods suggested that a reliable yet efficient algorithm for QTA is not available in the existing literature. Existing methods are either plagued by theoretical and/or practical suboptimality or high computational demand. For this reason, a new algorithm, named QPE, was devised with the intention to provide a compromise between accuracy and computational requirements. Following detailed study and comparison with earlier benchmarking results, it is concluded that the QPE indeed offers such a compromise. Interestingly, tuning of the QPE method leads to a diagnostic performance comparable to the previously developed SCS method while using the QPE method reduces computational requirements substantially. In addition to this excellent performance, the discussion section of this article also describes the method's weaknesses (e.g., the requirement for tuning), opportunities (e.g., multivariate and on-line applications), and threats (e.g., discontinuous trends, known qualitative sequence library). In summary, the QPE method provides a validated improvement over the existing methods in the QTA literature.

Acknowledgment

The author wishes to thank Dr. Raghunathan Rengaswamy (IIT Madras, Chennai, India) for helpful discussions leading to this article and Prof. Dr. Morgenroth (Eawag, Dübendorf, Switzerland; ETHZ, Zürich, Switzerland) for critical evaluation.

Literature Cited

1. Gertler J, Li W. Isolation enhanced principal component analysis. *AIChE J*. 1999;45:323–334.
2. Prakash J, Patwardhan SC, Narasimhan S. A supervisory approach to Fault-Tolerant control of linear multivariable systems. *Ind Eng Chem Res*. 2002;41:2270–2281.
3. Kuipers BJ. Qualitative Reasoning: Modeling and Simulation with Incomplete Knowledge. Cambridge, MA: MIT Press, 1994.
4. Venkatasubramanian V, Rengaswamy R, Kavuri SN. A review of process fault detection and diagnosis—part II: Qualitative models and search strategies. *Comput Chem Eng*. 2003;27:313–326.
5. Fu TC. A review on time series data mining. *Eng Appl Artif Intell*. 2011;24:164–181.
6. Maurya MR, Rengaswamy R, Venkatasubramanian V. Fault diagnosis using dynamic trend analysis: a review and recent developments. *Eng Appl Artif Intell*. 2007;20:133–146.
7. Rengaswamy R, Venkatasubramanian V. A syntactic pattern-recognition approach for process monitoring and fault diagnosis. *Eng Appl Artif Intell*. 1995;8:35–51.
8. Bakshi BR, Stephanopoulos G. Representation of process trends—part III. Multiscale extraction of trends from process data. *Comput Chem Eng*. 1994;18:267–302.
9. Flehmig F, Watzdorf R, Marquardt W. Identification of trends in process measurements using the wavelet transform. *Comput Chem Eng*. 1998;22:S491–S496.
10. Bakhtazad A, Palazoglu A, Romagnoli JA. Detection and classification of abnormal process situations using multidimensional wavelet domain hidden Markov trees. *Comput Chem Eng*. 2000;24:769–775.
11. Wong JC, McDonald KA, Palazoglu A. Classification of abnormal plant operation using multiple process variable trends. *J Process Control*. 2001;11:409–418.
12. Dash S, Maurya MR, Venkatasubramanian V, Rengaswamy R. A novel interval-halving framework for automated identification of process trends. *AIChE J*. 2004;50:149–162.
13. Charbonnier S, Garcia-Beltan C, Cadet C, Gentil S. Trends extraction and analysis for complex system monitoring and decision support. *Eng Appl Artif Intell*. 2005;18:21–36.
14. Ng AY, Jordan MI. On discriminative vs. generative classifiers: a comparison of logistic regression and naive Bayes. *Adv Neural Inf Process Syst*. 2002;2:841–848.
15. Bishop CM, Lasserre J. Generative or discriminative? Getting the best of both worlds. *Bayesian Stat*. 2007;8:2–23.
16. Villez K, Rengaswamy R, Venkatasubramanian V. Generalized shape constrained spline fitting for qualitative analysis of trends. *Comput Chem Eng*. 2013;58:116–134.
17. Villez K, Rosén C, Anctil F, Duchesne C, Vanrolleghem PA. Qualitative representation of trends (QRT): extended method for identification of consecutive inflection points. *Comput Chem Eng*. 2012;48:187–199.
18. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning. Data Mining, Inference, and Prediction. New York: Springer, 2001.
19. Rabiner LR. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc IEEE*. 1989;77(2):257–286.
20. Russell SJ, Norvig P, Canny JF, Malik JM, Edwards DD. *Artificial Intelligence: A Modern Approach*. Englewood Cliffs: Prentice Hall, 1995.
21. Birol G, Undey C, Çinar A. A modular simulation package for fed-batch fermentation: penicillin production. *Comput Chem Eng*. 2002;26:1553–1565.
22. Monroy I, Villez K, Graells M, Venkatasubramanian V. Fault diagnosis of a benchmark fermentation process: a comparative study of feature extraction and classification techniques. *Bioprocess Biosyst Eng*. 2011;35:689–704. Available at: URL http://dx.doi.org/10.1007/s00449-011-0649-1.
23. Stephanopoulos G, Locher G, Duff M, Kamimura R, Stephanopoulos G. Fermentation database mining by pattern recognition. *Biotechnol Bioeng*. 1997;53:443–452.
24. Villez K, Rosén C, Anctil F, Duchesne C, Vanrolleghem PA. Qualitative representation of trends: an alternative approach to process diagnosis and control. *Water Sci Technol*. 2008;57(10):1525–1532.
25. Thürlimann C, Dürrenmatt D, Villez K. Evaluation of qualitative trend analysis as a tool for automation. In: *12th International Symposium on Process Systems Engineering and 25th European Symposium on Computer Aided Process Engineering (PSE2015/ESCAPE25)*, Copenhagen, Denmark, 31 May - 4 June 2015, In Press.
26. Villez K, Habermacher J. Shape Constrained splines with discontinuities for anomaly detection in a batch process. In: *12th International Symposium on Process Systems Engineering and 25th European Symposium on Computer Aided Process Engineering (PSE2015/ESCAPE25)*, Copenhagen, Denmark, 31 May - 4 June 2015, In Press.
27. Charbonnier S, Gentil S. A trend-based alarm system to improve patient monitoring in intensive care units. *Control Eng Pract*. 2007;15:1039–1050.

Manuscript received June 17, 2014, and revision received Dec. 17, 2014.