Biomass estimation in batch biotechnological processes by Bayesian Gaussian process regression

Fernando di Sciascio*, Adriana N. Amicarelli
Instituto de Automática (INAUT), Universidad Nacional de San Juan, Av. Libertador San Martín 1109 (oeste), J5400ARL San Juan, Argentina

A R T I C L E   I N F O

Article history:
Received 21 December 2007
Received in revised form 26 April 2008
Accepted 29 May 2008
Available online 5 June 2008

Keywords:
Bacillus thuringiensis
Biotechnological process
Biomass estimation
Extended Kalman filter
Bayesian regression
Gaussian process

A B S T R A C T

This paper proposes a biomass concentration estimator for a batch biotechnological process based on Bayesian regression with Gaussian process. On the basis of experimental data, a two-stage bootstrap technique has been developed for the estimator design. In the first stage, the biomass data set was augmented with virtual filtered measurements, and in the second stage, the biomass estimator design was completed. The method provides information on the confidence level of the estimates, and the biomass estimator performances are illustrated for the Bacillus thuringiensis H9254-endotoxins production process.

© 2008 Elsevier Ltd. All rights reserved.

1. Introduction

Bioreactors are engineered systems in which the activity of living cells is regulated to produce a specific output. A bioreactor can be described as a heterogeneous system with at least three structural phases: the liquid phase, the gas phase, which together form the abiotic phase, and the biological phase, which consists of the cell population. The cells must be viable and also be maintained in a suitable state of metabolic activity that facilitates the cell growth and product synthesis by means of an appropriate control of the abiotic phase. Nowadays, most of the advanced process control techniques are based on state space models (Henson, 2006; Henson & Seborg, 1997), i.e., all states of the bioreactor must be known. Although some states in a biotechnological process can be easily measured online, e.g., dissolved oxygen concentration, temperature in non-isothermal reactors and gaseous flow rates, this is not normally possible or is economically non-feasible for biomass concentration. From the process control point of view, the biomass concentration is one of the states that characterizes the bioprocess; moreover, it is generally the main desired output, and therefore, for control purpose it is necessary to replace the unavailable biomass concentration measurements with reliable and robust online estimations.

The time evolution of biomass is conceived as a dynamic system perturbed by a certain process noise. Although the bioprocess is not truly stochastic, this noise is used for modeling the uncertainties in the system dynamics, i.e., the stochasticity is only used for representing the model uncertainties. Biomass concentration estimation is obtained indirectly through observed noisy measurements. Noise in the measurements refers to a disturbance in the sense that the measurements are uncertain, i.e., even in the hypothetical case that the true biomass concentration is known, the measurements would not be deterministic functions of this true biomass, but would have certain distribution of possible values. The major difficulty when the biomass estimation is implemented is related to the uncertainty of the models used to describe their dynamics. Each estimation method has its own advantages and drawbacks according to their ability to take into account the model uncertainties and the measurement errors.

Several more or less equivalent expressions for state estimators can be found in the biotechnological process literature, for example, state observers, optimal filters, observer-based estimators, software sensors, virtual sensors, model based sensor, indirect sensor, and so forth.

The design and the application of biomass estimators have been an active area of research over the past four decades. During that time, based on very different ideas, numerous estimation techniques considering the nonlinear models involved in bioprocesses...
have been proposed in the literature. A review of the commonly used techniques can be found in (Bastin & Dochain, 1990; Dochain, 2003) and references therein.

Observers found in the literature can be coarsely divided in two broad classes namely, first principles or phenomenological estimators and empirical estimators.

The phenomenological estimators can be subdivided in classical observers and asymptotic observers: classical observers include extended Kalman filter (EKF), extended Luenberger observer, the high gain observer, nonlinear observers, and the full horizon observer. In this class of estimators, the detailed knowledge of the reaction kinetics and associated phenomena are required to represent the balance equations. Modeling the biological kinetics reaction is a difficult and time-consuming task, and therefore the model used by the estimators could differ significantly from the reality. This is the main drawback of these phenomenological estimators, i.e., their efficiency strongly relies on the model quality. It should be noted that, except for the Kalman filter, and EKF that needs statistical knowledge (covariance matrices) of the noises acting on the states and on the output, these techniques were mainly developed in a completely deterministic context. On the other hand, asymptotic observers (Bastin & Dochain, 1990) are based on the idea that uncertainty in bioprocess models lies in the process kinetics models. The design of these observers is based on a state transformation in order to provide a model which is independent of the kinetics. The potential drawback of the asymptotic observers is that the rate of convergence is completely determined by the operating conditions, i.e., the rate of convergence can be very slow or cannot converge at all (Dochain, 2003).

Empirical estimators are based on constructing appropriate nonlinear models of biotechnological processes exclusively from the process input–output data without considering the functional or phenomenological relations between the bioprocess variables. However, the conventional empirical modeling approach must know the structure (functional form) of the data-fitting model in advance. This is a difficult task since it involves selecting heuristically an appropriate nonlinear model structure from numerous alternatives.

For machine learning community, data-based modeling the biomass concentration from a finite number of noisy samples (the training dataset) is a supervised learning problem. From this area, in recent years, the artificial neural network methodology has become one of the most important techniques applied for biomass estimation, e.g. (Amicarelli, di Sciascio, Álvarez, & Ortiz, 2006; Leal, 2001; Li, 2003) and references therein.

Neal’s work on Bayesian learning for neural networks (Neal, 1996) shows that many Bayesian regression models based on neural networks converge to a class of probability distributions known as Gaussian processes as the number of hidden neurons tends to infinity. Furthermore, Neal argued that in the Bayesian approach for real-world complex problems, neural network models should not be limited to nets containing only a small number of hidden units. Neal’s observation motivates the idea of replacing parameterized neural networks and working directly with Gaussian processes models for the high-dimensional applications to which neural networks are typically applied (Neal, 1997).

This work proposes a biomass concentration estimator for a biotechnological batch process based on Bayesian regression with Gaussian process. To the best of author’s knowledge, this is the first time that this type of biomass estimator for a real bioprocess is reported. The design is exemplified for a Bacillus thuringiensis δ-endotoxins production process on the basis of experimental data from a set of various fermentations.

For estimation purpose, the main variables of this bioprocess are: the concentrations of dissolved oxygen, primary substrate (basically glucose), and biomass. The collected data from these signals have been sampled at different sampling rates, 10 per hour for dissolved oxygen and substrate concentrations and, 1 per hour for the biomass concentration. A two-stage bootstrap technique has been developed for the biomass estimator design. In the first stage, the experimental biomass concentrations data set can be completed with “virtual filtered measurements” to have the same size as in dissolved oxygen and primary substrate data sets. This is a missing data problem (Little & Rubin, 2002), and Bayesian Gaussian process regression is proposed as an imputation strategy for filling missing values. In the second stage, again a Gaussian process regression schema is used for biomass estimation.

The paper is organized as follows: Section 2 presents the main characteristics of the B. thuringiensis (Bt) δ-endotoxins production process, and also a phenomenological model of the bioprocess is given. Section 3 is devoted to describing the Bayesian Gaussian process regression framework. Next, in Section 4 the proposed biomass estimator design is presented, whereas Section 5 gives an analysis of results and discussion. Finally, in Section 6 the conclusions are stated.

2. B. thuringiensis δ-endotoxins production process

2.1. Bioprocess description

In the last years, due to environmental reasons the interest in biological agents for their use in ecological insecticides (bioinsecticides) has notably increased. B. thuringiensis is one of the microorganisms most frequently studied as toxin producer. Bt is an aerobic spore-former bacterium which, during the sporulation; also produces insecticidal crystal proteins known as δ-endotoxins. It has two stages on its life span: a first stage characterized by its vegetative growth, and a second stage named sporulation phase. When the vegetative growth finalizes, the beginning of the sporulation phase is induced when the mean exhaustion point has been reached. Normally the sporulation is accompanied by the δ-endotoxin synthesis. After the sporulation, the process is completed with the cellular wall rupture (cellular lysis), and the consequent liberation of spores and crystals to the culture medium (Aronson, 1993; Liu & Tzeng, 2000; Starzak & Bajpai 1991).

This research has been conducted with the same process and fermentation conditions as the work of Atehortúa, Alvarez, and Orduz (2007). The microorganisms used in this work were B. thuringiensis serovar. kurstaki strain 172-0451 isolated in Colombia and stored in the culture collection of Biotechnology and Biological Control Unit (CIB) (Vallejo, González, Posada, Restrepo, & Orduz, 1999). The medium (CIB-1) contained: MnSO₄.H₂O (0.03 g/L), CaCl₂.2H₂O (0.041 g/L), KH₂PO₄ (0.5 g/L), K₂HPO₄ (0.5 g/L), (NH₄)₂SO₄ (1 g/L), yeast extract (8 g/L), MgSO₄.7H₂O (4 g/L) and glucose (8 g/L). Growth experiments of the fermentation process with B. thuringiensis were performed in a pilot-scale reactor with a nominal volume of 20 L (Fig. 1). The fermentations were developed with an effective volume of 11 L of cultivation medium, and they were inoculated to 10% (v/v) with the microorganism Bt culture. The inoculum added consisted of a vegetative phase culture: 5 mL spore suspension with 1 · 10⁷ UFC/mL (stored at −20 °C) was used to inoculate a 500 mL flask containing 100 mL of CIB-1, and incubated with shaking at 250 rpm at 30 °C during 13 h. Fifty milliliters of this culture were aseptically transferred to each one of two 2 L flasks containing 500 mL of CIB-1 and incubated as above for 5 h. The pH medium was adjusted to 7.0 with KOH before its heat sterilization. Culture conditions at harvest are typified by 90% free spores and δ-endotoxins crystals.
The temperature was maintained around 30 °C by using an ON/OFF control; whereas the pH was fixed between 6.5 and 8.5. The airflow was set up at 1320 L/h and the agitation speed at 400 rpm. Manometric pressure in the reactor was set at 41,368 Pa using a pressure controller. Temperature, pH, dissolved oxygen, and glucose concentration were registered by a data acquisition system using an Advantech® PCL card. Dissolved oxygen was measured by a polarographic oxygen sensor InPro 6000 (Mettler Toledo, Switzerland), and glucose concentration was determined with a rapid off-line measurement method through a glucose analyzer (YSI 2700).

The reagents concentration used for the pH control and foam formation were nitric acid (5N), potassium hydroxide (2N) and defoamer (33%, v/v). Cell growth was determined as dry cell weight (dry cell weight (DCW, g/L) = (final weight − initial weight)/(volume of microbial suspension filtered). The foam formation was avoided by manually aggregating a defoamer.

*B. thuringiensis* δ-endotoxins production is an aerobic operation, i.e., the cells require oxygen as a substrate to achieve cell growth and product formation (Ghribi, Zouari, Trabelsi, & Jaoua, 2007).

### 2.2. Bioprocess model

The biomass concentration estimator proposed in this work is founded on Bayesian regression with Gaussian process, this is a non-parametric technique based on experimental data, and a qualitative knowledge of the bioprocess. Therefore, the phenomenological model presented in this section is not necessary for its design. However, their inclusion has a twofold aim: firstly, to provide a fair basis of discussion for the appropriateness of the proposed approach; and, secondly, to give an explicit mathematical model for an extended Kalman filter design. As pointed out in Section 1, the EKF is a classical nonlinear state estimator, and it is implemented for the purpose of comparing with our biomass concentration estimator.

A first principle-based model for *Bt* δ-endotoxins production process consists of a set of differential and algebraic equations.
tions in the continuous-time case, and a set of difference and algebraic equations in the discrete-time case. A simple phenomenological model was proposed by Rivera, Margaritis, and De Lasa (1999), a modification to the Rivera model was given by Atehortúa, Álvarez, and Orduz (2006) and Atehortúa et al. (2007). Afterwards, Amicarelli et al. (2006), Amicarelli, di Sciascio, and Álvarez (2008) improved the model process adding the dissolved oxygen (DO) dynamics due to its importance in the biomass estimation problem and the posterior process control. The following state space model is a discrete-time version of the continuous-time counterpart developed by Amicarelli et al. (2008):

\[
\begin{align*}
\begin{bmatrix}
X_v(k+1) \\
X_s(k+1) \\
S_p(k+1) \\
DO(k+1)
\end{bmatrix}
&=egin{bmatrix}
\frac{(\mu(k) - k_s(k) - k_e(k)TS + 1)X_v(k)}{k_e(k)X_v(k)TS + X_s(k)} \\
\frac{(\mu(k) + m_s)X_v(k)TS + S_p(k)}{X_v(k)TS + S_p(k)} \\
\frac{Ts(k_i(k_e(k) - \mu(k)) + k_2)X_v(k) + k_3 TsX_v(k) + DO(k) + k_4 QsTs(DO^* - DO(k))}{X_v(k)TS + S_p(k)} \\
\frac{Ts(k_i(k_e(k) - \mu(k)) + k_2)X_v(k) + k_3 TsX_v(k) + DO(k) + k_4 QsTs(DO^* - DO(k))}{X_v(k)TS + S_p(k)} \\
\end{bmatrix}
\end{align*}
\]

where \(X_v\) is the vegetative cell concentration, \(X_s\) the sporulated cell concentration, \(S_p\) the limiting substrate concentration and \(DO\) the dissolved oxygen concentration.

The following algebraic equations define the specific growth speed (Monod equation) \(\mu\), the sporulation rate \(k_s\), and the death cell-specific rate \(k_e\):

\[
\begin{align*}
\mu(k) &= \frac{\mu_{max}S_p(k)}{K_s + S_p(k)} \\
k_s(k) &= k_s \max \left(\frac{1}{1 + e^{Ge(S_p(k) - Ps)}}\right) \\
k_e(k) &= k_e \max \left(\frac{1}{1 + e^{Ge(Tsk - Pe)}}\right)
\end{align*}
\]

The complete notation and model parameter’s values are presented in Tables 1 and 2.

Four batch cultures with different initial glucose concentration (8, 21, 32 and 40 g/L) were carried out to generate experimental data for model validation and fine parameters tuning. In this context, four parameter sets guarantee a representative covering of an intermittent fed batch culture (IFBC) with total cell retention (TCR) in the operation space according to the work of Atehortúa, see Table 2.

Maximun glucose concentration in the medium \((S_{p max})\) was used as the switching criteria among the estimated batch parameter sets.

### Table 2

<table>
<thead>
<tr>
<th>Parameter</th>
<th>(S_p &lt; 10 \text{ g/L})</th>
<th>(10 \text{ g/L} &lt; S_p &lt; 20 \text{ g/L})</th>
<th>(20 \text{ g/L} &lt; S_p &lt; 32 \text{ g/L})</th>
<th>(S_p &gt; 32 \text{ g/L})</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\mu_{max} (h^{-1}))</td>
<td>0.8</td>
<td>0.7</td>
<td>0.65</td>
<td>0.58</td>
</tr>
<tr>
<td>(Y_v (g/g))</td>
<td>0.7</td>
<td>0.58</td>
<td>0.37</td>
<td>0.5</td>
</tr>
<tr>
<td>(k_s (g/L))</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(m_s (g/g/L))</td>
<td>0.005</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(k_e (h^{-1}))</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(G_s (g/L)^{-1})</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(P_s (g/L))</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(k_{max} (h^{-1}))</td>
<td>0.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(G_s (mol/L)^{-1})</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Ps (h))</td>
<td>4</td>
<td>4.7</td>
<td>4.9</td>
<td>6</td>
</tr>
<tr>
<td>(k_e (\text{dimensionless}))</td>
<td>9.725 \times 10^{-4}</td>
<td>4.502 \times 10^{-3}</td>
<td>3.795 \times 10^{-3}</td>
<td>1.597 \times 10^{-3}</td>
</tr>
<tr>
<td>(k_s (h^{-1}))</td>
<td>1.589 \times 10^{-4}</td>
<td>0.046 \times 10^{-3}</td>
<td>0.729 \times 10^{-3}</td>
<td>0.561 \times 10^{-3}</td>
</tr>
<tr>
<td>(k_e (L^{-1}))</td>
<td>4.636 \times 10^{-4}</td>
<td>0.337 \times 10^{-3}</td>
<td>2.114 \times 10^{-3}</td>
<td>1.045 \times 10^{-3}</td>
</tr>
<tr>
<td>(Ts (h))</td>
<td>0.1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 3. Bayesian Gaussian process regression

#### 3.1 Bayesian regression framework

Suppose that we have a noisy training data set \(D\) which consists of \(m\) pairs of \(n\)-dimensional input vectors \(\{x_i\}\) (regression vector) joined in an \(n \times m\) matrix \(X\), and \(m\) scalar noisy observed outputs \(\{y_i\}\) collected in a vector \(y\):

\[
D = \{(x_i, y_i)\}_{i=1}^{m} = (X, y)
\]

In order to construct a probabilistic statistical model for \(D\), the following data-generating process is assumed:

\[
y_i = f(x_i) + \epsilon_i
\]

where the latent real-valued function \(f\) is the deterministic or systematic component of the model, and the additive random term \(\epsilon\) is the observation error. The aim of regression is to identify the systematic component \(f\) from the empirical observations \(D\). The Bayes’ rule (6) shows the components of Bayesian inference (Bernardo & Smith, 2006): the joint likelihood, the prior distribution, and the posterior distribution. Bayesian inference alludes to the process of updating our beliefs according to Bayes’ rule, i.e., computing the posterior from likelihood and the prior, integrating the information contained in the observed data:

\[
p(T|X, \Theta, \Theta') = \frac{p(y|f, \Theta, \Theta')p(f|X, \Theta')}{p(\Theta|\Theta')}
\]

where \(P(D|\Theta) = P(y|X, \Theta, \Theta') = P(y|x_i, \Theta, \Theta') = y|x_i\) are the latent function values, and \(\Theta, \Theta'\) denote additional parameters (hyperparameters) of the likelihood and prior distribution, respectively.

The evidence or marginal likelihood is the normalising constant appearing in the denominator of Bayes’ rule. This quantity is one of the most useful quantities in the Bayesian framework (e.g., in hypothesis testing applications) (Bernardo & Smith, 2006; O’Hagan, 2004), however the evidence is not considered in the remainder of the exposition.
If the measurements of the training data set \( \mathcal{D} \) are independent, and the observation error \( e \) it is assumed that is normal, independent and identically distributed with mean zero and variance \( \sigma^2 \), then, in this case the joint likelihood is

\[
p(y|f, \sigma^2) = \prod_{i=1}^{m} N(f_i, \sigma^2) = N(f, \sigma^2 I_{m \times m})
\]  

(7)

In the Bayesian non-parametric approach, a prior is put directly on the space of functions and the inference is carried out on \( f \). The prior distribution is usually chosen from a parametric family of distributions or a mixture of these. The expression "Gaussian process regression model" refers to using a Gaussian process as a prior on \( f \). This means that every finite-dimensional marginal joint distributions of function values \( f \) associated to any input subset of \( X \) is multivariate Gaussian:

\[
p(f|X, \theta_p) = N(m(X), K(X, \theta_p))
\]  

(8)

A Gaussian process is fully specified by a mean function \( m(X) = [m(x_1), \ldots, m(x_m)]^T \) and a positive definite covariance matrix \( K(X, \theta_p) \), and it can be viewed as a generalization of the multivariate Gaussian distribution to infinite dimensional objects. Choosing a particular form of covariance function, the hyperparameters \( \theta_p \) may be introduced to the Gaussian process prior. Depending on the actual form of the covariance function \( K(X, \theta_p) \) the hyperparameters \( \theta_p \) can control various aspects of the Gaussian process. For simplicity the prior mean function is set to be \( m(X) = 0 \), this is completely general provided that a constant term is included in the covariance function (Kuss, 2006; Williams & Rasmussen, 1996).

The posterior distribution over function values is obtained introducing (7) and (8) in (6):

\[
p(f|D, \sigma^2, K) \propto N(f, \sigma^2 I_{m \times m}) N(0, K) \propto N(K + \sigma^2 I_{m \times m})^{-1} f, y,
\]  

(9)

The distribution of the latent function value \( f_i = f(x_i) \) for an arbitrary new input \( x_i \) conditioned on the training function outputs is (Kuss, 2006):

\[
p(f_i|x_i, X, K) \propto N(K^{i-1} f, k_{xx} - K^{i-1} K_{xx})
\]  

(10)

The predictive distribution of \( f_i \) is obtained by integrating out the training function values \( f \) from (10) over the posterior distribution (9). The predictive distribution is again multivariate normal:

\[
p(f_i|D, x_i, K, \sigma^2) = \int p(f|x, X, K)p(f|D, \sigma^2, K)df \propto N(K^{i-1} C^{-1} y, k_{xx} - K^{i-1} K_{xx})
\]  

(11)

The predictive uncertainty, i.e., the covariance matrix of \( f_i \), does not depend on \( y \), but only on the dependencies induced by the covariance as a function of \( x \) and \( X \). This can be generalized to an arbitrary set of new inputs \( X' \), meaning that the posterior process \( f|D \) is again a Gaussian process with posterior mean and covariance function:

\[
\hat{f}_i = E(f|x_i, X, K, \sigma^2) = K_{xi} C^{-1} y
\]  

\[
\sigma^2_i = k_{xx} - K_{xi} C^{-1} K_{xi}
\]  

(12)

where \( C = K(X, \theta_p, \sigma^2) = K(X, \theta_p) + \sigma^2 I_{m \times m} \).

The resulting posterior (9) and the predictive distribution (11) are of the same family of distributions as the prior (8). The class of prior distributions with this property is called conjugate to a likelihood model (Bernardo & Smith, 2006; O'Hagan, 2004). The calculations are analytically tractable only for conjugate models with normal noise. For all other models the posterior and the predictive distribution cannot be computed analytically, so techniques for approximate inference have been used, for example, Markov Chain Monte Carlo (MCMC) sampling techniques (Kuss, 2006; Neal, 1997; Williams & Rasmussen, 1996).

### 3.2. Covariance function

The elements of the parameterized covariance matrix, \( C(X, \theta_p, \sigma^2) \), are denoted \( C_{ij} = C(x_i, x_j) \), and they are functions of the training input data \( X \), because these data determine the correlation between the training data outputs \( y \).

A suitable parametric form of the covariance function is

\[
C_{ij} = \theta_0 + \theta_1 \exp \left( -\sum_{l=1}^{n} \frac{(x_{i}^{(l)} - x_{j}^{(l)})^2}{r_l^2} \right) + \theta_2 \delta(i, j) + \theta_3 (x_{i}^{(l)} x_{j}^{(l)})
\]  

(13)

where \( x_{i}^{(l)} \) is the \( l \)-th dimension of the input vector, \( x_i \).

The four terms in this equation are now briefly described.

(i) A bias term \( \theta_0 \) controlling the scale of the bias contribution to the covariance. The constant term \( \theta_0 \), adds a constant offset to the estimated latent function value \( f = f(x_i) \). This justifies assigning \( m(X) = 0 \) to the prior mean function (8) without loss generality.

(ii) The exponential term (involving \( r_l \) and \( r_l \)) expresses our belief that inputs which are close to each other give rise to outputs which are close to each other or that are highly correlated; the \( r_l \) hyperparameters allow a different distance measure for each input dimension, and \( \theta_1 \) gives the overall scale of variations in the output space (local correlations).

(iii) The third term involving the hyperparameter \( \theta_2 = \sigma^2 \), i.e., the variance of the noise model for the outputs and therefore only occurs in \( C_{ij} \) when \( i = j \).

(iv) The fourth and last term characterize the non-stationarity of the covariance function. It is a linear regression term and involving \( \alpha_i, i = 1, \ldots, n \), these hyperparameters controlling the scale of the linear trends to the covariance.

Besides (13), there are other forms of the covariance function which could be used. The only restriction is that the covariance matrix be positive definite. Abrahamsen has written a comprehensive survey with on numerous valid covariance functions (Abrahamsen, 1997).

### 3.3. Hyperparameters determination

The hyperparameters of the covariance function are not known in advance, and they must be determined using the training data. The literature has reported several approaches to hyperparameter estimation: cross-validation method (Wahba, 1990), evidence maximization (Gibbs, 1997; MacKay, 1992), Monte Carlo methods (Neal, 1997), maximum likelihood, and maximum a posteriori method (Rasmussen, 1996). In this work the last method is used, i.e., the maximum a posteriori (MAP) approach.

For implementation purpose, the hyperparameters vector is defined as

\[
\log \theta = [\log \theta_0, \log \theta_1, \log r_1, \ldots, \log r_n, \log \theta_2, \log \alpha_1, \ldots, \log \alpha_n]^T
\]

The number of hyperparameters of the covariance function (13) increases linearly with \( n \), the dimension of the input space, i.e.,
Moreover, direct inversion implementation can run into numerical issues by using Cholesky, LU, or SVD decomposition (note that the positive definite property of the covariance matrix is guaranteed). The prior on \( p\) is assumed to be independent of the other priors.

The MAP estimates are found using (15) in a gradient descent, or conjugate-gradient optimization techniques to locate a local maximum of the posterior. The algorithm can get trapped in bad local maxima. In order to reduce this problem, suitable priors can be assigned and multiple random restarts for the optimization routines can be fulfilled. The initialization of the hyperparameters is important, because improper initial values will make the partial derivatives of the likelihood very small, thus creating problems for the optimization algorithm.

In order to implement the algorithm (Eqs. (12), (13), and (15)) it was necessary to invert the covariance matrix \( C \). Any exact inversion method has an associated computational cost that is \( O(m^3) \), moreover, direct inversion implementation can run into numerical problems because \( C \) is generally ill-conditioned, i.e., the condition number is large. In order to improve the condition number of the matrix inversion operation, \( C^{-1} \) can be computed indirectly by using Cholesky, LU, or SVD decomposition (note that the positive definiteness property of the covariance matrix is guaranteed). All these methods also require \( O(m^3) \) operations (Golub & Van Loan, 1990).

Observe that the optimization routines requires in each step the evaluation of the gradient of the log likelihood, i.e., the computation of \( C^{-1} \) and so calculating gradients becomes time consuming for large training data sets. If \( m \) is large (\( m > 10^3 \)), Skilling’s approximate inversion methods, which are \( O(m^2) \), can be used (Skilling, 1993).

### 4. Biomass estimator design

The duration of the batch fermentation is limited and depends on the initial conditions of the microorganism culture. All the fermentations were initialized with the same inoculate and different substrate concentration conditions (Aethertúa et al., 2007). When the medium is inoculated, the biomass concentration increases at an expense of the nutrients, and the fermentation concludes when the glucose that limits its growth was consumed, or when 90% or more of cellular lysis is presented. After that, the latency period was removed (the bioprocess dead time is not considered), and the duration of each experiment is approximately 18 h.

The collected data from the fermentations is a set of concentrations measurements of dissolved oxygen (DO), primary substrate (\( S_p \)), and biomass (\( X \)) which have been sampled at different speed, 10 per hour for the concentrations of dissolved oxygen and glucose and 1 per hour for the biomass concentration, that was quantified by a cell dry weight method. Practically, DO could continuously be measured, and \( S_p \) up to 20 times per hour. From the bandwidth estimation of system signals by using Fourier frequency analysis, the sampling time \( T_s = 1/10\ h \) has been selected for dissolved oxygen and substrate measurements.

In order to design a biomass estimator for the \( B. thuringiensis \) δ-endotoxins production process, we propose a two-stage method. In the first stage, the biomass concentrations data set can be completed to have the same size as in dissolved oxygen concentration and primary substrate (glucose) concentration data sets. This is a missing data problem (Little & Rubin, 2002), and Bayesian Gaussian process regression is utilized as an imputation strategy for filling the missing values. In the second stage, again a Gaussian process regression schema is used for biomass estimation.

### 4.1. First stage design—filling the biomass missing data

In this section, the biomass concentration data vector can be completed with virtual filtered measurements to have the same size as dissolved oxygen and substrate data vectors. As pointed out previously, this is a missing data problem, and the Gaussian process regression will be used as imputation method for filling the missing values (note that this task in a deterministic framework which can be viewed as a curve-fitting or interpolation problem).

For all experimental fermentations, the data-generating model for biomass concentration is

\[ X(t_k) = \hat{X}(t_k) + \epsilon(t_k) \quad (16) \]

The training data set \( D \) consists of 18 pairs of time inputs \( t = (t_k) = \{1, \ldots, 18\} \) (in hours), and noisy biomass measurements outputs \( X = (X_k) = \{X(t_1), \ldots, X(t_{18})\} \). The latent functions \( \hat{X} = (\hat{X}_1, \ldots, \hat{X}_{18}) \) are the estimated biomass concentrations.

From the training data \( D \), and by means of a conjugate-gradient routine, recursively through (13), and (15), the \#\( \Theta = 5 \) hyperparameters, and the matrix \( C \) are determined.

Afterwards, at different times, \( t = 0.1, 0.2, \ldots, 17.9, 18 \) by (12) the latent functions \( \hat{X} = (\hat{X}_1, \ldots, \hat{X}_{18}) \) and the variance \( \sigma^2 \) are estimated. The expression “virtual filtered measurements”, refers to the latent functions \( \hat{X} \), because the additive normal noise \( \epsilon \) has been removed (filtered) from the “virtual measurement” \( X \) in the data-generating model (16).

Fig. 2 gives an example of completion of biomass missing data for two fermentations (F01, and F02) of a set of various representative fermentations.

### 4.2. Second stage design—Gaussian process biomass estimator

The first step in this stage design, is selecting the regressors, i.e., the components of the input vector \( x \). This is a laborious task, and has been done heuristically, chosen from numerous alternatives. The best empirical results have been achieved with:

\[ x = [DO(KTs), S_p(KTs), \hat{X}(k-1)Ts, \hat{X}(k-2)Ts] \]

where \( k = \{1, \ldots, 180\} \) is the time index, \( Ts = 1/10\ h \) is the sampling time, \( DO \) is the dissolved oxygen concentration, \( S_p \) is the substrate concentration, and \( \hat{X}(\cdot) \) is the virtual filtered biomass measurement. The training data set \( D \) consists of 180 pairs of input vectors \( [x] \) collected in a matrix \( X \in \mathbb{R}^{180 \times 18} \), and scalars outputs \( [\hat{X}(KTs)] = (\hat{x}_i) \) collected in a vector \( \hat{X} \in \mathbb{R}^{180} \) (note...
that in this section, the virtual filtered biomass measurements \( \hat{X}_k \) are considered as true observed measurements.

The data-generating process is \( \hat{X}_k = \hat{X}_k + \varepsilon_k \), with the latent function \( \hat{X}_k(\cdot) \), and the additive normal noise \( \varepsilon \). Once more, the \#\( \Theta = 11 \) hyperparameters, and the new covariance matrix \( \mathbf{C} \) is determined from (13) and (15), via a conjugate-gradient routine. Furthermore, by (12) the biomass concentration \( \hat{X}_\ast = \hat{X}(t_\ast) \) and the variance \( \sigma^2 \) are estimated for a set of different times \( \{t_\ast\} \), \( 0 \leq t_\ast \leq 18 \) h.

For the training stage, a one-step ahead predicted output schema is performed, i.e., the input measurements, DO\( (k) \), \( S_p(k) \), and the previous output measurements \( \hat{X}(k-1), \hat{X}(k-2) \) are used as regressors:

\[
\hat{X}(k) = \hat{X}(\text{DO}(k), S_p(k), \hat{X}(k-1), \hat{X}(k-2))
\]

For on-line estimation the implemented estimator is the simulated output schema, i.e., only input measurements \( \text{DO}(k), S_p(k) \) are used. The simulated output is obtained as above, by replacing the measured outputs by the simulated output from the previous steps, i.e., previous outputs from the model have to be fed back into the model computations on-line (Fig. 3):

\[
\hat{X}(k) = \hat{X}(\text{DO}(k), S_p(k), \hat{X}(k-1), \hat{X}(k-2))
\]

One-step ahead predicted output scheme is also known as nonlinear auto regressive with exogenous input model (NARX), and as series-parallel model. Furthermore, simulated output schema is known as nonlinear output error model (NOE), and as parallel model (Ljung, 2006; Narendra & Parthasarathy, 1990).

The biomass concentration of fermentations F01 and F02 from the preceding section (see Fig. 2) has been adopted as training, and validation data, respectively. Figs. 4 and 5 show the measurements of dissolved oxygen concentration (DO) and glucose concentration \( (S_p) \), respectively. Both signals have been filtered with a low-pass filter with a 1/36 Hz corner frequency.

Fig. 6 shows the results for the proposed biomass estimator. With comparison aims, it has been included the biomass estimate of a standard extended Kalman filter (see Appendix A), and the results from the previous section, i.e., the true biomass measurements, the virtual filtered biomass measurements, and the 95% confidence intervals.
tem perturbed by process noise, i.e., a stochastic process, and the

The most probable explanation for poor results in few atypical

cal fermentations, is the aggregation of antifoam throughout the

ting evolution for a particular fermentation is

The best performing estimator in the set of regressors

The analysis of results and discussion

From Fig. 6, the correct behavior of the proposed biomass esti-
mator can be clearly seen. The estimated biomass follows the true

Concentrations measurements of glucose for fermentations F01 and F02.

5. Analysis of results and discussion

Fig. 6 also shows that the behavior of the standard EKF estimator

Fig. 6. Biomass estimator performance. The bold solid-line describes the behavior of

As we pointed out in the introduction, the time evolution of

6. Conclusions

In this paper, the design of a biomass concentration estima-
tor based on Bayesian regression with Gaussian process has been

![Graph](image-url)
The authors wish to thank the organizations who contributed to the completion of this paper. In particular, we wish to thank the Universidad Nacional de Colombia Sede Medellín, and Corporación para Investigaciones Biológicas (CIB), Medellín, Colombia. This work was partially supported by the Universidad Nacional de San Juan (Argentina), and by the CONICET (the Argentine National Council for Scientific and Technological Research).

Appendix A. Extended Kalman filter implementation

The underlying theory of the EKF is largely known in the literature devoted to filtering, estimation, and control; see, for example, the classic books by Jazwinski (1970), Anderson and Moore (1979), or most recently, the book by Simon (2006). Therefore, in this appendix only brief explanations of the specific EKF implementation are given.

In the EKF framework, the state transition and observation models are nonlinear differentiable functions of the states:

state transition model: \( x(k+1) = f(x(k), u(k), k) + w(k) \)

measurements model: \( y(k) = h(x(k), k) + v(k) \)

Where \( f(\cdot, \cdot) \) is the state transition function; \( h(\cdot, \cdot) \) is the measurement function; \( x(k) \) is the system state vector with initial condition \( x(0) \sim N(x_0, Q_0) \); \( u(k) \) is the input or control vector; \( y(k) \) is the observation vector; \( w(k) \) is a discrete-time normal white noise process with mean zero and (process noise) covariance matrix \( Q \); \( v(k) \) is a discrete-time normal white noise process with mean zero and (measurements noise) covariance matrix \( R \). The initial condition \( x(0) \), and the sequences \( w(k) \), and \( v(k) \) are uncorrelated for all time shifts.

In our case, the nominal state transition model (without the process noise \( w(k) \)) is obtained by introducing (2), (3) and (4) in (1):

\[ x(k+1) = f(x(k), k) \]

The system state vector is \( x(k) = [X_c(k)X_v(k)Sp(k)DO(k)]^T \), the input vector is \( u(k) = 0 \) (the bioprocess has no external input), and the bioproces outputs (observation vector) is \( y(k) = [Sp(k)DO(k)]^T \) (see Fig. 3).

The measurement model is linear in the states:

\[ y(k) = Hx(k) \]

where

\[ H = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \]

Taking into account the scales of the outputs, a balanced linear combination of \( Sp(k) \) and \( DO(k) \) can be considered as an alternative measurement model.

\[ y'(k) = H'x(k) = \alpha Sp(k) + \beta DO(k) \]

In this measurement model \( H' = [0 \ 0 \ \alpha \ \beta] \), where \( \alpha = DO_{max}/(Sp_{max} + DO_{max}) \), and \( \beta = Sp_{max}/(Sp_{max} + DO_{max}) \).

The next step is obtain the Jacobian matrices \( \frac{\partial f(x(k), k)}{\partial x} \) and \( \frac{\partial h(x(k), k)}{\partial x} \) evaluated at \( x(k-1|k-1) \):

\[ A(k) \triangleq \left[ \begin{array}{c} \frac{\partial f(x(k), k)}{\partial x} \\ \vdots \\ \frac{\partial f(x(k), k)}{\partial x} \end{array} \right] \]

\[ H(k) \triangleq \left[ \begin{array}{c} \frac{\partial h(x(k), k)}{\partial x} \\ \vdots \\ \frac{\partial h(x(k), k)}{\partial x} \end{array} \right] \]

The entries of the matrix \( A(k) \) are

\[ a_{11} = 1 \]

\[ a_{12} = \frac{k_{s_{max}}Ts}{1 + e^{G(Sp(k)-Ps)}} - \frac{k_{s_{max}}Ts}{1 + e^{G(Sp_{initial}-Ps)}} \]

\[ a_{13} = -\frac{k_{s_{max}}Ts X_c(k) G e^{G(Sp(k)-Ps)}}{(1 + e^{G(Sp(k)-Ps)})^2} \]

\[ a_{14} = 0 \]

\[ a_{21} = 0 \]

\[ a_{22} = 1 + \frac{k_{e_{max}}Ts}{1 + e^{G(t_e(t)-Pe)}} - \frac{k_{e_{max}}Ts}{1 + e^{G(t_e(t)-Pe)}} + \frac{k_{s_{max}}Ts}{1 + e^{G(Sp_{initial}-Ps)}} \]

\[ a_{23} = \frac{k_{s_{max}}Ts X_c(k) G e^{G(Sp(k)-Ps)}}{(1 + e^{G(Sp(k)-Ps)})^2} - \frac{Ts X_c(k) G e^{G(Sp(k)-Ps)}}{K_s + Sp(k)} \]

\[ a_{24} = 0 \]

\[ a_{31} = 0 \]

\[ a_{32} = -\frac{Ts Sp(k) G e^{G(Sp(k)-Ps)}}{Y_{x_c}(K_s + Sp(k))} - Ts m_s \]

\[ a_{33} = 1 + \frac{Ts X_c(k) G e^{G(Sp(k)-Ps)}}{Y_{x_c}(K_s + Sp(k))^2} - \frac{Ts X_c(k) G e^{G(Sp(k)-Ps)}}{Y_{x_c}(K_s + Sp(k))} \]

\[ a_{34} = 0 \]

\[ a_{41} = K_2 Ts \]

\[ a_{42} = K_2 Ts - \frac{K_1 Ts Sp(k) G e^{G(Sp(k)-Ps)}}{K_s + Sp(k)} + \frac{K_1 Ts Ke_{max}}{1 + e^{G(t_e(t)-Pe)}} - \frac{K_1 Ts Ke_{max}}{1 + e^{G(t_e(t)-Pe_{initial})}} \]

\[ a_{43} = \frac{K_1 Ts X_c(k) Sp(k) G e^{G(Sp(k)-Ps)}}{(K_s + Sp(k))^2} - \frac{K_1 Ts X_c(k) Sp(k) G e^{G(Sp(k)-Ps)}}{K_s + Sp(k)} \]

\[ a_{44} = 1 - K_3 Q_T Ts \]

Finally, initializing the elements of the matrices \( P, Q \) and \( R \), we have all the components of the EKF algorithm (See Table A.1). In order to obtain the best possible fit of the EKF to the experimental
data, the elements of the matrices Q and R are empirically adjusted by simulations.

References


